

PCI Compliance

Understand and Implement Effective PCI Data Security Standard Compliance

Fourth Edition

Branden R. Williams
Anton A. Chuvakin

Technical Editor
Derek Milroy



AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

SYNGRESS.

Syngress is an Imprint of Elsevier

Syngress is an imprint of Elsevier
225 Wyman Street, Waltham, MA 02451, USA

Copyright © 2015 Elsevier Inc. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the publisher. Details on how to seek permission, further information about the Publisher's permissions policies and our arrangements with organizations such as the Copyright Clearance Center and the Copyright Licensing Agency, can be found at our website: www.elsevier.com/permissions.

This book and the individual contributions contained in it are protected under copyright by the Publisher (other than as may be noted herein).

Notices

Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data

Application Submitted

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

For information on all Syngress publications
visit our web site at <http://store.elsevier.com/>

ISBN: 978-0-12-801579-7

This book has been manufactured using Print On Demand technology. Each copy is produced to order and is limited to black ink. The online version of this book will show color figures where appropriate.



Working together
to grow libraries in
developing countries

www.elsevier.com • www.bookaid.org

Foreword

APT. Cybercrime. Hacktivism. PCI. Those are a few of the subjects that keep security leaders up at night. If you are wondering how PCI ended up on that short list and why it may cause bouts of insomnia, simply ask someone who has to deal with PCI DSS (Payment Card Industry Data Security Standard) assessments on a regular basis and you are guaranteed to receive strong responses. Yelling matches between security leaders and their PCI assessors over terms such as “segmentation,” “isolation,” “unrecoverable,” and “significant change” have become all too commonplace.

There is little argument that the prescriptive nature and detailed requirements of the DSS are a good guide for security professionals to benchmark and improve immature information security programs. However, the PCI DSS presents a paradox for mature programs. The narrow focus of the DSS on credit card data requires artificial boundaries and duplicate control investments. This can lead to more complex network and security architectures as well as increased hardware, software, and labor costs. It can, in certain situations, also lead to bad business risk decisions in order to keep non-PCI systems out of scope of the annual assessment. It is for these reasons that PCI has become a controversial, disruptive, and insomnia-inducing influence inside many large (and some medium/small) organizations.

Even if PCI DSS assessments are nothing new to you, it would probably be a good time for a refresher course in not only the basics of the PCI standard but also the changes that will be going into effect with PCI DSS 3.0. Obviously familiarizing yourself with the changes in the standard from 2.0 to 3.0 is a great start but most likely not enough. One of the best things you can do to prepare yourself for the updated standard is to read this book cover to cover. Then re-read sections on managing the assessment scope, running the PCI assessment project as an ongoing program, and how to work well with your assessors (they’re not the enemy!). Once you’ve read the book I would suggest keeping it handy as a reference guide. I know that I will have this book in my office, highlighted, bookmarked, and within easy reach over the next few years as conflicts between business requirements and PCI compliance arise.

Dan Glass
Senior Manager Information Systems Security
American Airlines

Acknowledgments

PCI DSS 3.0 is here, and boy is it a doozy! Both Anton and I are very thankful that you continue to support our efforts and read our work.

This book is dedicated to my family for supporting the effort to make this work the central tome for the industry. When we started this journey, my youngest wasn't even a year old. Now she's going into Kindergarten.

Once again, we need to give a HUGE thanks to Derek Milroy for stepping up and providing great content around Windows, vulnerability management, and being the sole technical editor for this book. You will find his influence in every chapter of this edition.

And finally, to you, the reader. Whether you are in internal audit, a QSA, or simply someone responsible for some portion of PCI DSS, you live in the trenches implementing solutions every day. The bad guys will never stop, so remember to build securely!

— Dr. Branden R. Williams

About PCI DSS and this book

1

INFORMATION IN THIS CHAPTER:

- Who should read this book?
- How to use the book in your daily job
- What this book is not
- Organization of the book
- Summary

The Payment Card Industry Data Security Standard (PCI DSS) celebrated its ninth year (December 15, 2004) and the PCI Security Standards Council its eighth birthday (September 7, 2006) as of this writing. Most of you reading these words have probably heard about PCI DSS, worked on a project tied to PCI DSS compliance, or said a few words out loud about PCI DSS that would have earned at least one of the authors a big smack across the face from his mother. For those of you just starting with PCI DSS, we authors hope this book can be your guide to a successful end result—a sustainable compliance program that exceeds the baseline security standards set forth in PCI DSS 3.0.

If you are like most professionals, the idea of becoming compliant with PCI DSS, or countless other regulations, does not sound fun. Information technologists and information security professionals aren't the only ones who share this feeling. Not only have C-Level individuals and other non-information technology (IT) (business) personnel had to deal with compliance and regulation around payments at some point in the last 8 years of their career, but we have even given rise to a new C-Suite position—the Chief Compliance Officer (CCO). While the CCO is not a new position with articles dating back to the mid-1970s referencing the moniker, the challenging landscape that companies must navigate necessitated more focus upon this function in the wake of Sarbanes–Oxley (SOX), PCI DSS, Health Insurance Portability and Accountability Act (HIPAA), and others.

Compliance efforts are rarely described as fun among those working with them. Painful is probably a better description. Whether it is the pain of not knowing what to do, pain of failing the assessment, or pain of “doing compliance” without an adequate budget, there are plenty of challenges that compliance—PCI DSS compliance in particular—have in common with pain.

Thus, we face the seemingly impossible challenge to write a fun and insightful book about PCI DSS. We realize the near impossible task ahead, and we are

committed to the challenge. We'd like to invite you, our reader, to travel with us in the hopes that when you turn the last page, you would come to realize that PCI DSS compliance can indeed be (YES) fun!

There are many standards and regulations out there. If your company's stock is publicly traded in the United States, you must adhere to the SOX mandates. Financial companies fall under the Gramm–Leach–Bliley Act. Those in the energy sector work toward North American Electric Reliability Corporation, Federal Energy Regulatory Commission, or Critical Infrastructure Protection standards. If you are in the health care industry, your network must comply with the HIPAA standards as updated recently in legislation focused on electronic health records. Other countries have their own "alphabet soup" of standards such as British Science Institute (BSI), Russian GOST (Russian for "gosudarstvenny standart" or "state standard"), worldwide International Organization for Standardization/International Electrotechnical Commission, and so on. PCI DSS occupies a special place among the standards for two reasons: broad, worldwide applicability, and the presence of enforcement mechanism that is seen as imminent and unavoidable, unlike for some other mentioned regulations.

The overarching theme of all these standards, laws, and regulations is that organizations need to secure data and protect their networks to keep citizens' data safe. In some cases, weak information security may only affect one company. However, when the data on the corporate network contains personal information about patients, customers, or employees, a breach of security can have implications far beyond the victimized company. A breach dealing with hundreds of millions of customers, such as a payment card processor, will have implications touching nearly every family; thus, decreasing such occurrences is in the public interest. Recent breaches have brought this concept back to the forefront as malware authors have advanced their capabilities and tenacity; thus, even subverting some of the very basic controls designed in many of these compliance initiatives.

Visa, MasterCard, American Express, Discover, and JCB developed PCI DSS together to ensure that credit card customer information and the associated payment systems are adequately protected from fraud. Breaches of customer information lead to financial loss and damaged reputations. The credit card industry wants to protect itself from financial loss or eroded consumer confidence in credit cards, which could lead to expensive and invasive governmental regulation.

We will use our experience with PCI DSS, both from the PCI Qualified Security Assessor (QSA) side and the information security side, to explain the most up-to-date PCI DSS guidelines to you (version 3.0 as of this writing). The objective of this book is not only to teach you about the PCI DSS requirements but to help you understand how the PCI DSS requirements fit into an organization's information security framework and how to effectively implement information security controls so that you can be both compliant and secure. In addition, we will cover ways to do this in the easiest and most pain-free way without compromising security in the process.

This book will make constant reference to the PCI DSS. PCI DSS, and its related standards, is owned by the PCI Security Standards Council, sometimes known in the industry as PCI Co. Before you start reading this book, you should go to the Council's

Web site at www.pcisecuritystandards.org and download PCI DSS version 3.0 and the Report on Compliance Reporting Instructions. You can find the relevant documents by clicking on “PCI Standards & Documents,” then “Documents Library.”

As of this publication, PCI DSS is at version 3.0. This book will highlight any significant changes between the previous version 2.0 and this version, and give you compliance tips as someone complying with the standard.

WHO SHOULD READ THIS BOOK?

Every company that accepts card payments, processes credit- or debit card transactions, stores payment card data, or in any other way touches personal or sensitive data associated with payment card processing is affected by the PCI DSS. Nowadays, it means that virtually all businesses, no matter how big or small, need to understand their scope of PCI DSS and how to implement PCI controls to reduce their compliance risk, or face penalties potentially to the point of losing their ability to cost-effectively and legally process payments.

Even with such a broad audience compelled to comply with PCI DSS, this book had to be written for a specific technical level. This book could have been written in very simple terms to educate the general population about PCI DSS. We could have written an in-depth technical tome providing every bit of detail a network engineer or security administrator might need to configure and implement all controls mandated by PCI DSS. This book aims in the middle and is more of a strategic guide to help management and practitioners understand the implications of PCI DSS and what it takes to be compliant. Ultimately, our goal in writing this book was to demystify some of the challenges with PCI DSS and allow readers to understand the right questions to ask of their peers to work toward compliance.

Overall, the book is useful for every stakeholder in an organization dealing with credit cards. This would include executive management, IT and IT security management, network, server, application developers, database managers, legal, marketing, sales, HR, front-line managers, and anyone interested in payment security.

Because of the wide impact that PCI DSS has on any organization, this book is like the small business with five employees—it can wear multiple hats and will appeal to multiple audiences. This book is for the IT managers and company managers who need to understand how PCI DSS applies to their organizations. This book is for the small- and medium-size businesses that don’t have an IT department to delegate to. This book is also for large organizations whose PCI DSS project scope is immense. It is for all organizations that need to grasp the concepts of PCI DSS and how to implement an effective security framework that is also compliant. This book is intended as an introduction to PCI DSS, but with a deeper and more technical understanding of how to put it into action. Finally, even PCI (and anti-PCI) “literati” will benefit from the stories and case studies presented by us!

HOW TO USE THE BOOK IN YOUR DAILY JOB

You can use the book during the entire lifecycle from complete PCI unawareness to ultimate security and compliance enlightenment. Specifically, you can use it as provided in the following:

- Learn what PCI DSS is and why it is here to stay,
 - Understand how it applies to you and your organization,
 - Learn what to do about each of the 12 main requirements,
 - Learn how to deal with PCI assessors and internal auditors,
 - Learn how to plan and manage your PCI DSS project,
 - Understand all the technologies referenced by PCI DSS,
 - Learn how to form strategies for removing portions (or indeed all) of your company from scope,
 - Get the best experience out of what can be seen as a painful assessment and remediation process.
-

WHAT THIS BOOK IS NOT

While reading the book, remember that this is not the book that will unambiguously answer every esoteric PCI DSS question. There is simply no way to create a book with every use case in it with the goal of answering PCI DSS questions as the regulation applies to your own environment. Indeed, there is similarity in how networks and systems are deployed, but given the broad applicability of PCI DSS—from small e-commerce sites to huge worldwide retailers—there is no way to have a book “customized” for your networks, systems, and applications. It is not meant to be the final authority for all issues related to PCI DSS, and it is not the unabridged guide to all things of PCI DSS. Finally, even though the book is written using one of the authors’ QSA¹ and consulting experiences, your Acquiring Bank is the ultimate judge of most PCI “puzzles” you will face on your journey to compliance and your QSA (or other similarly credentialed and experienced individual) should be your guide to lead you to top of PCI Compliance Mountain.

ORGANIZATION OF THE BOOK

Each chapter of the book is designed to provide you the information you need to know in a way that you can easily understand and apply. The chapters in this book follow a common structure which, wherever possible, includes the description of the PCI DSS requirement, the value of the requirement for PCI DSS and security, common tips and select tools useful for satisfying the requirement, as well as common mistakes and pitfalls.

¹The term QSA and the role of QSAs in PCI DSS assessments will be explained in Chapter 3.

In simple and direct terms, we will first explain the control or concept we are talking about in a way that illustrates its intent. Then, we explain where this concept sits in PCI DSS and why it is needed for information security, that is, how it reduces risk. Next, we explain what you should do with this concept to be secure and compliant using examples and common practices. Most chapters have detailed and entertaining case studies. When we said that we will make PCI DSS fun, we really mean it! Most chapters have a summary that provides a brief recap of the concepts discussed to reinforce what you read or to help you identify areas that you may need to re-read if you feel you don't understand them yet. Where possible, we also try to highlight common mistakes and pitfalls with these requirements or PCI concepts.

SUMMARY

This section provides a brief description of the information covered in each chapter:

- Chapter 1: About PCI and This Book—This chapter explains why PCI DSS is special and what this book is about.
- Chapter 2: Introduction to Fraud, Identity Theft, and Regulatory Mandates—This chapter explains cybercrime and regulations and is a brief look at payment card fraud, cybercrime, Identity theft, and other things around PCI DSS.
- Chapter 3: Why Is PCI Here?—This chapter gives an overview of PCI DSS and why the card industry was compelled to create it. This chapter also includes some discussion about the benefits of PCI DSS compliance and the risks of noncompliance.
- Chapter 4: Determining and Reducing Your PCI Scope—Every successful project around PCI DSS hinges on correctly scoping the environment. Expect that you should learn exactly how to scope your environment, learn ways to reduce it, and get tips for planning your PCI DSS projects.
- Chapter 5: Building and Maintaining a Secure Network—This chapter explains fundamental steps in protecting PCI DSS and other electronic data: making your network secure in the first place. This chapter discusses the basic components of a secure network and lays the foundation for building the rest of your PCI DSS compliance.
- Chapter 6: Strong Access Controls—This chapter covers one of the most important aspects of PCI DSS compliance: access control. The information in this chapter includes restricting access to only those individuals who need it, as well as restricting physical access to computer systems.
- Chapter 7: Protect Cardholder Data—This chapter explains how to protect the card data stored in your systems, as well as how to protect data while it is in transit on your network.
- Chapter 8: Using Wireless Networking—This chapter covers wireless security issues and wireless security controls and safeguards managed by PCI DSS. We include concepts that can be widely applied to Wi-Fi, Bluetooth, cellular, satellite, and emerging standards like Zigbee.

- Chapter 9: Vulnerability Management—This chapter explains performing vulnerability assessments to identify weaknesses in systems and applications, and how to mitigate or remediate the vulnerabilities to protect and secure your data.
- Chapter 10: Logging Events and Monitoring the Cardholder Data Environment—This chapter discusses how to configure logging and event data to capture the information you need to be able to show and maintain PCI compliance, as well as how to perform other security monitoring tasks.
- Chapter 11: Cloud and Virtualization—This chapter is a long time in the making, and we hope will serve as a fantastic guide to the rather challenging topic of leveraging these technologies in a PCI DSS environment.
- Chapter 12: Mobile—We are increasingly becoming reliant on mobile devices in our interactions with the world from our customers to our employees. You can safely use Mobile technologies, and we will discuss how.
- Chapter 13: PCI for the Small Business—PCI DSS isn't just for big box retailers and large banks. Whether you handle millions or hundreds of cards per year, you must comply with the DSS. This chapter includes tips on how to achieve PCI Compliance in a small business, subsidiary, or satellite office setting.
- Chapter 14: Managing a PCI DSS Project to Achieve Compliance—This chapter gives an overview of the steps involved and tasks necessary to implement a successful PCI compliance project. This chapter includes a discussion of the basic elements that should be included in future projects and to proactively ensure they are PCI compliant.
- Chapter 15: Don't Fear the Assessor—This chapter makes you understand that an assessor is there to work with you to validate your compliance and help you with security. They are only your enemy if you treat them this way. This chapter explains how to use the findings from a failed assessment to build ongoing compliance and security.
- Chapter 16: The Art of Compensating Control—This chapter explains how compensating controls are often talked about and misunderstood. This chapter will help build understanding and confidence in the reader when dealing with this tricky and often ambiguous component of PCI DSS, and most importantly, give you tips on creating your own controls.
- Chapter 17: You're Compliant, Now What?—This chapter covers the details you need to keep in mind once you have achieved compliance. Security is not as simple as just getting it implemented. You have to monitor and maintain it. This chapter contains information about ongoing training and periodic reviews, as well as how to conduct a self-assessment to ensure continued compliance.
- Chapter 18: Emerging Technologies and Alternative Payment Schemes—This chapter looks to the future of payments and how they will impact your PCI DSS strategies.
- Chapter 19: PCI DSS Myths and Misconceptions—This final chapter explains common but damaging PCI myths and misconceptions, as well as the reality behind them.

For those of you new to PCI DSS, we recommend going right through the chapters in order. They build upon themselves as concepts continue to get more complex and we apply what we learn. Once you are through the book, you will be able to reference specific content a little bit easier.

And with that, let's delve into fraud, identity theft, and regulatory mandates.

Introduction to fraud, data theft, and related regulatory mandates

2

Credit card fraud, identity theft, and broader personal data theft are problems that plague our information-dependent society and predate the age of the Internet. Ironically, things such as automated processing of financial data that make your life easier and more convenient also make crime easier and more convenient. Moreover, the Internet allowed crime that only happened on a small scale to grow and spread globally, and the Internet's scalability turned electronic-based crimes into a global concern.

Some crime was automated and changed from rare to widespread, for example, Nigerian e-mail or UK Lottery scams. Gone are the days where criminals need to be in the same location, country, or even continent to scam you out of your hard-earned cash. Nigerian e-mail scams started many years ago and are profitable for the scammers. They send out millions of e-mails claiming to be a relative of a Nigerian dignitary with frozen assets and want you to transfer the money for them. You give them your bank account information and/or send them “seed money” to get things moving and end up with nothing. UK Lottery scams aren’t much different with the same basic constructs to get you a cash prize.

Criminals have gone high-tech and have discovered that there is a significant amount of money to be made with very little risk. Hacking a company database or orchestrating a phishing attack while sitting in your pajamas and eating Extreme Doritos in the living room of your house has much more appeal than physically robbing banks or convenience stores. The advancement of automated exploit kits such as Metasploit has made couch-hacking more effective for even the slightly knowledgeable. Add to that the lower risk of a confrontation with firearms and electronic crime becomes even more attractive! Depending on the company being targeted, the sophistication of the attack, and sheer luck, sometimes the high-tech crime may also be significantly more lucrative than traditional armed robbery. Sadly, cross-border prosecution issues significantly fuel a cybercriminal’s activity. When a criminal physically robs a convenience store, he is probably caught on tape and there are witnesses. In addition, law enforcement will mobilize quickly to find and catch the criminal so he may be brought to justice. Cybercriminals have a couple of things working in their favor, the first of which is their ability to commit crime without ever stepping into the physical location of their victim(s). Couple that with lagging cybersecurity laws in most countries and the limited ability for the victim’s law enforcement bodies to prosecute outside their borders and you have an idea on why cybercrime is on the rise. In addition, the whole ecosystem of criminal outsourcing

now allows other criminals to only focus on the activities they do best, such as creating malicious software or conducting crime through botnets.

Malicious software (malware) and cybercriminals are not the only threat. Sadly, the very companies and organizations that are entrusted with sensitive information are often to blame because of a lack of adequate controls to protect sensitive information. In some companies information security is treated with apathy; in others, a lack of effective controls enables an insider to commit fraud. Consumers and businesses are faced with a wide variety of threats to their data and personal information on any given day.

Spyware, phishing attacks, drive-by downloads, and botnets are all computer attacks that are on the rise and pose a significant threat to corporate and home users as they connect to the Internet from their computers. However, those threats pale in comparison with the amount of personally identifiable information and sensitive data available to be compromised due to carelessness or negligence by individuals and corporations.

TOOLS

Did you know that the Privacy Rights Clearinghouse has tracked all reported breaches since the ChoicePoint breach on February 15, 2005 (as well as including additional breaches disclosed prior)? To see all these breaches with an explanation and amount of records lost, point your browser at www.privacyrights.org/data-breach.

DataLossDB at <http://datalossdb.org/> is another useful site for tracking the impact of data breaches. Despite its name, most of the recorded and analyzed data “loss” incidents are really data theft and abuse incidents. DataLossDB crew does an awesome job of tracking all publicly reported incidents and digs out the details on them.

As of today, over a 500 million various personal information records have been lost or stolen. Every year since the ChoicePoint breach, we've seen major companies fall victim to Payment Card Industry (PCI)-related security breaches. DSW Retail in 2005, The U.S. Department of Veteran's Affairs in 2006 (and in later years), The TJX Companies in 2007, Hannaford Brothers in 2008, Heartland Payment Systems in 2009, Albrecht Discount in 2010, Sony in 2011, KT Corporation in 2012, and the various retailers in 2013–2014 that have reported breaches including Target and Neiman Marcus continue to demonstrate both the poor state of security and increasing sophistication and numbers of the bad guys (as more and more countries have growing populations on the Internet) who want this data and know how to profit from it.

In an “Information is King” era, when more consumers are using computers and the Internet to conduct business and make purchases, taking the proper steps to secure and protect personally identifiable information and other sensitive data has never been more important. It is bad for companies, individuals, and the economy at large if consumer confidence is eroded by having personal information exposed or compromised. Credit card brands are definitely not the only entities suffering from such possible loss of confidence.

NOTE

Take a step back from the text for a minute and adjust your mindset to think of yourself as a general consumer, Internet user, or citizen—not as a security or payment professional. What data do you hold dear? Think through the following list of scenarios:

What data or information about me can be considered sensitive and should not be disclosed, be corrupted, or be made permanently or temporarily unavailable? Think of a broad range of types of information—from a rare photo to your bank account number, medical history, or information about anything you've done that you are not proud of.

Think whether this information exists in any electronic form, on your computers or anywhere else? Is that picture on your “private” Facebook page—an oxymoron if there ever was one—or present in an e-mail spool somewhere?

Next, think whether this information exists on some system connected to the Internet (possibly indexed by a helpful engine). Sadly, the answer today would be “yes” for almost all (!!!) information people consider sensitive. For example:

Credit card information—check,
Bank account information—check,
Personal financial records—check,
Tax records—check,
Legal proceedings—check,
Sensitive personal files—check,
Health records—check.

Think what will happen if this information is seen, modified, or deleted by other people. Will it be an annoyance, a real problem, or a disaster for you? What if it's just on a decommissioned hard drive that fell off the back of a truck?

Now, think about what protects that information from harm. Admittedly, in many cases, you don't know for sure. We can assure you that sometimes your assumption that the information is secure will be just that—an assumption—with no basis.

Going through this list helps you not only understand data security rationally but also feel it in your “gut.”

Information technologists are affected by a number of laws and regulations designed to coax businesses into addressing their security problems. Depending on what industry a company serves, they may fall under Sarbanes–Oxley (SOX), the Gramm–Leach–Bliley Act of 1999, the Health Insurance Portability and Accountability Act (HIPAA), Federal Information Security Management Act, and other regulatory mandates that we mentioned in the very beginning of Chapter 1. Maybe this confusing hodgepodge of alphabet soup—and that is without European and other regional mandates and regulations—makes for a tough job understanding how to comply with all these measures, as many organizations still fail to enforce adequate security. The Unified Compliance Framework that can be found at www.unified-compliance.com tracks hundreds of IT-relevant regulations, and many commercially available e-Governance Risk and Compliance (eGRC) tools such as RSA's Archer or IBM's OpenPages can help build, manage, and reference a common control set to cover all of these compliance initiatives.

NOTE

If you feel lost and out of control, don't. Remember, all these crazy compliance initiatives are trying to minimize the risk associated with an underlying problem—poor security. Taking a step back and looking at a standard security framework, like ISO27002, would do more to boost your global compliance efforts than attacking any one of these by themselves. A mature ISO27002 program would be able to adapt to future compliance initiatives or changes in a way that would minimize the overall impact compliance has on your organization.

Breaches often target consumer credit card information because of the revenue this type of data can generate on the black market. Since our last publication, the value of magnetic stripe data on the black market continues to decline as big breaches flood the market, but that doesn't stop the attacks or the desire to capture other data like Personally Identifiable Information (PII) and Personal Identification Number (PIN) information. Card companies recognized the rising threat to their brands and the large payment systems they invested in, and eventually they came together to develop the PCI Data Security Standards (DSS). In essence, the credit card industry has taken steps to assure the security of credit card data and transactions and maintains the public trust in credit cards as a primary means of transacting business. If you want to accept credit cards as payment or take part in any step of the processing of the credit card transaction, you must comply with the PCI DSS. Failure to do so can result in penalties stiff enough to cause public disclosures, or worse, bankruptcy.

NOTE

Most of the above regulations focus on the issues of data protection from theft or confidentiality of sensitive data. When we think about fraud and the abuse of somebody's identity, we think about people stealing data as if it were a thing to stash in your pocket. Indeed, to assume an identity and apply for credit under that name, a thief needs that identity's most sensitive personal information. In the United States, the typical combination needed for identity (ID) theft ("ID theft bundle") is as follows:

- Social security number (SSN),
- Your mother's maiden name,
- Your full name,
- Your current and past addresses and phone numbers,
- Your employer name and address.

From this pack, only the first two are not truly public (even though the secrecy of the latter is at best debatable and the predictability of the assignment of SSNs in conjunction with the multiple methods to obtain this information runs rampant) and require some work to obtain. The rest of the bundle can be assembled later after the most sensitive information is in the possession of the attacker.

However, think what happens after your identity has been stolen and assumed by the attacker who now lives "your" life and applies for credit cards, loans, and bank accounts using your name.

He now modifies or corrupts your data by harming your stellar credit score, reputation, standing with financial institutions, employers, government agencies (e.g., if he commits crime and then shows fake ID—or, worse, illegally obtained "real" ID—with your name).

Thus, remember that ID theft is not just about information theft; the damage comes from actual changes to your critical information!

And while the attacker (excluding the most "special" cases that we are not prepared to discuss here...) cannot "erase" your life from the systems, the damage done to your future life can be significant, especially if the case of ID theft is detected late in the game.

Unlike SOX or HIPAA, the PCI DSS is not a law; however, in many ways, it is more effective. Noncompliance probably won't land you, the merchant, in jail, but on the rare and extreme side, it could mean having your merchant status revoked (or changed such that processing payments becomes illegal or cost prohibitive). For some organizations, losing the ability to process credit card payments would drastically affect their ability to do business and possibly even bring about the death of the company. Although PCI DSS can be effective in stopping security breaches, companies still seem to struggle with its implementation.

WARNING

Although PCI DSS itself is not a law, both Nevada and Minnesota have enacted laws requiring that companies serving their residents comply with PCI DSS.

NOTE

By the way, credit card theft and identity theft are not the same. In fact, they have literally nothing to do with each other, despite what you hear from misinformed journalists.

To explain it further, you might not care much if your credit card information is stolen due to legally mandated card liability limits (that are typically reduced even further—to \$0), but you must and, in fact, will be made to care if your identity is stolen and then used by the criminals.

There is nothing extraordinary or magical about the PCI DSS requirements—with the exception of the interpretation. The guidelines spelled out are all, essentially, common security practices that any organization should follow without being told. Companies with mature information security programs have had few problems adding unique PCI DSS requirements to their programs, even when some had trouble proving that their controls are as good (or better) than PCI mandated controls. Even so, some of the requirements leave room for interpretation and complying with PCI DSS can be tricky.

Here's a hint: if one particular requirement for PCI DSS seems too hard to comply with, you might be approaching it all wrong. Think less about how to get out of complying, and think more about how to incorporate and build upon the baseline of security provided by PCI DSS. Or even better, think about how to remove your compliance burden all together by outsourcing it to a third party.

As with any information security regulation or guideline, you need to keep your eye on the ultimate goal. When executing a compliance program, some organizations follow the letter rather than the spirit or intent of the requirements. The end result may be that they were able to check off all the compliance boxes, declaring their network compliant, but not really be secure. Remember, if you follow the requirements and seek to make your network as secure as possible, you are almost guaranteed to be compliant. But, if you gloss over the requirements and seek to only make your network compliant, there is a fair chance that your network could still be insecure. It could even happen while your assessors are on site!

Major retailers and larger enterprises are well aware of the PCI DSS—and have been aware of it for years. They have dedicated teams that focus on security and on PCI DSS compliance. They have the resources and the budget to bring in third parties to assess and remediate issues. The scope of PCI DSS affects almost every business, from the largest retail megastores down to a self-employed single mother working from her home computer. If the business accepts, processes, transmits, or in any other way handles credit card transactions, they must comply with PCI DSS.

SUMMARY

The purpose of this book is to provide an overview of the components that make up the PCI DSS and to provide you with the information you need to know in order to get your network PCI DSS compliant and keep it that way. We've discussed how larger Compliance-Driven Alphabet Soup Initiatives can really confuse the business side of operations. Security is a business issue, and a good security program puts a framework in place to address issues like compliance before they become a problem.

Each major area of security covered by the PCI DSS is discussed in some detail along with the steps you can take to implement the security measures on your network to protect your data. Anton and Branden, your humble authors for the next 15 chapters, are established information security professionals. We've been there and done that, and we have acquired wisdom through trial and error. We hope our experience will help you implement effective solutions that are both secure and compliant.

Why is PCI here?

3

INFORMATION IN THIS CHAPTER:

- What is PCI and who must comply?
- PCI DSS in depth
- Quick overview of PCI requirements
- PCI DSS and risk
- Benefits of compliance
- Case study

Chances are if you picked up this book, you already know something about the Payment Card Industry Data Security Standard (PCI DSS); however, you might not have a full and clear picture of PCI DSS—both the standards and its regulatory regime—and why they are here. This chapter covers everything from the conception of the cardholder protection programs by the individual card brands to the founding of the PCI Security Standards Council (PCI SSC) and PCI DSS development. It also explains the reasons for PCI DSS's arrival that are critical in understanding how to implement PCI DSS controls in your organization. Many of the questions people ask about PCI DSS and many of the misconceptions and myths about PCI have their origins in the history of the program, so it only makes sense that we start at the beginning.

WHAT IS PCI DSS AND WHO MUST COMPLY?

First, “PCI” is not a government regulation or a law.¹ As you know, when people say “PCI,” they are actually referring to the PCI DSS Version 3.0 (at the time of this writing). However, to make things easy, we will continue to use the term *PCI* to identify the payment industry standard for card data security interchangeably with PCI DSS.

Unlike many other regulations, PCI DSS has a very simple and direct answer to the question “Who must comply?” Despite its apparent simplicity, many misunderstand

¹PCI DSS or the elements of it have been adopted as actual law in at least two US states at the time of this writing. The State of Nevada explicitly called PCI DSS by reference and made it mandatory for some businesses operating in this state. The States of Minnesota and Massachusetts have adopted language from PCI DSS into their own information security statutes.

the question to the point that they incorrectly name specific players as “in” or “out,” which leads the authors to believe that many of such people have their own agenda. This always reminds us of a quote from Upton Sinclair, a noted American novelist, who said “It is difficult to get a man to understand something when his job depends on not understanding it” [1]. So, PCI’s answer to “who must comply?” is any organization that accepts payment cards or stores, processes, or transmits credit or debit card data must comply with the PCI DSS.

NOTE

PCI DSS applies to you if your organization accepts, processes, stores, and/or transmits member-branded card data. Member-branded card data is any card that is part of the Visa, MasterCard, American Express, Discover, and JCB payment schemes, including their subsidiaries or international partners. Should a new member be added to this list, their cards would also be included in the scope of PCI DSS compliance (rumors are running rampant that China Union Pay and PayPal may join). Because of so-called “check” cards, you can expect that nearly every debit card will fall into the PCI DSS scope simply because they can be used as either a debit or member-branded credit card.

It is very easy to understand the motivations for such broad applicability. It is pointless to protect card data only in a few select places; it needs to happen wherever and whenever said card data is physically and electronically present. You might be thinking, “why is the data present in so many places?” A recent MasterCard presentation at a payment security conference presented a curious statistic that there are more than 200,000 locations where payment card data is stored in large amounts. Visa believes that they work with over 32,000,000 acceptance locations, worldwide! Each of those could potentially be storing months or years of payment card data in places where criminals can steal it. Keep those statistics in mind as you read through the book to provide context on both the macro- and microscales. Without jumping too far ahead into our story, we’d say that in many cases, adjusting your business processes to not touch the card data directly will save you from a lot of security and compliance (and not just PCI DSS compliance!) challenges!

In this book, we are primarily concerned with merchants and service providers. Merchants are pretty easy to identify—they are the companies that accept credit cards in exchange for goods or services. The PCI official definition of a merchant [2] states: “a merchant is defined as any entity that accepts payment cards bearing the logos of any of the five members of PCI SSC (American Express, Discover, JCB, MasterCard, or Visa) as payment for goods and/or services.” For example, a retail store that sells groceries for cash or credit cards is a merchant. An e-commerce site that sells electronic books is also a merchant.

However, when it comes to service providers, things get a bit trickier. The PCI Council Glossary [3] defines them as: “[a] business entity that is not a payment brand [but] directly involved in the processing, storage, or transmission of cardholder data. This also includes companies that provide services that control or could impact the

security of cardholder data. Examples include managed service providers that provide managed firewalls, Intrusion Detection System (IDS) and other services as well as hosting providers and other entities. Entities such as telecommunications companies that only provide communication links without access to the application layer of the communication link are excluded.” This definition is clunky and verbose. A better way to express service providers would be any entity that can affect the security of payment card information (excluding the same companies as the above definition does). If you have a provider that does something that can impact the security of cardholder data, they are a service provider and should be validated as compliant with PCI DSS.

Sometimes a merchant can also be a service provider at the same time: “...a merchant that accepts payment cards as payment for goods and/or services can also be a service provider, if the services sold result in storing, processing, or transmitting cardholder data on behalf of other merchants or service providers” [2]. As an example, a merchant could stand up a business model whereby a company accepts credit cards as a payment for services it provides to other merchants who also accept credit cards. In this case, such an entity is both a merchant and a service provider. For example, if you provide hosted shopping cart and processing services to merchants and accept payment cards, you would be both.

Now that we have some baseline definitions described, we will describe the whole payment ecosystem for the purposes of PCI DSS.

ELECTRONIC CARD PAYMENT ECOSYSTEM

Before we go into detail on PCI compliance, we’d like to paint a quick picture of an entire payment card “ecosystem” ([Figure 3.1](#)).

[Figure 3.1](#) shows all the entities in payment card “game”:

- Cardholder, a person holding a credit or debit card.
- Merchant, who sells goods and services and accepts cards.
- Service provider (sometimes Merchant Service Provider [MSP] or Independent Sales Organization [ISO], who provides all or some of the payment services for the merchant.
- Payment processor, which is a particular example of an MSP.
- Acquiring bank, which connects to a card brand network for payment processing and also has a contract for payment services with a merchant.
- Issuing bank, which issues payment cards to consumers (who then become “cardholders”).
- Card brand (also known as a payment brand or card scheme depending on regionalization), which is a particular payment “ecosystem” (called “association network”) with its own processors, acquirers, and for the purposes of PCI DSS includes the member brands (Visa, MasterCard, American Express, Discover, and JCB).

The primary focus of PCI DSS requirements is on merchants and service providers. This is understandable since this is exactly where most of the data is lost

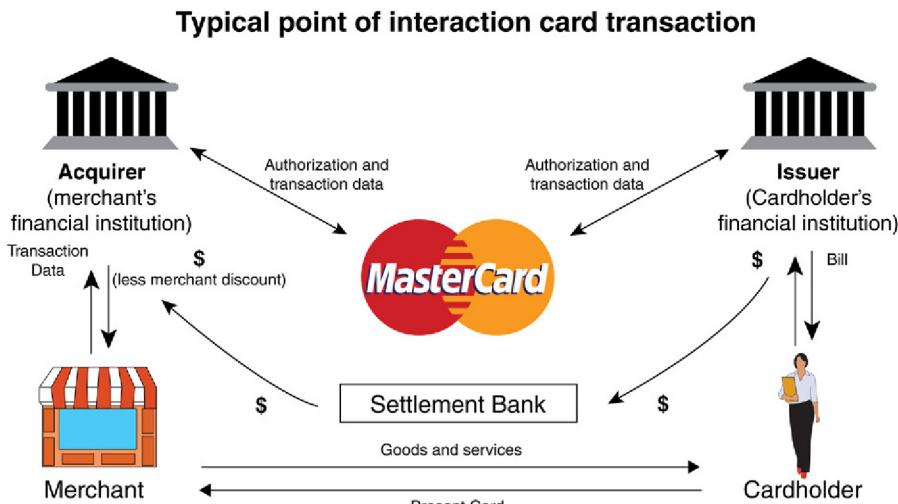


FIGURE 3.1 PCI Payment Ecosystem

to malicious hackers. Whether TJX in 2005–2007 (45 or 90 million cards stolen, depending on the source), Heartland Payment Systems in 2008–2009 (more than 100 million reported cards stolen), or Target in 2013 (more than 40 million cards), merchants and service providers have had cards stolen from them and paying fines to go toward reissuance. Prior to some of the regulations in PCI DSS becoming mainstream, issuing banks were replacing compromised cards at their own cost and incurring other administrative and fraud costs as well. Thus, PCI DSS was born to restore the balance to the system by making sure that merchants and service providers took care of protecting the card data. The motivation for merchants to comply with PCI DSS comes in the form of fines, higher processing costs, and litigation risk.

Goal of PCI DSS

In light of what is mentioned above, PCI DSS is here to reduce the fraud risk of payment card transactions by motivating merchants and service providers to protect card data. Whether this goal is worthy, whether there are other secondary goals, or even whether this goal is being achieved by the current version of the data security standard is irrelevant. What matters to us is that PCI DSS is aimed at reducing the fraud risk of transactions. It seeks to accomplish that by forcing merchants and service providers to pay attention to many key aspects of data security including network security, system security, application security, security awareness, incident response, and policies. Even more importantly, it indirectly encourages merchants to drop cardholder data entirely and conduct their business in a way that eliminates

costly and risky data storage and on-site processing. The focus on security practices and technologies naturally begets a reduction of fraud. One of the original PCI DSS framers also described it as the following: “the original intent was to design, implement, and manage a comprehensive, cost effective and reliable security effort” [4] and not a patchwork of security controls.

Interestingly enough, the “Ten Common Myths of PCI DSS” document from the PCI Council presents the six domains of PCI DSS as its goals [5]:

1. Build and maintain a secure network,
2. Protect cardholder data,
3. Maintain a vulnerability management program,
4. Implement strong access control measures,
5. Regularly monitor and test networks,
6. Maintain an information security policy.

While the above six domains can be seen as tactical goals during a PCI DSS implementation, the strategic focus of PCI DSS is card data security, payment card risk reduction, and ultimately the reduction of fraud losses for merchants, banks, and card brands.

Overall, while motivating security improvements and reducing the risk of card fraud, PCI DSS serves an even higher goal of boosting consumer confidence in what is currently the predominant cashless payment system—plastic cards. While we can debate whether paper, plastic, and metal money is truly on the way out, the volume of cashless transactions is increasing annually though the percentage numbers will vary depending on how you slice the research. Some countries like Nigeria are attempting to move to entirely cashless payment systems (see <http://www.cenbank.org/cashless/> for info). If anything—whether malicious hackers, insiders, or any other threat—can hinder it, our global economy will suffer losses. Thus, PCI DSS defends something even bigger than “bits and bytes” in computer systems—primarily attempting to protect a major money-exchanging cog in the economic system itself.

Applicability of PCI DSS

Although the statements about accepting, processing, storing, and transmitting payment card data will probably sound tiresome by the time you are finished reading our book, remember that PCI DSS applies to all organizations that perform the above and there are no exceptions. Our Chapter 19, covers some of the common, industry-wide delusions and clarifies that the above PCI applicability is indeed the reality and not the myth.

The question of validating or proving PCI compliance is a bit different from the argument of PCI DSS applicability to organizations that deal with card data. The type of validation and requirements you must follow can differ for merchants and service providers, and by card brand and transaction volume.

First, there are different levels of merchants and service providers. **Tables 3.1 and 3.2** show the breakdown.

Table 3.1 Merchant Levels

Merchant Level	Description
Level 1	Any merchant that has suffered a hack or an attack that resulted in an account data compromise (can vary based on payment brand), or any merchant deemed Level 1 by any payment brand
	Any merchant that processes more than 6 million Visa, MasterCard, or Discover transactions annually
	2.5 million American Express Card transactions or more per year, or any merchant that has had a data incident; or any merchant that American Express otherwise deems a level
	Merchants processing over 1 million JCB transactions annually, or compromised merchants (as RECOMMENDED), however, JCB doesn't have firm levels anymore. This is an approximation of level based on requirements from other payment brands
Level 2	Any merchant that processes between 1 and 6 million Visa or Discover transactions annually
	Any merchant with greater than 1 million but less than or equal to 6 million total combined MasterCard and Maestro transactions annually
	Any merchant that processes between 50,000 and 2.5 million American Express transactions annually
Level 3	Merchants processing less than 1 million JCB transactions annually
	Any merchant that processes between 20,000 and 1 million Visa or Discover card not present (e-commerce) transactions annually
	Any merchant with greater than 20,000 combined MasterCard and Maestro e-commerce transactions annually but less than or equal to 1 million total combined MasterCard and Maestro e-commerce transactions annually
Level 4	Any merchant that processes less than 50,000 American Express transactions annually
	All other Visa, MasterCard, and Discover merchants

NOTE

Some Visa levels may vary, and it is always up to an acquiring institution or payment brand to make adjustments to your level. For example, Visa Europe is a separate organization that has different rules, especially as it relates to compliance around their Technology Innovation Program (TIP) and Chip & Personal Identification Number (PIN) (EMV) transactions. For more specific information, contact your acquiring bank to provide level and validation guidance.

As we mentioned above, these levels exist for determining the type of compliance validation required as discussed in the next section. The levels are also sometimes used by the payment brands to determine which fines to impose upon the merchant for noncompliance.

Table 3.2 Service Provider Levels

Level	MasterCard	American Express	Visa Inc
Level 1	All third-party providers (TPPs), all data storage entities (DSEs) that store, transmit, or process greater than 300,000 total combined MasterCard and Maestro transactions annually	2.5 million American Express Card transactions or more per year; or any Service Provider that American Express otherwise deems a Level 1 service providers	VisaNet processors or any service provider that stores, processes, or transmits over 300,000 transactions per year
Level 2	Includes all DSEs that store, transmit, or process less than 300,000 total combined MasterCard and Maestro transactions annually	50,000–2.5 million American Express Card transactions per year	Any service provider that stores, processes, or transmits less than 300,000 transactions per year
Level 3		Less than 50,000 American Express Card transactions per year	

PCI DSS IN DEPTH

In the next section, we take a detailed look at the PCI DSS standard, its entire regulatory regime, as well as related security vendor certification programs.

COMPLIANCE DEADLINES

Now that we touched upon the compliance basics, it is time to face the painful fact: for the most part, all the PCI DSS compliance deadlines are *in the past* (Table 3.3). This means that yesterday was the time to be compliant. There are additional dates for various other related requirements (such as card brand dates for Payment Application Data Security Standard [PA-DSS] compliance with specific programs like Visa's TIP), but all core PCI DSS compliance dates have indeed passed. In some

Table 3.3 Original Compliance Dates for Merchants: All Passed

Level	American Express	MasterCard	Visa Inc
Level 1	October 31, 2006	June 30, 2005 or December 31, 2010 for merchants that are self-certified previously	June 30, 2004
Level 2	March 31, 2007	June 30, 2004	June 30, 2007
Level 3	N/A	June 30, 2005	June 30, 2005
Level 4	N/A	N/A	N/A

cases, you will find some dates that are in the future, but those are in the future typically because the criteria changed, thus merchants have a period of time to adjust their operations accordingly.

Some of you may recall receiving a letter that had a target compliance date. Such letters typically happen when your acquirer is going to track your specific compliance, or when you change levels or processors. These dates may or may not be aligned with the card brands' global published dates. This is because the card brands may not have a direct relationship with you and are working through your acquiring bank that sponsors you into the payment network.

TIP

When in doubt, always follow the guidance of the legal professionals responsible for reviewing and advising on your contracts.

Various predecessor versions of the PCI 3.0 standard had unique dates associated with them, so if your compliance efforts have not been aligned to the card brand programs, you are way behind the curve and will not likely get any sympathy from your bank.

As far as additional dates by card brands, please refer to the following resources²:

- *Visa:* <https://usa.visa.com/merchants/protect-your-business/cisp/key-dates.jsp>. This page includes dates such as “US Level 1 Merchants Full PCI DSS Compliance Validation Deadline” (September 30, 2010) and “US Level 2 Merchants Full PCI DSS Compliance Validation Deadline” (December 12, 2010). You will also see future dates around liability shifts that may serve as motivations to roll out specific technologies, further impacting your PCI DSS compliance programs.
- *MasterCard:* www.mastercard.com/us/company/en/whatwedo/determine_merchant.html. This page includes program information on how to validate, and shows that all dates passed in 2012 at the latest. Note, all Level 1 and Level 2 merchants must validate through an Internal Security Assessor (ISA) or Qualified Security Assessor (QSA).
- *Discover:* www.discovernetwork.com/merchants/data-security/disc.html. This page contains no additional deadlines and simply refers to the PCI Council site. Merchant levels and reporting criteria are listed on subsequent pages after clicking links on the left.
- *American Express:* [https://www209.americanexpress.com/merchant/services/en_US/data-security](http://www209.americanexpress.com/merchant/services/en_US/data-security). This page does not contain any additional deadlines, but validation requirements and merchant levels are different than other brands.

²Please note that these URLs change frequently. Please go to picompliancebook.info for more up-to-date links if these do not work.

- **JCB:** <http://partner.jcbcard.com/security/jcbprogram/index.html>. Merchant levels here are not labeled in levels, but split into two thresholds and two different classes based on the types of payments you accept.

COMPLIANCE AND VALIDATION

As we mentioned before, depending on your company's merchant or service provider level, you will need to go through an annual on-site PCI assessment by either a QSA or ISA, or complete a Self-Assessment Questionnaire (SAQ) to validate compliance. In addition to this, you may have to present the results of the quarterly network perimeter scans to be performed by an Approved Scanning Vendor (ASV).

If you are filling out an SAQ or doing other self-assessing under the ISA program, keep in mind that third parties may question your documentation a bit more simply because there is no third-party validation. We recommend that you use a combination of ISA and QSA (where an on-site assessment is required) resources to perform your annual assessment to be sure you get a thorough and fair result.

When submitting an SAQ, it will have to be physically signed by an officer of your company.³ At the present time, there is no court precedent for officer liability as a result of false PCI DSS compliance attestation. However, industry speculation is that this person may be held accountable in a civil court, especially if he or she intentionally misrepresents information while certifying. Translation: don't do anything unethical or illegal.

If you are planning on submitting a Report on Compliance (ROC) instead of the SAQ, you will need to follow the document template outlined in the PCI DSS ROC Reporting Instructions document. This document is intended to be used by QSAs and ISAs to promote a complete and accurate report coming from a PCI assessment. After the SAQ has been filled out or the ROC has been completed, it must be sent along with all the necessary evidence and validation documentation to the acquiring organization or processor. It depends on who requested the compliance validation in the first place.

It is a common misconception that the degree to which you must comply with PCI DSS varies among the different levels. Both merchants and service providers must comply with the entire DSS, regardless of their level and validation requirements. What varies is the way and frequency you report compliance upstream. If you determine you are a Level 4 Merchant, don't interpret the "recommended" under validation requirements to mean PCI DSS compliance is optional. Furthermore, don't download the PCI DSS Prioritized Approach from the PCI Council's Web site and decide that completing tasks through milestone three should be fine. Visa's Web site explains it like this: "In addition to adhering to the PCI DSS, compliance validation is required for all service providers" [6].

³Electronic attestation of a full digital copy has also been considered acceptable.

NOTE

Discover and JCB handle merchant PCI compliance validation differently. Contact your acquirer for more information.

The validation mechanisms, as of the time of this writing, are given in [Table 3.4](#).

Further, the scope of PCI DSS validation differs based on the exact way the organization interfaces with card data. Specifically, quoting from the PCI Council Web site, the circumstances that affect what sections of the SAQ the merchant should complete for validation are provided in [Table 3.5](#).

So, to summarize, the exact scope of any one's PCI DSS validation depends on the following:

- Merchant or service provider status,
- Transaction volume,
- Card brand,
- The method of accepting cards and interacting with card data.

NOTE

Although American Express, Discover, and Visa allow Level 1 merchants to have their PCI compliance validated by the merchant's internal audit group, MasterCard does not explicitly allow this. To qualify for internal validation under MasterCard's new rules, the professional performing the assessment must be a current ISA. Check the PCI Council's Web site for more information on this program, including the steps required to join.

Table 3.4 PCI DSS Validation Requirements

Merchant or Service Provider Level	Visa USA		MasterCard	
Level 1	ASV scan	QSA on-site assessment	ASV scan	QSA on-site assessment
Level 2	ASV scan	SAQ self-assessment	ASV scan	QSA/ISA on-site assessment or assisted SAQ completion
Level 3	ASV scan	SAQ self-assessment	ASV scan	SAQ self-assessment
Level 4	ASV scan if requested by the acquirer	SAQ self-assessment	ASV scan if requested by the acquirer	SAQ self-assessment

Table 3.5 SAQ Validation Types Based on Card Acceptance Methods

Card Processing	Self-Assessment Validation
Card-not-present (e-commerce or mail/telephone-order) merchants, all cardholder data functions outsourced. This would never apply to face-to-face merchants	SAQ type A, which is the smallest. It only includes parts of two out of 12 requirements, 14 questions
E-commerce only merchants that partially outsource their payment processing to PCI DSS validated entities and do not electronically process, store, or transmit any cardholder data	SAQ type A-EP that covers a subset of all 12 requirements
Imprint-only merchants with no electronic cardholder data storage, or standalone terminal merchants with no electronic cardholder data storage	SAQ type B (for dial-only terminals) or SAQ type B-IP (for IP-enabled terminals), which covers requirements most commonly related to those terminals
Merchants using only Web-based virtual terminals, no electronic cardholder data storage	SAQ type C-VT, which covers sections of nine out of 12 requirements, omitting unique IDs (Req 8) logging (Req 10), and regular testing (Req 11)
Merchants with payment application systems connected to the Internet, no electronic cardholder data storage	SAQ type C, which covers parts of all 12 requirements; and as of 3.0 it now does include logging requirements (Req 10)
Merchants who process payments through hardware payment terminals included in the P2PE Validated list on the Council's Web site	SAQ P2PE-HW, which includes a subset of requirements from 3, 4, 9, and 12
All other merchants not included in descriptions for SAQ types A through C above, and all service providers defined by a payment brand as eligible to complete an SAQ	SAQ type D (either the Merchant or Service Provider version), which includes all the 12 requirements and a full set of questions

WARNING

Don't let yourself become complacent. If you are a Level 4 merchant and are not required to do anything to validate your compliance with PCI DSS—remember, by accepting even one card per year you must comply with PCI DSS. Many Level 4 merchants end up in big trouble when they realize they had to comply with PCI DSS regardless of their validation requirements. In addition, due to different validation levels across major card brands, their situation in regards to PCI compliance may be much worse. Don't let a breach put you out of business from fines and fees. Ensure you are complying with PCI DSS at all times for all levels.

The specific validation requirements can change. For example, MasterCard announced in 2012 that Level 2 merchants now need to be validated via an on-site assessment from a QSA or have the person filling out the SAQ be a current ISA. Expect validation requirements to become stricter in the future from all payment brands.

Interestingly enough, companies can be merchants and service providers at the same time. If this is the case, the business should be described in detail in the assessment documentation and the compliance validated at the most stringent level. In other words, if a company is a Level 3 Visa merchant and a Level 1 Visa service provider, the compliance verification activities should adhere to the requirements for a Level 1 Visa service provider.

One notable PCI assessor, the late Walt Conway, related the following educational story about merchant and provider validation:

“My favorite is when the vendor replies that they are compliant as a Level 3 (or 2 or whatever) merchant. That response is completely irrelevant and inexcusably misleading. That they are compliant as a merchant is meaningless to you when you use them as a service provider. They can self-assess as a merchant—they cannot do as a Level 1 service provider. That extra step is meant to protect you. If you get that kind of reply, you are likely dealing with an over-eager and/or ill-informed sales rep...ask to talk to an adult.” [7]

HISTORY OF PCI DSS

To better understand the PCI DSS role, motivation, and future, let's review its origins and history.

PCI DSS evolved from the efforts of payment brands battling fraud and counterfeiting. In the 1990s, the payment brands developed various standards to improve the security of sensitive information. In the case of Visa (formerly a regional association model), different regions came up with different standards because European countries and Canada were subject to different standards than the United States. In June 2001, Visa launched the Cardholder Information Security Program (CISP, mostly pronounced KISP). The CISP Security Audit Procedures document version 1.0 was the granddaddy of PCI DSS. These audit procedures went through several iterations and made it to version 2.4 in mid-2004. At this time, Visa was already collaborating with MasterCard on creating a single mechanism for merchants to go through. Their agreement was that merchants and service providers would undergo annual compliance validation according to Visa's CISP Security Audit Procedures and would follow MasterCard's rules for vulnerability scanning. Visa maintained the list of approved assessors and MasterCard maintained the list of ASVs.

This collaborative relationship had a number of problems. The lists of approved vendors were not well-maintained, and there was no clear way for security vendors to get added to the list for each particular card brand. To complicate things further, every card brand division did not endorse the program. Other brands such as Discover, American Express, and JCB were running their own programs as well, further clouding the compliance requirements and process. The merchants and service providers in many cases had to undergo several independent assessments by different “certified” assessors just to prove compliance to each brand, which was cost too much and yielded a low-quality result. For that and many other reasons, the five major payment brands came together and created the PCI DSS 1.0, giving us the concept of PCI compliance.

Unfortunately, the issue of ownership still was not fully addressed, and just under 2 years later on September 7, 2006, we saw the founding of the PCI SSC and its Web site www.pcisecuritystandards.org. Comprised of American Express, Discover Financial Services, JCB, MasterCard Worldwide, and Visa International, the PCI Council (as it came to be known) maintains the ownership of the DSS, approved vendor lists, training programs, and interpretation responsibility for the requirements.

NOTE

Since our last edition, the Council has been busy adding to their sphere of influence. Forensic investigators are now qualified by the Council under the PCI Forensic Investigator (PFI) program, Integrator/Resellers now have the Qualified Integrators and Resellers (QIR) program, and the Payment Application Qualified Security Assessor (PA-QSA) program is in full swing listing approved assessors for the PA-DSS.

Even today, each card brand/region still maintains its own security program beyond PCI DSS. These programs go beyond the data protection charter of PCI and include activities such as fraud prevention. The information on such programs can be found in [Table 3.6](#). In certain cases, your PCI ROC may need to be submitted to a payment brand's program office separately if you have separate processing relationships with them.

PCI COUNCIL

The PCI Council or, fully, PCI Security Standards Council or PCI SSC, describes itself as “open global forum, launched in 2006, that is responsible for the development,

Table 3.6 Brand Security Programs

Card Brand	Additional Program Information
American Express	Web site: https://www209.americanexpress.com/merchant/singlevoice/dsw/FrontServlet?request_type=dsw&pg_nm=home&ln=en&frm=US E-mail:American.Express.Data.Security@aexp.com
Discover	Web site: http://www.discovernetwork.com/merchants/data-security/index.html E-mail:askdatasecurity@discoverfinancial.com
JCB	Web site: http://partner.jcbcard.com/security/jcbprogram/index.html E-mail:riskmanagement@jcbati.com
MasterCard	Web site: http://www.mastercard.com/us/company/en/whatwedo/security_fraud_management.html E-mail:sdp@mastercard.com
Visa Inc.	Web site: www.visa.com/cisp . E-mail:cisp@visa.com
Visa EU	Web site: http://www.visaeurope.com/en/businesses_retailers_payment_security.aspx
Visa Canada	Web site: www.visa.ca/ais

management, education, and awareness of the PCI Security Standards, including the Data Security Standard (PCI DSS), Payment Application Data Security Standard (PA-DSS), and PIN Transaction Security (PTS) requirements” [8].

The PCI Council charter provides oversight to the development of various PCI security standards (including PCI DSS, PA-DSS, and PTS) globally as well as maintaining the four vendor certifications programs for on-site assessments (ISA, QSA, PTS Labs, and PA-QSA), scanning companies (ASV), PFI, and QIR. The PCI Council publishes the updated DSS, at the time of this writing at version 3.0, which is accepted by all brands and international regions; it also updates the supporting documents such as the “PCI Quick Reference Guide” and “Prioritized Approach Tool 3.0” and a slew of supporting documents ranging from interpretation guidance to output from the various Special Interest Groups.

The lists of current PA-QSA, QSA, QIR, PFI, PCI Professional (PCIP), Point To Point Encryption (P2PE), PCI Recognized Labs, and ASV companies are located at the council Web site: https://www.pcisecuritystandards.org/approved_companies_providers/index.php. In addition, the Council also runs the Quality Assurance Program (QA Program) for QSAs, PA-QSAs, and ASVs, which are aimed at boosting the overall quality and maintaining the integrity of site assessments and vulnerability scans.

The PCI Council is technically an independent industry standards body (although on the surface it looks more like a training and certification company), and its exact organizational chart is published on its Web site (https://www.pcisecuritystandards.org/organization_info/org_fact_sheet.php). Since our last writing, the Council staff continues to grow, but it still remains a relatively small force of people managing the standards side of the PCI ecosystem.

The PCI ecosystem immediately felt the positive impact of the PCI Council. Merchants and service providers can now play a more active role in the compliance program and the evolution of the standard, whereas the QSA, PA-QSA, PFI, and ASV companies find it much easier to train their personnel.

TOOLS

At the time of this writing, PCI Council provides a few useful tools to help track PCI DSS compliance. These are explained in the following (all available here: https://www.pcisecuritystandards.org/security_standards/documents.php):

The “Prioritized Approach” tracking spreadsheet allows for compliance program tracking and reporting, whether internal or to the card brands or acquirers. One key note about this tool is that it is designed to be one-size-fits-all. Experience tells us that this is rarely the case, and you should customize this for your specific needs.

The “SAQ Instructions and Guidelines” document is helpful for those validating PCI compliance via an SAQ. The PCI Council provides the fillable documents that can be used for tracking compliance at a small organization. All the SAQs can be obtained for free.

“Attestation of Compliance” forms are also provided by the PCI Council. These forms accompany the SAQ during self-assessment or the ROC after the on-site assessment.

To summarize, the most important things to know about PCI Council are as follows:

- The Council maintains and updates the PCI DSS, PA-DSS, and PTS, as well as all of their related supporting documents.
- The Council does *not* deal with PCI validation process and, specifically, with enforcement via fines or other means. These responsibilities are retained by the payment brands.
- The Council also certifies and maintains the lists of security vendors as QSAs, PA-QSAs, ASVs, and PCIPs, as well as polices the vendors to maintain the integrity of PCI validation.

Let's look at QSAs, PA-QSAs, PFIs, and ASVs in more detail.

QSAs

The PCI Council administers the QSA⁴ program in which members are allowed to conduct on-site DSS compliance assessments. These companies have gone through the application and qualification process, having to show compliance with tough business, capability, and administrative requirements. QSAs must invest in personnel training and certification to build up a team of assessors, also called QSAs.

NOTE

QSAs are only permitted to conduct on-site DSS assessments. They are not automatically granted the right to perform perimeter vulnerability scans, unless they also certify as an ASV. Many companies today can be found on multiple lists (QSAs, PA-QSAs, PFIs, and ASVs) to be able to provide complete PCI validation services to merchants and service providers.

QSAs have to recertify annually via a computer-based training module. The exact qualification process and the requirements are outlined on PCI Council Web site; however, of particular interest are the insurance requirements. QSAs are required to carry high coverage policies, much higher than typical policies for the professional service firms, which becomes important later. A lawsuit (“Merrick Bank v. Savvis,” see more details in [9]) presents an example of the risk that QSAs face. In this suit, a bank is suing the assessor who validated CardSystems, a victim of a massive card data breach, as PCI compliant. Use your favorite search engine to find more lawsuits filed against prominent QSAs.

⁴In the past, there was a different name for a company (QSAC or Qualified Security Assessor Company) and an individual professional employed by such company (QSA or Qualified Security Assessor).

NOTE

The QSAs are approved to provide services in particular markets or subsets of markets: US, Asia Pacific, CEMEA (Central Europe, Middle East, and Africa), LAC (Latin America and the Caribbean), and Canada. The qualification to service a particular market depends on the QSA's capabilities, geographic footprint, and payment of appropriate fees.

Individuals wanting to become a QSA must first and foremost work for a QSA company or for a company in the process of applying to become a QSA. Then, they must attend official training administered by the PCI Council and pass a written test. They must also undergo annual requalification training to maintain their status. An individual may not be a QSA, unless he or she is presently employed by a QSA company; however, a QSA can carry the certification between QSA companies when changing jobs.

TOOLS

Anybody can look up the individuals with current QSA certification by using QSA Employee Lookup at https://www.pcisecuritystandards.org/approved_companies_providers/verify_qsa_employee.php.

See [Figure 3.2](#) for an example.

We cover the tips on working with your QSA in Chapter 11. If there is one thing to remember when engaging a QSA, the QSA should be your partner. Treating your

Search Result**Valid QSA**

Name: **Erin Jacobs**

QSA Certified Through: **10/14/2012 (MM/DD/YYYY)**

Company: **Urbane Security**

Company Phone: **1-312-970-0317**

The assessor appears to be in good standing with the PCI SeCUrity Standards Council (SSC) as a Qualified Security Assessor.

We advise that you call the assessor company to validate the identity of the assessor you are working with.

If the assessor has been appropriately identified but the QSA and/or PA-QSA Company displayed next to their name is no longer current, please advise the assessor to update their records with the PCI SSC with the new QSA Company.

FIGURE 3.2 QSA Employee Lookup Tools

QSA like an auditor will only lead to a painful process whereby both parties end up frustrated and disillusioned. This does require you to vet your QSA as not all QSAs are equal. More on this later.

PA-QSAs

PA-QSAs assess payment applications as part of the PA-DSS program that used to be Visa's Payment Application Best Practices program. Individuals wanting to assess applications under this standard must apply for a special designation called a PA-QSA and take additional training. You cannot be a PA-QSA without first becoming a QSA, and just like in the QSA world, your company must be signed up as a PA-QSA company in order to perform PA-DSS assessments against payment applications. PA-DSS is outside the scope of this book, but you can read about it on the PCI Security Standards Web site.

In addition, we now have the P2PE designation to assess P2PE Solutions and Payment Applications. In order to qualify, you must already be a PA-QSA in good standing. Only those individuals who have the P2PE designation are approved to perform these Domain 2 assessments.

PRINCIPAL-ASSOCIATE QSAs

This special case, not to be confused with the PA-QSA above, can allow certain individuals to perform functions as a QSAs under the agreement of another, larger QSA company. The use case for this special group of people is for new or small markets that may not be able to sustain a full QSA company, but need to have a local, trained consultant to perform an on-site assessment for a willing merchant or service provider. Branden used this concept at VeriSign when he needed local services in Australia but didn't have an official presence there after we exited the market as a business. Branden found a local Australian company that was willing to pay the reduced Principal-Associate QSA registration fees, so he hired them and they fell under VeriSign's global QSA designation.

PFIs

PFIs combine all the individual payment brand programs around forensic investigations into one Council program like all the others you have read about. Like PA-QSAs, PFIs must be a QSA, and companies with the designation must also be QSAs in each region they want to do PFI work. There is no training requirement for PFIs; however, they must have experience on their resume and should include copies of certificates from their forensic-related training courses for review.

Contrary to popular belief, even though the Council manages the PFI program, they do not get copies of the forensic reports to create that closed-loop feedback channel the industry is asking for as it relates to negligence by a QSA, PA-QSA, or ASV after a breach.

PCIPs

The PCIP designation is new to the program since our last edition and is open to anyone who passes the test and pays the fees. It was designed to be a designation that individuals can take with them, regardless of their employment status. It works like a typical certification in that manner.

QIRs

The QIR program was created to combat the rather poor job that many POS integrator or resellers do with deploying PCI DSS compliant solutions. As of this writing, there are only three on the list. If you are using an integrator or reseller, check to make sure they are on the list.

ASVs

As you know, PCI DSS validation also includes network vulnerability scanning by an ASV.

To become an ASV, companies must undergo a process similar to QSA qualification. In addition to a training class for each analyst to be trained and performing ASV-related duties, ASVs must submit a scan report conducted against an out-sourced test network perimeter. ASVs must certify at least two analysts before they can be approved. An organization can choose to become both a QSA and an ASV, or they could simply do one or the other.

ASVs are authorized to perform external vulnerability scans from the Internet, but PCI DSS also mandates internal vulnerability scans (performed from inside the company network), which can be performed by any qualified individual like an internal security team or consultant.

We cover all the tips on working with your ASV in Chapter 8.

QUICK OVERVIEW OF PCI REQUIREMENTS

Now it is time to briefly run through all 12 PCI DSS requirements, which we cover in detail in the rest of this book.

PCI DSS version 2.0 comprises six control objectives that contain one or more requirements:

- 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:* Protect all systems against malware and regularly update anti-virus software or programs.
 - *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:* Identify and authenticate access to system components.
 - *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 for all personnel.

The above-mentioned 12 requirements cover a vast spectrum of information technology (IT) areas as well as venture outside of IT in Requirement 12. Some requirements are very technical in nature (e.g., Requirement 1 calls for specific settings on the firewalls), and some are process and policy-oriented (e.g., Requirements 7 and 12) and even go into contract law (some of the subrequirements in Requirement 12 cover the interactions with MSPs).

The detailed coverage of controls makes things easier for both the companies that have to comply with the standards, the auditors (in case of Sarbanes–Oxley Act of 2002 [SOX] or other laws and standards), or the assessors (in case of PCI DSS). For example, when compared with SOX, companies do not have to invent (or pay for somebody to invent) the controls for them; they are already provided. This can also create challenges as compliance initiatives become more prescriptive about their required controls, companies are forced to create a common control set and map them back to all of the individual compliance requirements.

What is interesting is that almost every time there is a discussion about PCI DSS, someone would claim that PCI is too prescriptive. In reality, PCI being prescriptive is the best thing since antivirus solutions invented automated updates (hopefully, you can detect humor here). PCI DSS prescriptive nature simply means that there is some specific guidance for people to follow and be more secure as a result (if they follow the spirit and not only the letter of PCI standards)! Sadly, in many cases, the merchants who have to comply with PCI DSS and who still think it is “too fuzzy” and “not specific enough” are the ones either fighting to comply in the first place, trying to avoid changing anything at all in spite of PCI DSS, or looking for a simple compliance and security *to do* list or a task list, and no external document that guarantees that your organization will be secure can ever be created.

In particular, when people say “PCI is too prescriptive,” they actually mean that it engenders “checklist mentality” and leads to following the letter of the mandate blindly without thinking about why it was put in place. For example, it says “use

a firewall,” so they deploy a shiny firewall with a basic “ALLOW ALL<->ALL” rule—an obvious exaggeration that clarifies the message here. Or, they have a firewall with a default password unchanged, or maybe slightly more secure by allowing all outbound and denying most inbound traffic. In addition, the proponents of “PCI is too prescriptive” tend to think that fuzzier guidance (and, especially, prescribing the desired end state *and* not the tools to be installed) will lead to people actually think about the best way to do it.

So, the choices to write security-motivated regulatory guidance are as follows:

1. Mandate the tools (e.g., “must use a firewall”) and risk “checklist mentality,” resulting in both insecurity and “false sense” of security.
2. Mandate the results (e.g., “must be secure”) and risk people saying “yes, but I don’t know how” and then not acting at all, again leading to insecurity and a wide interpretation of intent.

The author team is of the opinion that in today’s reality #1 works better than pill #2, but with some pause to think, for sure. Although the organizations with less mature security programs will benefit at least a bit from #1, organizations with more mature programs might be able to operate better under #2. However, data security today has to cover the less-enlightened organizations, which makes the #1 choice—embodied by PCI DSS—the preferred one.

As far as scope of PCI DSS within the organization is concerned, PCI compliance validation may affect more than what you consider the “cardholder environment.” According to PCI DSS 3.0, the scope includes the cardholder data environment as well as anything connected to it. Chapter 4 will help you with the scoping problem including giving you some ideas on how to reduce its impact. If you do not have basic network segmentation controls in place, the scope of PCI compliance validation will cover your entire network. Think about it: if you cannot ensure that your cardholder data is confined to a particular area, then you cannot focus on this area alone, and you have to look everywhere.

NOTE

Just because a POS system is on the list of compliant payment applications (PA-DSS), it does not mean that your particular implementation is compliant. Also, it definitely does not mean that your entire organization is PCI compliant. Only applications configured and maintained according to their PA-DSS Implementation Guide will be able to operate in a compliant manner. You should work with the application vendor and with your QSA to verify this.

In order for the device to be added to the PA-DSS list, the payment application, online shopping cart, or POS vendor has to show and document the secure method for their application deployment. However, it is ultimately the merchant responsibility to follow the secure and compliant deployment guidance.

For merchants using an integrator or reseller, ensure they are deploying and managing your POS in a compliant manner. If they are doing things properly, they should be on the QIR list.

For the benefit of consumers who may be more familiar with a brand name rather than a parent company (e.g., TJX is the corporate parent of TJ Maxx), PCI compliance validation should always follow the merchant ID. Any transaction processed under that merchant ID, regardless of origin, should fall under that company's PCI validation process.

You may discover that you are unable to always comply with the strict letter of PCI DSS while striving for its spirit. For example, you may need to temporarily store cardholder data unencrypted for troubleshooting purposes or to use a password of less than mandated minimum length on a legacy system. Another example may include recording certain call-center conversations for customer service purposes. Again, card brands understand that these recordings may contain cardholder data, so accommodations must be made accordingly to protect that data.

In many cases, compensating controls have to be used to achieve compliance when your company cannot exactly meet a given requirement. The important thing to remember about compensating controls is that they have to go beyond the requirements of PCI to provide the same or higher assurance of cardholder data protection. When compensating controls are used you must gather and supply additional documentation about the control. Please see Chapter 14 for detailed coverage of compensating controls.

CHANGES TO PCI DSS

One of the key challenges for any security standard is to change fast enough to address the changes to the threat environment (and this changes literally every day since the criminal computer underground has to evolve to stay in business) and to change slow enough to still be considered a technical standard (and not simply advise to “do the right thing”). For prescriptive technical standards that directly call out security controls such as firewalls, network intrusion prevention, and vulnerability scanning, the challenge is even more extreme.

PCI DSS is sometimes criticized for being “constantly in flux” and for “not moving fast enough” at the same time, but by different people.

The PCI standards are governed by a process called the “Lifecycle Process for Changes to PCI DSS” [10]. By the time you read this, we will be in the “Feedback Begins” phase of the lifecycle:

1. Standards Published (October),
2. Standards Effective (January 1 following release),
3. Market Implementation (All year),
4. Feedback Begins (November),
5. Old Standards Retired (December 31, which is 15 months past Phase 1),
6. Feedback Review (April–August),
7. Draft Revisions (November–April),
8. Final Review (May–July).

The overall process takes 3 years and always includes extensive public commenting and review periods to incorporate the input from all stakeholders.

PCI DSS AND RISK

The relationship between PCI DSS and risk management isn't harmonious. PCI DSS's goal is to reduce the risk of card transactions and to build consumer confidence in the payment card systems. On the contrary, many people point out that PCI DSS presents a list of controls with no regard to an organization's own risk assessment. Let's explore the relationship of PCI and risk a bit further.

First, a common question: can one claim that complying with PCI increases the merchant's overall business risk? When people ask that question they usually imply that PCI added the risk of loss via noncompliance fines and raised fees to the risk of direct losses due to card theft from a merchant's environment (such as reputation damage, cost of new security measures, and monitoring). The answer is clearly a "no," since before PCI, most of the negative consequences of a card theft, even a massive one, were not falling upon the merchant shoulders but on others such as card-issuing banks. PCI, on the contrary, creates a powerful motivation for protecting the data on the merchant side.

Despite that reality about PCI, many CEOs or CFOs are still asking the question, "Why would I need to spend money on PCI?" And, no, the answer is not "Because there are fines" (even though there are noncompliance penalties). The answer is that the list of negative consequences due to neglecting data security and PCI DSS is much longer than fines.

Your company's contract with the acquiring bank probably has a clause in it that any fines from the card brand will be "passed through" to you. With all compliance deadlines passed, the fines could start tomorrow. Visa maintains a global Compliance Acceleration Program that fines acquirers (which will pass on the costs to the merchant) between \$5,000 and \$25,000 per merchant per month if their Level 1 or Level 2 merchants are not reported as compliant. In addition, fines of \$10,000 per month may already be imposed today for storing prohibited data.

On top of that, if your organization is not compliant with PCI DSS when you are compromised, higher fines are imposed as well. And if you find yourself in that situation, you might just end up with more data compromised, including that dreaded sensitive authentication data, which drives fines up even further. However, believe it or not, if compromised, this will be the least of your concerns. Possible civil and criminal liabilities could dwarf the fines from the card brands in overlitigious societies that cultivate class-action lawsuits. Some estimates place the cost of compromise at \$50–\$250 per stolen account (note stolen, and not per one used for fraud, which will likely be a subset of the whole stolen card pool). Some companies that have been compromised have been forced to close their doors or sold to competition for nominal amount. Smaller merchants fall victim to this reality often, and companies in the business of protecting cardholder data don't last long after compromises.

Let's use The TJX Companies, which operates stores like TJ Maxx, Marshalls, and so on, as a case study. On January 17, 2007, TJX announced that they were compromised. Because they did not have robust monitoring capabilities such as those

mandated by PCI, it took them a very long time to discover the compromise. The first breach actually occurred several years prior. TJX also announced that more than 90 million credit card numbers were compromised. In addition to the fines, volatility in the stock price, and direct costs of dealing with the compromise, over 20 separate law suits were filed against TJX; some of which were converted to class-action status. At the time of this publication, most if not all have been settled. This is good news for the rest of us because most of the outcomes are public record. It might take you some time, but with a good search engine and some time you could add up all of those losses into one big, fat scary number.

At the time of this writing, the industry is reeling from the Target breach announced late in 2013. While there are still a number of theories out there, the fact that an unattended remote access account could be the source for the breach is significant. Don't forget that this itself may be a violation of PCI DSS, though the circumstances of how that particular account was accessed is unclear. Spend a few moments in your favorite search engine to see what lawsuits are now public around this. Chances are, you will find some very interesting documents surrounding the players here and we assert that this increases risk for QSAs.

Whether you believe your company to be a target or not, the fact is that if cardholder data comes into contact with your network at some point *you are a target!* The resale value of cardholder data has plummeted dramatically in the last few years, but that doesn't mean that the size of the target on your back is smaller. You and your organization are simply someone's sheep to be fleeced, and your losses are their gains. Organized crime units profit from credit card fraud, so your company is definitely on their list if you deal with card data. International, federal, and state law enforcement agencies are working hard to bring perpetrators to justice and shut down the infrastructure used to aid in credit card-related crimes; however, thousands of forum sites, Internet chat channels, and news groups still exist, where the buyers can meet the sellers. Data breaches like the ones at TJX and Target are not the work of simple hackers looking for glory. Instead, well-run organizations from the Eastern European block [11] and selected Asian countries [12] sponsor such activity and earn a great living from various illegal hacking activities.

The Web site <http://datalossdb.org> maintains the history of the compromises and impacts in terms of lost card numbers and other records. Over 800 million personal records (a mix of cards, identities, etc.) were compromised in 2013 alone. This includes companies of all sizes and lines of business. If the industry does not get this trend under control, the US Congress will give it a try.

Finally, and few people actually know it, but PCI DSS does mandate a formal risk assessment, not just a list of controls to implement! Requirement 12.2 requires "an annual process that identifies threats, and vulnerabilities, and results in a formal risk assessment." One of the best things you can do for your business is take this requirement very seriously, and expand it beyond the confines of cardholder data. Don't perform this to check an audit box, take it as an opportunity to get a broad look at how your business operates and the kind of risk it carries.

BENEFITS OF COMPLIANCE

While the inclusion of benefits is irrelevant—after all PCI DSS compliance is *mandatory* for the organizations that deal with payment cards—it is worthwhile to highlight the fact that PCI DSS has important benefits for the merchants, acquiring banks, issuing banks, as well as for the public at large.

If we are to mention one benefit, PCI DSS has motivated security improvements in businesses (especially retail) like nothing else before it. Many of us lived through the virus-infested 1980s, then worm-infested and spammy 1990s, and then through heavy data loss early 2000s without doing anything on security. PCI DSS moved the needle farther toward the “Secure” end for the laggards that fell victim to that reality.

CASE STUDY

Much of this book focuses on case studies where a company makes a mistake, or fails to do something that result in a breach. This case study is a nice change of pace where we examine someone doing something right!

THE CASE OF THE DEVELOPING SECURITY PROGRAM

Yvette’s Evangelical Emporium is a small chain of 50 stores supplying religious supplies to local churches and individuals. Yvette started her business in 1990 with a single store. Throughout the 1990s, she was able to open several new stores in neighboring counties and states, eventually building a 10 retail location business by 2000. In 2002, she took advantage of a depressed economy, and using some capital from investors and a significant trust that matured, she expanded her operation to 25 stores in 3 years and continued to expand over the next 4 years to double her size.

In 2005, Yvette realized that she needed to formalize her IT division and hired Erin, a progressive and security-minded IT executive, as her chief information officer. Erin presented a plan to standardize and build out her infrastructure so that future growth could be done in a cookie-cutter fashion, thus saving millions in deployment and maintenance costs.

By 2006, they crossed the threshold from a Level 2 Visa merchant to a Level 1 Visa merchant and knew they would quickly need to put a solid PCI compliance program in place. Erin knew from her previous experience that small companies struggled with information security and made it a point to build in basic information security fundamentals into her IT operations, but they did not meet the baseline PCI DSS requirements and needed to be reworked.

Because of her new reporting levels for PCI, Yvette hired Steve to serve as the chief information security officer, reporting directly to her. Steve’s task was to build an information security program that addressed PCI immediately, but would expand to be more applicable to information security such that future regulation would only require minor tweaks to the program.

Steve and Erin worked closely together to build a common set of controls to be rolled out to the entire company. Steve knew that PCI was a priority but considered everything he did in light of the ISO security framework (ISO17799 at the time). In some cases, he found that ISO far exceeded specific PCI requirements like in Business Continuity and Risk Assessments, and he found unique parts of PCI that were much more granular than ISO, like the treatment of sensitive authentication data (PCI Requirement 3.2). Steve's efforts ultimately paid off in spades as his information security program matured. Recent changes and additions to restrictions on health care data (which Yvette housed as part of an employee self-insurance program) and state data breach notification laws were already addressed by the program as it matured, and Yvette's cost associated with protecting data was much less than her competitors who only chased standards with immediate noncompliance repercussions.

NOTE

It is well-known that initial PCI DSS creators were well aware of ISO 17799 and other security standards. This awareness leads to the fact that if your organization has a solid security management program based on ISO IEC 27002 (a modern descendant of ISO/IEC 17799 and BS7799), your PCI effort will be relatively easy and you will gain both solid security and compliance as a result. It is also likely that compliance with other regulations will not be overly onerous. PCI is more granular, whereas ISO is more broad, but they are largely in sync!

Our technical reviewer suggested using the concept of control leveling. The idea being that you define a standard or control that works for all systems, not just PCI systems, and build it into your organization's base build. This may not always work in the case of some legacy systems, but it's a good place to build fundamental security into your organization. Another example could be standards for audit lots and retention standards for all systems.

THE CASE OF THE CONFUSING VALIDATION REQUIREMENTS

Garrett's Gas Guzzling Garage operates 800 car repair locations across the United States. Garrett's recently opened 20 locations in Mexico City to help maintain and upgrade the fuel efficiency of old cars. Garrett is considered a Level 1 merchant in the United States but set up a different entity in Mexico City, and processes and settles locally with Bancomer. Although Garrett authorizes and settles locally in Mexico City at his small regional headquarters, he shares the data with the US-based parent for backup and analysis purposes.

His business is booming in Mexico City, and that business quickly became a Level 2 merchant. According to MasterCard's new validation requirements, this would mean that Garrett must have a QSA perform an on-site assessment of compliance for both locations, or get someone internally trained under the ISA program. He was already doing this in the United States based on his Level 1 status, but now faces additional costs for doing this locally in Mexico City.

"But wait," some of you are saying, "What about Visa's rules?" Certainly glad you asked that! According to Visa, if a smaller, wholly owned subsidiary shares

infrastructure and data with a parent company considered a Level 1, then that smaller subsidiary should also be viewed as a Level 1 and perform the same level of validation.

Back to Garrett. Even though the 20 Mexico locations process through Bancomer, the data is shared with the US headquarters for backup and analysis. According to Visa's rules, the Mexican entity is considered Level 1 based on its relationship with the parent, and a Level 1 assessment must be performed.

As always, when in doubt, ask your acquirer what is expected of you. Your mileage may vary when it comes to some of these intricate rules. Some acquiring institutions may still treat certain subsidiaries as lower levels depending on the circumstances.

NOTE

There are two books that every IT and security person should read, both lead by the great Gene Kim. The first is The Phoenix Project, and the second is Visible Ops Security. Do yourself a favor and buy both of these books, and don't forget to tell Gene how much you enjoyed them!

SUMMARY

PCI refers to the PCI DSS established by the credit card brands. Any company that stores, processes, or transmits cardholder data has to comply with this data protection standard. Effectively, all the target compliance dates have already passed, so if your company has not validated compliance, you are at risk for fines and other negative consequences of insecurity and noncompliance. The PCI is composed of 12 requirements that cover a wide array of business areas. All companies, regardless of their respective level, have to comply with the entire standard as written. If you end up filling out an SAQ, you are responsible for validating the subset that applies to you, but don't forget about the rest of the standard, *especially* if the nature of your business changes! The actual mechanism for compliance validation varies based on the company classification, driven by the individual card brand, transaction volume, exact method of accepting cards, and so on. The cost of dealing with data breaches keeps rising, as does their number; noncompliance exacerbates the loss in case of a breach. Companies that do not take data security and compliance efforts seriously may soon find themselves out of business.

Now is the time to start the journey toward data security and compliance: get an endorsement from the company's senior management and business stakeholders, and start fulfilling your obligations and protecting the data.

REFERENCES

- [1] UB Sinclair Jr, I, Candidate for governor: and how I got licked (1935), ISBN 0-520-08198-6; repr, University of California Press, (1994) p. 109.

- [2] PCICouncil Website, Article #5410. <<http://selfservice.talisma.com/article.aspx?article=5410&p=81>>. [accessed 31.08.09].
- [3] PCI Council Glossary, Entry Service Provider. <<http://selfservice.talisma.com/display/2n/index.aspx?c=58&cpc=MSdA03B2IfY15uvLEKtr40R5a5pV2lnCUb4i1Qj2q2g&cid=81&cat=&catURL=&r=0.73831444978714>>; 2009 [accessed 17.07.09].
- [4] Joel Weise, private communication, e-mail dated July 1, 2009.
- [5] Ten Common Myths of PCI DSS. <http://www.pcisecuritystandards.org/pdfs/pciscc_ten_common_myths.pdf>; 2009 [accessed 17.07.09].
- [6] Visa Cardholder Information Security Program for Service Providers Web page. <http://usa.visa.com/merchants/risk_management/cisp_service_providers.html>; 2009 [accessed 02.08.09].
- [7] PCI and Your Third-Party Service Providers – First, the Bad News. <<http://treasuryinstutepcidss.blogspot.com/2009/07/pci-and-your-third-party-service.html>>; 2009 [accessed 10.08.09].
- [8] PCI Security Standards Council Web site. <www.pcisecuritystandards.org/>; 2011 [accessed 05.12.11].
- [9] Merrick Bank v. Savvis Update: Savvis Files Motion to Dismiss. <<http://infoseccompliance.com/2009/06/23/merrick-bank-v-savvis-update-savvis-files-motion-to-dismiss>>; 2009 [accessed 17.07.09].
- [10] Lifecycle Process for Changes to PCI DSS. <http://www.pcisecuritystandards.org/pdfs/OS_PCI_Lifecycle.pdf>; 2009 [accessed 17.07.09].
- [11] Black Hat: Fighting Russian Cybercrime Mobsters. <www.informationweek.com/blog/main/archives/2009/07/black_hat_fight.html>; 2009 [accessed 11.08.09].
- [12] Chinese Hackers Attack Web Site Over Uighur Film. <www.bloomberg.com/apps/news?pid=20601081>; 2009 [accessed 11.08.09].

Determining and reducing the PCI scope

4

INFORMATION IN THIS CHAPTER:

- The basics of PCI DSS scoping
- The “gotchas” of PCI scope
- Scope reduction tips
- Planning your PCI project
- Case study

Scoping your Payment Card Industry (PCI) environment is one of the most critical things you must get right in your quest to comply with this daunting standard. So many companies have cost themselves thousands and even millions of dollars by over- or underscoping their environments and applying controls to the wrong subset. It also seems like the easiest way to get into a heated debate around Payment Card Industry Data Security Standard (PCI DSS) is to find something wrong with a peer’s scoping process or end result. A Special Interest Group (SIG) was put together on this and while ultimately didn’t come out with a special report like other SIGs did, four different documents came out of that group’s body of work, all related to scoping. If you have been watching the flurry of documents released this year, you might remember the EMV, Tokenization, Roadmap for Encryption, and Point-to-Point Encryption guidance documents—all of which contain content produced by the scoping SIG. Throughout this chapter we will talk basics, get through some of the “gotchas,” give you some tools on planning your project, talk scope reduction, and provide a couple of case studies.

THE BASICS OF PCI DSS SCOPING

If you look at scoping on the surface, it simply can’t be as hard as people make it out to be. If your environment contains Primary Account Numbers (PANs) either in storage or flowing through it, some part of your network must comply with PCI DSS.

Simple, right? On the surface, everything looks simple. Especially from this high horse way up here.

The majority of the discussions around scope typically end up argumentative because one person is interpreting the standard more leniently than another. Most of the discussions the authors have witnessed, or been thrust into the middle of, on scoping start because one party didn’t want to comply with the standard in part or at all. We’ve learned through the years that denial is a very powerful human defense

mechanism. It's easy to ignore the requirements, or come up with arguments to why the rules shouldn't apply to you. It's not easy to do things the right way. Reducing the scope and making business decisions about PCI DSS becomes easier when you define your scope properly from the start. It's probably better to have the scope exclusion discussion about some part of your network if you automatically include everything in scope in the beginning. Then it really comes down to a strong case on why certain components should be excluded over others from scope.

Per the PCI DSS, the scope is now defined as follows:

The PCI DSS security requirements apply to all system components included in or connected to the cardholder data environment. The cardholder data environment (CDE) is comprised of people, processes and technologies that store, process, or transmit cardholder data or sensitive authentication data. “System components” include network devices, servers, computing devices, and applications (page 10 of the PCI DSS).

The definition isn't only about technology, so don't stop there. You also have to consider the people, processes, and technology that store, process, or transmit cardholder data or sensitive authentication data.

Scoping guidance like what you see above starts on page 10 of the PCI DSS, and it is not meant to be an exhaustive list. They do provide examples of the different types of elements that should be in scope, so you should be able to find an analog for your particular item. You can expect other strange scenarios to present themselves while going through this exercise. The authors have gone through this process with many customers, and it's amazing how many times we've both said, "Wow... never saw that one before." Even in 2013! You will run into many scenarios like that while involved with PCI DSS, and you are better off assuming it is in scope first, and then look for ways you could possibly exclude it.

Hopefully you aren't terrified yet, but the wheels are turning. Your scope includes any place where cardholder information is present at any given time. The duration of time in which it is present is irrelevant, as is the complexity of an attack required to capture that information during a compromise. Don't even start with those discussions yet as it will take valuable time away from defining the size of the problem.

Both authors suggest leveraging the Socratic Method to help define the problem (check your search engine for more details). It will force you to ask the right kinds of questions in order to put all of the required elements on the table (so to speak). While this may sound extreme, defining your scope is the most critical part of this process, so find whatever tool you believe to be most effective and use it!

Smaller businesses have some advantages over larger businesses in that their processes around cardholder data tend to be simpler, and if it is a relatively young small business, more electronic and potentially already outsourced. Small businesses struggle with advanced information technology (IT) concepts because they typically have neither the budget nor the staff to tackle them. On the other side of the fence, larger businesses typically have many of the IT processes in place to protect cardholder data, but they have no idea where it is. Sure, there are areas where they can say it exists with some assurance they are correct, but for the most part they cannot exclude

large portions of their environment because they have no idea if these “noncritical” areas have cardholder data.

Cardholder data is more mobile than people are willing to accept. For example, in an environment where IT people are diagnosing and fixing equipment, sensitive data routinely travels beyond its intended borders. Point Of Sale (POS) technicians debugging faulty terminals will end up with cardholder data on their laptops. Is it securely erased? What happens when that laptop re-joins the corporate network? Is there an automatic backup process that will further proliferate this cardholder data into systems in which it is not supposed to exist?

Things get worse when you consider how a concierge-like service could be carried out by a small business owner. Let's say that Ryan wants to keep card information for his top customers so they only need to tweet in an order and it will be paid for and ready for pickup that evening. How does he store that information? Chances are, he's dropped it into the contact record for the customer. And let's say he has an iPhone, so this is now syncing to iCloud. And of course, to the other three or four machines tied into that account (iPad, Mac, etc.), one of which is also syncing information to Google, and the problem balloons out of control.

We're dragging you through this field of broken glass to hopefully help you understand that determining the scope the right way is painful and will probably require some kind of tool to rescue you. You won't be able to do this entirely on your own with sheer manpower. You need a way to proactively discover and map cardholder data in your environment. You may not need to go the full Data Loss Prevention (DLP) route for your environment, but you will definitely need some amalgamation of tools to help you wrangle this problem.

TOOLS

There are both free and commercially available solutions for finding certain types of data and controlling their destiny. Here are a few examples:

Spider from Cornell Labs (<http://www2.cit.cornell.edu/security/tools/>). This particular tool is available, with source code, for Windows, Unix, and OS X environments and is good for singular scans. It may not scale well if you have hundreds or thousands of servers to check, but for small IT environments it is quite effective. Another alternative to this is opendlp (<https://code.google.com/p/opendlp/>).

GNU Grep (<http://www.gnu.org/s/grep/>)—the original data discovery tool. Any technophile that has administered servers professionally knows about this tool and has used it to track down wayward data in various forms. If you wanted to use this tool to triage systems and files that may contain cardholder data, you might combine it with a regular expression like this:

```
grep -rl "\((4[: digit:])\{3\}\)\|(5[1-5][: digit:]\)\{2\}\)\|(\(6011\)\)[-,: space:]\)?[: digit:]\{4\}\[-,: space:]\)?[: digit:]\{4\}\[-,: space:]\)?[: digit:]\{4\}\|3[4,7]\[: digit:]\{13\}\|3[0,6,8]\[: digit:]\{12\}\)" /
```

The issue with this approach is the large volume of false positives you must sift through. You could pipe the output from the above to script that could reduce the false positives by running each hit through a Luhn check (<http://j.mp/IveOgk>), and then piping that output to a mail script to dump the contents into an e-mail box for individual follow-up (but be sure not to dump the card number in there!).

(Continued)

TOOLS (cont.)

Commercial solutions tend to come in two forms, system-based and network-based detection. Companies like Websense, Cisco, and RSA have solutions built into their portfolio. For system based, you will find agent or agentless solutions provided by companies like Symantec, RSA, GroundLabs (www.groundlabs.com), and iScan Online (www.iscanonline.com). The benefit to the commercial solutions tends to come in the form of automatic false-positive reduction techniques, scalability, and native file format searching. When determining whether you want to roll your own solution or go with a commercial provider, be sure to include the salary cost of someone maintaining the home-grown version, and run this through the “What happens if Sally wins the lottery?” scenario.

Finally, some systems now have powerful searching capabilities built into their software. In order for them to be effective for your purposes, they must include the ability to do pattern-based searches. Because you want to search for anything that would meet the pattern of a valid credit card number, you have to be able to simplify it into something like the regular expression above.

You will most likely end up with a combination of the above tools to accomplish your goals as none of the above are silver bullets. Each has benefits and limitations.

Once you put your tools in place, you will have to go through the process of determining what is real and what is a false positive. One of the authors had a customer who used a 16-digit routing number to track certain kinds of packages as they moved from location to location. Now, of course, every tracking number didn’t show up as card number to track down, only the ones that started with 36, 37, 4, 5, and 6011, and only one in every 10 or so of those. But when one of their locations was known internally as 60110202, many packages destined for that location set off credit card data alarms. In this case, those 16 digit numbers that passed a Luhn check would not be considered cardholder data, but it certainly is a discussion that will be had with your assessor.

After your first tool run you may feel overwhelmed by the amount of cleanup work you have to do. Don’t be. Think of it as a huge opportunity to shore up your environment and greatly reduce your liability and risk by remediating those areas.

Now, after validating all of your false positives and looking at your final pile of work to handle you will see your true scope. Yep, it really is that dire. This is the reason why you want to go through this process early so you can spend the majority of your time over the next few months by learning how the business operates and reducing the scope of your assessment by destroying data where it doesn’t need to exist.

For the areas where the data absolutely must exist, you have a few options yet to help make this process easier. First, you can choose to outsource your processes to a third party, thus effectively transferring that liability to them. There are exceptions to that liability transfer. The cleanest solution would be to have the third party own responsibility over the merchant account and identity (ID) and only send you wire transfers after the transactions settle. You definitely don’t want to resort to trying to reclaim losses from a compromise by reviewing the damages clauses of your contract with that provider. Is it more expensive per transaction to have someone process payments for you? Yep, but how much of those dollars are you spending on information security related to PCI DSS? Check with your CFO, but we see a significant

trend where companies are pushing processes like this into their operating expenses and calling it a cost of doing business (which it is). Unless you are going to invest in a payment processing mechanism to generate revenue by processing other companies’ payments, outsourcing should be a serious consideration for your business.

If this is not an option—which we argue in most cases is in fact the best option—you will need to spend some time building security controls around the areas where you do have cardholder data. Don’t try to bring your entire network into compliance with PCI DSS. You most likely won’t succeed, and it will unnecessarily cost your company thousands or millions. Instead, focus on segmentation, data centralization, and strong access controls to access the raw card data.

THE “GOTCHAS” OF PCI SCOPE

As we discussed earlier, most of the contentious discussions around PCI DSS are from people who are trying to find ways around the requirements, mostly so they don’t have to make any changes to the environment for which they are responsible. We find it ironic that if people would put the same effort into complying with PCI DSS that they did into fighting it, we would see higher compliance rates and more examples of companies really doing it right. Unfortunately, both are not nearly as prevalent as they need to be regardless of what you might hear from a payment brand, acquirer, or even a merchant. If you are a techie person reading this book, you probably will walk away realizing there are a few things your company does that skirt the line of compliance or maybe even blatantly destroy it. Remember, denial is a powerful human defense mechanism—don’t let it be yours.

With that in mind, let’s walk through a few examples of mistakes people have made in determining their PCI DSS scope, in both directions. There are very few examples of over scoping a PCI DSS environment, but we’ll review one now.

One of the authors was brought into a situation by the finance group of a merchant to help them determine the next steps in their PCI compliance process. The internal audit team was taking the first crack at building a case for PCI compliance and had grossly overscoped the environment. They wanted every electronic device in the company to be included in the scope of PCI DSS and put a massive remediation budget in place, including the creation of a new department and a team of 20 heads to whip the company into shape.

The audit team’s rationale for including everything in scope was not too far off the mark, but they missed some simple scope reduction techniques that were much cheaper with a smaller impact to the end systems. The company used a mainframe system to process their cardholder data. Once the data entered the mainframe for processing, it only left to go to the bank over a direct internet protocol (IP) connection on a private telco-provided data line. Because the mainframe was not segmented from the network, it was assumed everything should be in scope. Data was entered into the mainframe from a specially crafted payment terminal that encrypted the traffic over the network. Once inside the mainframe, it was further encrypted and tokenized, such that the only

data used by the company after settlement was token data (dummy information that is tied to a card number, but meaningless without the association to the real card number). What we suggested was to audit the access controls to that data and remove human access entirely. Then create alerts when the access control tables changed to be followed up on by the audit team. The terminals were firewalled off at the store location such that the only way to access them remotely was to use strong authentication and a Virtual Private Network (VPN) connection. With a few other controls, we were able to remove the need for the massive capital expenditure and instead helped that company boost their security and comply with the standard in a matter of months.

The “gotcha” with that example was a very loose interpretation of the scoping statement from PCI DSS. Yes, it does say “*any network component, server, or application that is included in or connected to the cardholder data environment*,” but that tends to break down a bit when you ignore the capabilities of the underlying technology driving the environment. We’ve also learned that mainframe environments tend to throw a monkey wrench into the works because few people really understand how they work, and the security implications of running one. Ultimately, there are a few ways to meet the above italics and still reduce the impact that PCI DSS has to your environment.

NOTE

Mixed Mode is the concept whereby a virtual infrastructure hosts both guests that must comply with PCI DSS and others that are considered out of scope of PCI DSS. If you are considering using this method, be sure the underlying virtualization fabric complies with PCI DSS and the out of scope hosts are sufficiently isolated from the ones that must comply with PCI DSS through access controls and virtual segmentation. Your mileage may vary here, and you should understand both the complexities of the hypervisor and its controls as well as the application or function that is virtual.

Another overscoped example comes to us from interpreting the standard as it relates to virtualized environments. Virtualization as a technology continues to become more present in our environments, even down to our desktops and mobile devices. The PCI Council released a guidance document from the Virtualization SIG in 2011 (and since updated it), and there was both great information and guidance as well as a terrible interpretation of some parts of the standard with editorial comments left in the document for assessors and assesses to argue over for the next couple of years. One author helped a company educate their Qualified Security Assessor (QSA) on what virtualization can do the scope of an environment, and how tackling the “Mixed Mode” problem correctly can help IT departments meet their virtualization targets while keeping data safe and secure. The QSA argued that because the virtual host held both in and out of scope guests, that all guests must comply with PCI DSS regardless of their scope determination. The QSA incorrectly made the mental leap that a virtual host could not be locked down in the same manner a physical data center could. In fact, virtualized infrastructure can be deployed in a way that makes

it more secure than traditional physical deployments, but that's a topic for another book. What ultimately came out of the discussion was a scope that matched the intent of PCI DSS, and focused on making the hosts and in-scope guests fully compliant with PCI DSS. The QSA was able to complete the assessment to their satisfaction, and even learned a thing or two during the assessment process that will ultimately provide their customers with a better assessment experience.

Briefly, you will hear arguments on both sides of the fence here. The reality is this: we have to trust the hypervisor developers to code securely just like we trust Cisco to write IOS securely. If a QSA asked for the source code to the version of IOS on your core router, he would get laughed out the door. He reviews the configuration of the device, just like a QSA should do for the hypervisor.

Now, let's discuss some underscoped examples. One of the requirements we will discuss in this book is Requirement 2.2.1, *“Implement only one primary function per server to prevent functions that require different security levels from co-existing on the same server.”* Now, this can be interpreted in a number of ways as you might imagine, and this one tends to be a key area where scope can be over or underdone. In this case, a company came to one of the authors stating that the in-store server should be considered out of scope because it doesn't store any cardholder information. That particular machine performed a tremendous amount of back-of-house reporting for each store and allowed managers to check their company e-mail and do some basic Web-surfing, perform local antivirus distributions to the POS systems, participate in the corporate Active Directory system as a tree in the forest, and contact internet systems for local DNS resolution and network time protocol (NTP) syncing. There was no segmentation in the store, and the machine did in fact contain one day's worth of credit card data on it that was pulled from the point of sale controller to assist in that back-of-house reporting.

The first step was to convince the internal groups that the machine was, in fact, in scope for PCI DSS. Not only was it on the same network as the point of sale systems, but it contained a day of cardholder data on it for reporting. The company saw that they did make a mistake and the machine should be in scope. The next argument made was that it didn't violate the “one function per server” concept because its function was to support the store, and that was the only function it had. “Supporting the store” can be a broad view of a business function, but it is certainly not a single function as intended by this requirement. Once they understood the intent of the requirement, they changed a few things to keep that server functioning as intended but without the scope issues associated with leaving it connected to the same network. It was segmented off in the store, and the daily reporting information was cleaned and pushed from the POS controllers (as opposed to pulled by the in store server) such that no PCI data was included in the dump.

Another classic underscoping problem is calling a service provider out of scope because you have put contractual language in place between the companies to address compliance from a legal perspective. Beginning with PCI DSS version 2.0, the Council clarified the need to visit some service providers on some kind of basis to ensure they support your PCI DSS compliance. In the old days, QSAs would often see certain companies listed as service providers for certain functions and just

assume they are acting in a compliant manner. Iron Mountain is a classic example of this. Rarely would you see a QSA asking to go visit the Iron Mountain facility where cardholder data might be stored offsite. In fact, some companies would simply call the relationship out of scope of PCI because they had earned such tremendous industry trust and they didn't want to re-open contract negotiations. This is simply not acceptable if cardholder data is being stored, transmitted, or otherwise processed by that service provider. They are absolutely in-scope and should be evaluated just like any internal group that stores, transmits, or processes PAN data.

Now, one way to potentially remove such a service provider from scope is to send them only encrypted data with no access to any keys (meaning you can't include the keys on any media you send along). According to PCI DSS FAQ 1233 (https://pcissc.secure.force.com/faq/articles/Frequently_Asked_Question/Are-third-party-storage-providers-storing-only-encrypted-cardholder-data-in-scope-for-PCI-DSS/), "*if a merchant stores media containing only encrypted data at a third-party back-up storage facility, and the third-party provider has no access to decryption keys and no ability to decrypt the data, then the presence of encrypted data alone would not bring the third-party provider into scope for PCI DSS.*" There are a couple of key concepts to consider here. The delineation is means, not knowledge. So if you are using some kind of obscure crypto technique and include the means to reverse it, you can't count on the lack of the entity's knowledge to exclude them from this requirement.

To avoid all the "gotchas" associated with scoping just keep in mind that you must consider anything in scope that is included in, or connected to, the cardholder data environment. The best way to remove systems from the "connected to" clause is to deploy firewalls around the cardholder data environment's perimeter, separating it from the rest of the enterprise, and eliminating data interchange over the border as much as possible. The more you can do here, the easier your assessment process will go, and the less you will have to rely on QSA interpretation to dictate your fate.

SCOPE REDUCTION TIPS

Now that you have built your scope and know how serious the problem is, you are no doubt taking a serious look at ways to reduce this scope such that the impact to your organization is minimized. The authors have had many of these discussions over the years as companies facing PCI Compliance for the first (or even second and third) time tend to have sloppy IT environments that focus on availability and data processing over segmentation, data privacy, and security. We love working with retailers because they tend to be some of the most innovative thinkers on running their businesses, and they don't hesitate to use every single tool in their arsenal to either solve problems or get ahead of the competition. What they lack is the understanding of how their actions impact the larger company's security and compliance postures. That's where user education comes in, and why companies facing their annual assessment will typically have new problems that pop up year over year.

The first scope reduction technique has already been briefly discussed; complete outsourcing of your payment environment. One of the authors frequently has a more confrontational discussion with CIOs that starts something like: “What business do you have running a payment processor? You are a merchant. Your core competencies are marketing and supply chain management, not payment processing. So why on earth would you put company resources towards doing it in a half-baked way?”

These conversations aren’t as contentious as they sound as they tend to be delivered with a smile, but the point is valid and it forces executives to have a hard discussion about how their business operates. PCI DSS isn’t going away any time soon, and legislation around personal information is only growing. If you are considering global expansion, your regulation minefield just got scarier. Companies must focus on what is important to their bottom line, and an investment in building and maintaining a payment processing arm just isn’t as good as it was in the 1980s and 1990s. Back then, we didn’t have PCI DSS, and the interconnectedness of our enterprises was virtually nonexistent when you compare it to today’s IT infrastructures. When CIOs work with CFOs to truly determine the amount of money spent toward maintaining these environments versus paying a point more on each transaction to outsource, they can get a better handle on what complete outsourcing means for their company. Add to that the research done in transaction theory, which has described how per-transaction costs are declining. Each enterprise is a bit different as is each processing agreement, but you can bet that more than one executive has built additional fees into their business model for the long term as opposed to continually living with compliance costs around PCI DSS.

If you are a small business, there is absolutely no reason to build your own gateway. Small businesses really get the core competency concept: “What do we do? We make the best pizzas around. So why would we invest any money on anything not related to making the best pizzas around?” Cashless payments are a way of life, and one way we conduct cashless business is via credit cards. There are many other methods that are cheaper per transaction, and as these new technologies incubate, every Chief Information Officer (CIO) and Chief Financial Officer (CFO) should look to see if incorporating them makes a significant impact on the bottom line. Until then, outsourcing payment processing is easily one of the best decisions you can make around PCI DSS.

Now, what if you choose not to outsource? There are many options for you to reduce your scope. The first of which is to investigate tokenization. In the purest sense of the word, a token is a replacement value for another piece of data. Meaning, instead of using 4111 1111 1111 1111 for a Visa card number, you would use some other value to represent that card number, and have a way to look up the original number should you need it. The token could be something alpha-numeric, numeric only, or even binary values. Based on the amount of existing data and the design of the applications using it, most tokens tend to take the form of a 16-digit numeric value.

Regardless of the makeup of the token, there should not be any mathematical (or otherwise) relationship between the token value and the original value. The only relationship that should exist between a token and the original value is the index table of numbers you would use to associate a worthless token with a potentially valuable PAN.

NOTE

A schema for such an index might look like this:

```
CREATE TABLE Tokens {  
    original_value CHAR(16) PRIMARY KEY,  
    token CHAR(16)  
};
```

This is a rather simplistic view, but tokens don't need to be complex to be effective. Generating tokens would happen outside of the database layer in this case, but you could build that generation process into the database layer. Making the original value the primary key will prevent two tokens from representing the same original value. Keep in mind that the original value should be encrypted. It's been a long time since either author has done database design, so we wouldn't suggest implementing this directly. This is simply a way to illustrate a point.

Original values should not be able to be reversed or derived from token values. If they are, tokens should be treated like cipher text instead of tokens. When tokens and PANs are cryptographically related it opens the door for cryptanalysis and the potential to reverse the crypto operations.

Another concept is to look at how you process information, and choose a highly centralized and protected model for doing so. One method for doing this would be to centralize all of your data into a single enclave and only provide access to the applications and data through a “window-pane” powered by a virtualization technology like VMWare View or Citrix. Several companies have taken this approach to keep their sensitive data centralized, and put a virtual air-gap between the user and the data.

In these instances, data is tightly controlled in a small environment and all interactions are done through a virtual desktop that acts like an abstraction layer. In most instances, companies will treat this as the true PCI DSS perimeter, so any user accessing the environment will typically use a 2-factor token (or some other form of 2-factor authentication) to meet Requirement 8.3. Any traffic to and from this environment is tightly controlled via firewalls, access control choke points, other access and authorization management tools, and network monitoring including technologies like DLP. PCI DSS does not require DLP directly, however, companies use the tools to ensure their scope stays where they expect it, and to create an early warning system to alert administrators to potential problems before a breach occurs.

Reducing the scope of your PCI environment is a business decision that should be included in every journey to compliance. More often than not, simply removing the data from the environment and making it someone else's problem can go a long way to minimizing the issues most companies face when complying with PCI DSS. The main goal for minimizing the impact is to use people, process, and technology

to contain the spread of cardholder data. The above methods are a few examples, but the general methodology you would go through is

1. Understand how your business uses credit card information (useful for the Executive Summary section of the Report on Compliance).
2. Understand the business and legal requirements for retention of data (Requirement 3.1).
3. Completely map the flow of cardholder data throughout the entire enterprise (this is much more than what is required for Requirement 1.1.2 and 1.1.3, and should include the business process flows that can map to the technology endpoints).
4. Now that you have the scope of the problem (see what we did there?), look for ways to reduce or remove cardholder data by changing business processes and isolating technology segments.
5. Create remediation plans for areas where you cannot remove cardholder data that would include budgetary requirements for maintaining the data, costs for removing the data, soft costs associated with the long-term management, and a 3–5 years total cost projection.
6. Approach finance teams and business leaders to explain the available options and get buy in from the C-level to execute a plan.
7. Execute the plan.

The authors have used this methodology many times to help companies reduce their compliance costs and affect change in the business to ensure a workable long-term solution.

PLANNING YOUR PCI PROJECT

If you are reading and working your project at the same time, you now have a really good idea of how serious the problem is, you have a solid list of projects for your company to complete during the journey, and you have executive buy-in to proceed. But what order should you go in? And how do you take a loose grouping of projects and demonstrate measured progress toward compliance? Luckily, the Council has something ready for you to use.

NOTE

The PCI Council created a tool called the Prioritized Approach for PCI DSS which can be downloaded from the PCI Standards & Documents section of the PCI Security Standards Web site. There is both a PDF version and a spreadsheet version that includes graphs and completion estimates for customization to your organization. Keep in mind, that this is not a one-size-fits-all type of project. You will want to customize their milestones for your organization.

The Prioritized Approach for PCI DSS details a Council-endorsed roadmap for becoming compliant that goes through six key phases defined as:

1. Remove sensitive authentication data and limit data retention. This milestone targets a key area of risk for entities that have been compromised. Remember—if sensitive authentication data and other cardholder data are not stored, the effects of a compromise will be greatly reduced. If you don't need it, don't store it.
2. Protect systems and networks, and be prepared to respond to a system breach. This milestone targets controls for points of access to most compromises, and the processes for responding.
3. Secure payment card applications. This milestone targets controls for applications, application processes, and application servers. Weaknesses in these areas offer easy prey for compromising systems and obtaining access to cardholder data.
4. Monitor and control access to your systems. Controls for this milestone allow you to detect the who, what, when, and how concerning who is accessing your network and cardholder data environment.
5. Protect stored cardholder data. For those organizations that have analyzed their business processes and determined that they must store Primary Account Numbers, Milestone Five targets key protections mechanisms for that stored data.
6. Finalize remaining compliance efforts, and ensure all controls are in place. The intent of Milestone Six is to complete PCI DSS requirements and finalize all remaining related policies, procedures, and processes needed to protect the cardholder data environment.

Each requirement is broken into its various subrequirements and assigned to one of the six phases. For the large number of you reading this that validate compliance via a Self-Assessment Questionnaire (SAQ), you will need to do some editing as the tools from the Council are not broken into the various versions of the SAQ. Regardless of your level and validation requirements, you should run through the entire standard at least once to see if you are missing any of the 250+ tests. It's better to identify and remediate issues now so that as your business grows or as the standard changes, so you are not caught with a massive remediation bill down the road. Once you complete your initial gap analysis for the requirements you must validate against, you should see how your project list lines up with the six phases. This is where the spreadsheet tool can be a huge help for your project as you can adjust the requirements to fit your projects, even if they don't exactly line up with the phases as defined by the Council. The closer you leave it to the predefined phases the better when talking to your acquirer or processor about your progress to compliance. They will most likely be familiar with this document, and in some geographies like Europe, you may be expected to report compliance to a certain phase to receive certain exceptions. More on this in Chapter 17.

The biggest challenge you will face while remediating PCI DSS issues is during the execution of your project. You will invariably have one or two teams that cannot execute to the original remediation plan due to some unforeseen issue. This is

where flexibility, knowledge, and experience really pay off in your organization. If you don't have resources on-hand to quickly assess and adjust the plans with deep knowledge and experience in PCI DSS, you should consider augmenting your staff with a contract resource. Not all contract resources are alike, and each one should be interviewed like you were going to hire them. It helps if you have been through some formal training on PCI DSS such as the Internal Security Assessor program or even the PCI Professional offered through the Council. You will not only be able to handle most of the minor compliance issues yourself, but you will know what kinds of questions to ask a prospective contractor to see if they are a good fit and worth their price tag. Unfortunately, these types of issues that pop up don't have a basic formula to solve. This is where the real magic happens during your PCI journey.

For those of you who are fans of classic management tools that can help bring clarity to complex situations like PCI DSS, head over to your nearest library and grab this seminal article from 1995 as published in *Sloan Management Review*. Paul J. H. Shoemaker penned an article called "Scenario Planning: A Tool for Strategic Thinking" (ISSN: 0019848X) that lays out a process for helping management strategically decide the best path forward. Be prepared to have your assumptions challenged at the highest levels as your scenarios should include things like 100% outsourcing given the nature of the market.

CASE STUDY

The case studies presented in this section build upon what we have learned so far in this chapter. The first will take you through a company's quest to fully understand their data sprawl problem and the second through a company looking to reduce PCI DSS scope with business leaders that are fighting change.

THE CASE OF THE LEAKY DATA

Tracey's Tin Trimmings is just beginning their PCI project with Daniel at the helm. Daniel works for a regional retailer with 11 locations specializing the sale and service of high quality aftermarket automotive products. In order to compete with the larger big box retailers, Tracey has created a highly customized shopping and delivery experience that values customer service and satisfaction above all else. They store extensive information on their customers in their corporate data center, as well as with several third-party providers that enable customers to watch repairs and modifications through a browser, deliver customized information to customers about their vehicles, and interactive applications that allow users to scan product codes and chat with a live expert on the integration with their vehicle.

Since Tracey's was founded on a shoestring budget 4 years ago, the majority of the innovative customer interaction systems are cloud-based or delivered by third parties. Part of Tracey's secret sauce is the ability to run analytics over all of the disparate sources of information to ensure their customers receive timely updates

enticing them to spend money. Daniel is a relatively new employee, now with the company for 18 months, and is in charge of starting the PCI DSS compliance process as credit card volume skyrockets. He first talks to all of the business owners of the various divisions inside of Tracey's. Daniel knows that if he doesn't have a good working knowledge of all of the service providers and data interchange points, he won't be able to properly scope the PCI environment. As he learns how the business operates, he realizes that like most small companies, the early architects at Tracey's favored utility over security and privacy, and while customer information is fairly well protected, it has no bounds by which it moves.

Daniel maps out the business processes describing how things should work inside of Tracey's systems. He then goes about validating the business process documentation by putting network sniffing technology at key choke points, and working with some of the third-party providers to discover the kinds of data that Tracey's uses as well as has access to. Once Daniel has that information, he updates the business process documentation to take the real-world happenings into account with the architect's vision.

Now that Daniel has a true picture of what is happening inside Tracey's, he realizes that major adjustments need to be made to the way information is processed to keep the scope manageable. As it stands today, everything is in scope because there is no real separation among the various systems and functions, and a full-scale remediation is neither affordable nor doable from a timetable perspective. He also opens discussions with his service providers to understand their compliance status as well as revisit their contracts to ensure compliance with PCI Requirement 12.8. During this entire process, he limits the scope to two sets of systems, and two third parties. He is able to set up encrypted tunnels between the providers and the systems, and segment those systems from the rest of the network with firewalls. With the smaller scope defined, Daniel now looks to perform his gap analysis, plan his compliance project for the next 12 months, and investigate tools to ensure he can automatically enforce the scope definition.

THE CASE OF THE ENTRENCHED ENTERPRISE

Jason's Jump-Up, a large fitness chain targeting family health and nutrition, has grown by acquisition by merging with several regional gyms with similar cultures and customers. Jason started his business 15 years ago in Atlanta and is now a major shareholder in the larger enterprise that spans from Texas to Virginia. The board hired a new CEO, Chief Operations Officer (COO), and CFO to handle the larger enterprise as it plans to Initial Public Offering (IPO) in 24 months to raise capital for a westward expansion. For the most part, the management staff from each of the acquired companies stayed in place and each is run as a separate division with its own Profit & Loss (P&L) accountability. As the new executive management comes together, they realize that a massive overhaul in the corporate structure is necessary to sustain a larger organization, and Jason is charged with streamlining business processes across the enterprise and getting buy-in from all the divisional managers.

Jason knows how his original 18 locations operated, but is unfamiliar with all the inner workings of the other various companies that merged into the fold. As he visits

with each manager, he learns that not only are things very different from division to division, but the managers are quite set in their ways and have an aversion to change.

After Jason meets with everyone, he puts together several strategies for moving forward, one of which includes partnering with a third-party payment processor to manage all of the monthly membership dues and daily incidental fees that members incur by using certain amenities inside the clubs. The vast majority of the divisional managers fight the outsourcing proposal because they realize that the added fees will ultimately be charged to their divisional P&L, and they will receive lower revenues.

In order for Jason to sell his plan, he needed to get creative. He knew that processing payments was not something he wanted the company to focus on as it took away from the core competency of health and fitness. On his initial run through the divisions, he brought a few consultants with him to analyze business processes, financials, and review the payment systems for PCI compliance. Since he had detailed information on the gaps in each environment, he was able to work with a consultant to get an approximate remediation cost and ongoing maintenance costs once the gaps were remediated. Each division's cost projections over the next 5 years were well into seven figures, with nearly 60% taken in the first 2 years. Jason went to each of the divisional managers and showed them the cost projections. He informed each of the divisions that these costs would be hitting their P&L, or they could opt for the much lower operating costs of outsourcing per his original plan. Jason knew that the outsourcing plan made the most financial sense for the larger company in the long term, but he put the decision to each manager to make. Overwhelmingly, the managers opted to go with the outsourced payments model, and ultimately remove some IT and operating expense from their P&L as they spun down systems responsible for processing payments.

The key to Jason's success was not only doing all of the diligence required to paint the picture accurately, but by providing complete alternatives with future cost projections while involving the managers in the decision. Each manager knew that Jason would be taking the overall analysis to the board, and that unprofitable divisions would not fall into good favor.

SUMMARY

Determining the correct scope for your PCI environment is the single most critical thing you must get right while planning anything related to PCI DSS in your company. While the officially chartered SIG didn't create the panacea for all scoping ills, it did produce quite a bit of content that is useful in both determining the scope and providing guidance when using more advanced technologies like EMV. The Council provided us with the Prioritized Approach for PCI DSS, which can be tremendously useful in planning and executing our PCI-related projects. But no amount of tools will compensate for a lack of support from the C-suite. If they don't believe that compliance is important, and reducing their exposure to compliance is equally important, you will learn what it's like to push a large rock up a very steep hill.

Building and maintaining a secure network

5

INFORMATION IN THIS CHAPTER:

- Which PCI DSS requirements are in this domain?
- What else can you do to be secure?
- Tools and best practices
- Common mistakes and pitfalls
- Case study

The concepts of defense-in-depth and layered security best represent the idea of building and maintaining a secure network. It would be great if organizations could rely on one type of technology or a single device to provide all of our security but that's not realistic, whether now or in the future, as history proves there can be no "silver bullets" in information security. Some professionals use the analogy that security is like an onion—it has layers. Alone, each layer might be weak and translucent, but together they're tough and solid.

A firewall is one layer, but not necessarily even the first layer and definitely not the last one. [Figure 5.1](#) shows examples of different layers. The packet-filtering router that connects your company to the Internet may be the first layer, or there could be layers even further upstream at the Internet Service Provider (ISP)—such as for Denial of Service protection (which is not mandated by Payment Card Industry Data Security Standard (PCI DSS), of course—if your Web site is down due to an attack, it is pretty hard to steal the cards from it, after all). There you might configure a small rule set to filter out basic unwanted traffic like Internet Control Message Protocol, telnet, other outdated protocols and anything else that you can (and, in fact, must) live without crossing into your network space.

The next layer contains the devices that make up your internal network infrastructure. Firewalls, intrusion prevention systems (IPSs), and even switches with security functionality all contribute to this layer of security. Some organizations also choose to deploy a Web Application Firewall (WAF) that is mentioned in PCI DSS Requirement 6 as one possible choice for Web application security.

Next is the host-based security that you might have installed on specific machines. Host-based intrusion detection and prevention, antimalware software, application control and "whitelisting," and other protective controls may include hardening of the operating system itself.

The next layer covers the application. Any hardening of the application, access controls, and file or library permissions fall into this layer.

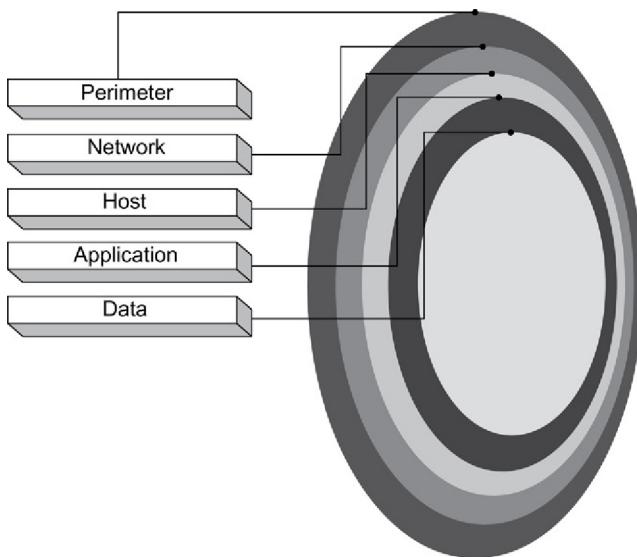


FIGURE 5.1 Layers of Security

The final layer covers protecting the data. Encrypting the data stored on the system is one of the effective ways to protect it, provided that you do need to keep that data (!). As we say elsewhere in the book, don't encrypt if you can delete!

WHICH PCI DSS REQUIREMENTS ARE IN THIS DOMAIN?

This chapter assumes that the reader has a working knowledge of firewalls and network security technologies as well as network segmentation basics. Luckily, these devices have been documented in papers and books and explored by bloggers all over the Internet. If you find yourself lost on a certain term, try punching it into a search engine! From time to time, we'll refer back to the Payment Card Industry (PCI) Self-Assessment Questionnaire (SAQ) and other PCI Data Security Standard (DSS) documents for clarification.

There are literally dozens of firewall manufacturers in the world but only a few different types of firewalls. The PCI DSS v3 standard does not specify what brand of firewall to use, but it does dictate functionality it needs to provide.

There are two main requirements that make up this domain:

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

ESTABLISH FIREWALL CONFIGURATION STANDARDS

Requirement 1.1 of the PCI DSS guides you through the process of configuring and maintaining your firewalls and routers. There is no real science to deciding on what type of device and configuration to use, just some forethought. PCI requirements make it easy for you by telling you what type of firewall to use, how it must be configured, how to maintain it, and what to protect against.

PCI DSS doesn't waste any time getting into change management. Requirement 1.1.1 details "a formal process for approving and testing all network connections and changes to the firewall and router configurations" (here and elsewhere in the book, quotes with no source are from the latest version of PCI DSS standard, v3 at the time of this writing). There are a couple of things going on here.

First, all external connections must be approved and tested. Approval implies that management, or relevant delegates, must know and agree to the connection.

Second, once the connection is made, it must be tested. Testing can range from full-on analysis to simple port scans to verify that the port is opened as intended and designed. If this is your first time through PCI DSS, the firewall should be baselined (as a best practice), and any changes to the configuration thereafter must be approved. Each stakeholder should have a say in whether or not the changes actually get implemented.

It's much easier to understand what needs to be secured if you can see it on paper. Requirement 1.1.2 asks for a current network diagram: "Current network diagram that identifies all connections between the cardholder data environment and other networks, including any wireless networks." This diagram needs to remain current at all times. With a good change management process, keeping the network diagram up-to-date is a simple task.

Moreover, a Requirement 1.1.3 explicitly asks for a data flow diagram, not just a network topology map: "Current diagram that shows all cardholder data flows across systems and networks." The flow of data throughout the enterprise is often forgotten when putting together an accurate diagram. In fact, network diagrams in general are poor canvases for graphically depicting the flow of data. Most diagrams are created in Visio® or a similar tool. Although the tools are quite powerful in what they can do, most engineers think in different ways, thus creating different looking diagrams that may represent the same underlying theme. A method for creating data flows published in an article entitled *Data Flows Made Easy* illustrates a way you can simplify this process (www.brandenwilliams.com/brwpubs/DataFlowsMadeEasy.pdf). These flow matrices are useful to Qualified Security Assessors (QSAs) and are much easier to maintain than graphical depictions of data flows.

All ports and services allowed through the firewall must be documented—secure or not—and insecure protocols must have additional documentation about their use, business justification, and potentially a risk assessment performed against them in the environment per Requirement 1.1.6. Requirement 1.1.5 may help us meet this requirement as you need accurate documentation of all groups, roles, and responsibilities for logical management of network components. This is especially helpful when implementing the rule sets.

WARNING

The only services, ports, and protocols that are allowed are those that are required for business purposes. These must be secured and documented appropriately. A simple way to document these is to add the justification to the configuration in a comment field for each rule. More formal methods of documenting these items could be through an assembly of change control tickets, a firewall rule set review by a third party qualified to perform such a review (which could be an engineer or a third-party provider), or a formal corporate standards or requirements document. Remember that all documentation must be cross-checked against the current firewall rule set, thus making documentation inline the most efficient way to handle this. Documentation is key, and if someone says “I need that port open because I said so,” a QSA will dig deeper.

Now that you have all your Internet connections documented and your network and data flows clearly defined, Requirement 1.1.4 states that firewalls need to be implemented at each Internet connection point and between any De-Militarized Zone (DMZ) and the internal network. Finally, Requirement 1.1.7 mandates 6-month firewall and router rule set reviews. Be sure to take your review process seriously.

Going through the motions on any of the periodic PCI DSS maintenance tasks will quickly put you on track for a breach. Your engineers should scrutinize every rule and ensure it needs to be there. One good way to check a rule’s use is add a command to log a message every time it is used. If you find rules that are listed in your policy but never actually used during a 6-month period, chances are the rule is out of date and should be removed.

Denying traffic from untrusted networks and hosts

Confidentiality and integrity of cardholder data (note that PCI DSS is not at all concerned with the availability of such data—that is typically your problem, not your QSA’s and PCI DSS in general) are at the heart of Requirement 1.2. Your firewall configuration has to accomplish several things. The rule of thumb here is to deny nearly all traffic. Only the minimum traffic truly required to conduct business should be allowed through the firewall—both inbound and outbound. It is much easier—and much safer—to filter everything initially and only open the required ports and protocols on those ports leading into the cardholder data environment and toward PCI-relevant systems. This is where a good network diagram with data flows (see Requirement 1.1.3 on card data flows), coupled with an accurate list of required services, ports, and protocols (with business justification), is worth its weight in gold.

Denying all traffic from “untrusted” networks and hosts is easy to conceptualize, but not always easy to accomplish in a complex and distributed environment. Many firewall solutions do this right out of the box with some degree of effectiveness. If not, there is usually a rule that can be configured to do this. It all boils down to failing safe. After processing all the traffic permitting rules in the firewall policy, all firewalls should deny everything else. For example, a common deny—all rule is called the “cleanup rule” and should be placed as the last rule in the list, whereby any traffic not matching a rule above it is automatically dropped. Another common deny—all rule is called the “stealth rule,” whereby all traffic, inbound or outbound,

specifically targeting the firewall device itself is dropped. Of course, before either of those rules goes into play, you must have a rule that allows an administrator to access the management function of the firewall to make changes.

With the traffic being denied for all inbound and outbound traffic, specific rules need to be applied to enable your business to function. Verify the business need against your list of ports, protocols, and services first. To add even more security, if the source of traffic can be narrowed down to specific networks or hosts, make your rule stricter and only allow those through (this approach will add to both security and your firewall management tasks!). In order to pass the requirements listed under 1.2, you must lock down access to and from the cardholder data environment to only what is necessary and deny all else Requirement (1.2.1). You must also verify that the configurations are secured and start-up and running configurations match Requirement (1.2.2), and install firewalls between any wireless networks and cardholder data networks Requirement (1.2.3). But what if your cardholder network IS wireless? See Chapter 8 for more info.

Restricting connections

Requirement 1.3 of PCI DSS gets pretty granular with restricting connections between publicly accessible servers and any system component in scope for PCI. What does this mean to you? The database containing cardholder data cannot be in a DMZ that is publicly accessible. Stateful inspection firewalls must be used. You should never allow spoofing to occur. If traffic is not explicitly allowed in the rule set, it should be denied. Any Request For Comment (RFC) 1918 addresses are not allowed from the Internet, and Internet Protocol (IP) Masquerading should be used where appropriate with Network Address Translation or Port Address Translation such that those IPs cannot pass from the Internet into the DMZ and the internal IP addressing scheme is not exposed.

NOTE

The PCI DSS states in Section 1.3.6 that the firewall solution must provide stateful inspection. Most commercial and open-source firewalls have expanded beyond basic port blocking techniques and have stateful inspection capabilities. Cisco provides this capability on top of basic access lists (ACLs) in a feature new to IOS 12.0 called Reflexive Access Lists (or RACLs), which can be useful when extending firewall capabilities to satellite locations such as retail locations and distribution centers in nonfirewall specific equipment.

RFC 1918, originally submitted in February 1996, addresses two major challenges with the Internet. One is the concern within the Internet community that all the globally unique address space (routable IP addresses) will be exhausted—and as of this publication it has already happened. Additionally, routing overhead could grow beyond the capabilities of the ISPs because of the sheer numbers of small blocks announced to core Internet routers. The term “private network” is a network that uses the RFC 1918 IP address space. Companies can allocate addresses from this address space for their internal systems. This alleviates the need for assigning a globally routable IP

address for every computer, printer, and other device that an organization uses, and this provides an easy way for these devices to remain sheltered from the Internet.

TOOLS

RFC 1918 space is often quoted and misunderstood. According to the original RFC, which can be downloaded at www.faqs.org/rfcs/rfc1918.html, there are three blocks of IP addresses that are considered private and nonroutable over the Internet. Those are 10.0.0.0–10.255.255.255 (10.0.0.0/8), 172.16.0.0–172.31.255.255 (172.16.0.0/12), and 192.168.0.0–192.168.255.255 (192.168.0.0/16). Any private networks in your corporation should be numbered within those allocations, or in rare cases, on non-RFC 1918 space that is owned by the company and not advertised to the Internet. This can be dangerous, however, as a fat-fingered change could cause the space to be publicly routable. Avoid using IP space that is publicly routable but does not belong to you as it can be very dangerous.

Parts of Requirement 1.3.4 and 1.3.8 dictates preventing internal address space from accessing passing through choke points to the DMZ or internal network addresses such that they can exist on both sides of an interface. Some devices call this “Anti-Spoofing” technology, mainly because an old trick to get around firewalls is to spoof internal IP addresses from external hosts. Internal addresses originating from the external side trying to come in to the DMZ or internal network should raise a red flag in the logs for the device. The firewall rule set should only allow valid Internet traffic access to the DMZ, and vice versa. Requirements 1.3.2–1.3.3 add more color on restricting traffic from the Internet to only those addresses that are in the DMZ and restricting direct inbound routes from untrusted networks into the cardholder environment.

Why can’t Internet traffic pass to the internal network? Because Requirement 1.3.7 requires the in-scope database to be on the internal network segregated from the DMZ. The cardholder database should never be able to connect directly to the Internet. Front-end servers or services should only be accessible by the public. These servers and services access the database and return the required information on behalf of the requester just like a proxy. This prevents direct access to the database.

In recent years, some organization want to use cloud-based (really, SaaS-based) solutions instead of traditional software. For example, Office 365 or Google Applications may replace some of the common office software. However, if the desktop is located inside the CDE, using such applications may require direct access to the Internet, thus violating PCI DSS compliance.

WARNING

There is no reason whatsoever to allow a database or other application to be directly accessed from the Internet. Along the same line, there is no reason to allow a database or other application to directly access the Internet, by passing the DMZ. This could cause cardholder information to be vulnerable to unauthorized access. It is just as risky to allow a database server to have two network interfaces: one on DMZ and one on the internal network, even if no actual routing takes place. Multihomed servers effectively remove the security that is designed to be effective with a DMZ entirely. Can it be done? Sure. But it is another compromise point that wouldn’t normally exist.

Personal firewalls

Finally, rounding out Requirement 1, Requirement 1.4 mandates personal firewall software on devices that are used to access the organization's network. The devices in question can be employee-owned (maybe a home PC with a VPN Client on it), mobile (such as a laptop, tablet, or a smartphone with Wi-Fi), or both. The firewall must be present on the device, must be active, and cannot be disabled by the user. The built-in Windows Firewall can be used on Windows systems, provided that an appropriate group policy removes the capability to disable it. Compensating controls can be used for devices that lack such functionality (e.g., if a device can never have an open port, it probably does not need a firewall). In addition, many companies are moving to sanitized Web services, or other network segregation, to allow mobile devices to do basic things like e-mail and accessing some internal applications. You may find that you can meet this requirement simply with some architecture and firewall work.

Other considerations for Requirement 1

As of the last version of PCI DSS (currently in 3.0) the Council added more granularity to the requirements around routers, specifically taking Requirements 1.1–1.3 and extending them to routers. The one requirement that seemed specifically targeted at Cisco routers and firewalls has been enhanced is Requirement 1.2.2, even though it specifically only mentions routers. If any network device in scope for PCI has the capability to have a different running and startup configuration, this requirement applies and you need something to check to make sure they are actually in sync. No changes should be made to the running configuration without first going through the appropriate change management procedures.

Additional firewall considerations should be taken with regard to wireless networks and mobile or personal computers. Systems with cardholder information must be segregated from wireless networks for Requirement 1.2.3, and those firewall rules limited only to what is necessary for business. Chapter 8 has more information for you on how to get your wired and wireless networks working securely. These systems may not always get critical patches in a timely manner, and the personal firewall provides some assurance.

The oddball Requirement 11.4

Requirement 11.4, although not grouped in with Requirements 1 or 2, is part of building and maintaining a secure network.

IDSs detect unwanted activity on networks and systems, mainly from the Internet, but increasingly on hosts (host-based intrusion detection) and Wi-Fi networks (wireless intrusion detection). This activity is usually the product of a hacker probing or executing an attack. IDS can detect malicious activity not normally prevented by firewalls including Trojan horses, worms, viruses, attacks against vulnerable services, unauthorized logins, escalation of privileges, and attacks on applications while IPS will be equipped to block it (most commercially available devices today are network-based intrusion prevention system [NIPS], not mere network

intrusion detection system [NIDS], but they can certainly function as a NIDS with no blocking).

In addition, the organization must “keep all intrusion-detection and prevention engines, baselines, and signatures up to date.”

There are many types of intrusion detection or prevention systems that can be used to satisfy this requirement. This is an example of a requirement that companies can leverage as a component of a solid intelligence feed for their network and produce real-time threat analysis data that can be exported to various risk management software. Below, you will find many types of IDSs that could be used to demonstrate compliance with PCI DSS. For those that you are unfamiliar with, try a couple of Internet searches for updated information on these technologies.

- NIDS is an independent platform that examines network traffic patterns to identify intrusions for an entire network. It needs to be placed at a choke point where all traffic traverses. A good location for this is in the DMZ.
- Host-based intrusion detection system (HIDS) analyzes system state, system calls, file-system modifications, application logs, and other system activity. Modern application whitelisting tools are an evolution of a classic HIDS/host-based intrusion prevention system (HIPS).

In most networks, an IDS is placed in one of three configurations:

- Network Test Access Port (TAP) allows passive monitoring on a network segment. TAPs are more reliable than hubs or switches and relatively inexpensive to implement. Hubs have a potential for bottlenecks and packet collisions. Switches can also cause bottlenecks depending on the amount of traffic being mirrored to the SPAN port and have a tendency to not receive error packets. Handling virtual local area network (VLAN) can be complex or impossible.
- HIPS solutions protect workstations and servers through software that resides on the system. It catches suspect activity on the system and then either allows or disallows the event to happen, depending on the rules. Finally, it can also monitor data requests and read or write attempts and network connection attempts, potentially allowing it to be used as a compensating control for other requirements.
- NIPS is a network security solution, while HIPS protects hosts. It monitors all network traffic for suspect activity and either allows or disallows the traffic to pass. For a NIPS to work properly, it needs to be positioned in-line on the network segment so that all traffic traverses through the NIPS. The implementation of a NIPS is similar to a NIDS with one exception: because a NIPS has two NICs, a network TAP, switch, or hub is not required. The network only needs to be architected with the NIPS in a position where it can monitor all the network traffic inbound and outbound. Replace the “N” with a “W” and the same rules apply, but for wireless networks.
- For high availability networks, you may want to consider using external bypass units when implementing NIPS in-line.

WARNING

If you do not tap in your IPS you may need to schedule a network outage whenever you need to swap an appliance. Appliances tend to fail more often than taps. Depending on where the device lives in the network, swapping an appliance can cost much more time than expected, which can affect your PCI project downstream. External bypass units can be used to facilitate swapping appliances without having to schedule an outage. This assumes that your IPS is implemented in pairs to allow failing over as you swap out the appliances.

NOTE

An IDS is a detective—not a preventive—control in nature. It only monitors and sends alerts of suspect activity. In contrast, an IPS will not only alert but can also take action to mitigate the problem. So, if the functionality of an IPS to take corrective actions is not required, why spend the money to implement an IPS? The answer to this stems from the concept of acceptable risk. An IPS solution provides the capability for corrective actions to be taken before a system administrator has the opportunity to respond, which can be desirable during an active attack against systems. Without human intervention, it is possible to cause a Type I error (or false positive) and block legitimate traffic from legitimate customers. Certain types of attack are clearly articulated and can easily be effectively blocked with an IPS. Some network IPS devices also now have “DLP Light” features that can also look for and specifically block PANs leaving the CDE automatically.

Again, PCI DSS does not dictate which solution should be used. In many cases, this may come down to cost—cost to purchase and maintain.

It is critical to realize that the QSA may “interview responsible personnel to confirm intrusion-detection and/or intrusion-prevention techniques alert personnel of suspected compromises.” Thus, it is not only about buying a tool (whether NIDS or NIPS), but about having personnel that can (and will!) effectively operate and run it! Recent retail payment data breaches, such as the massive Target breach of 2013, revealed examples where the organization has the technology, but did not pay attention to alerts until it was way too late....

PCI DSS v3 does explicitly say that “Daily review of security events—for example, notifications or alerts that identify suspicious or anomalous activities—as well as logs from critical system components, and logs from systems that perform security functions, such as firewalls, IDS/IPS, file-integrity monitoring (FIM) systems, etc. is necessary to identify potential issues.”

Requirement 2: defaults and other security parameters

A lot of thought goes into securing a network. You have to think not only about the network devices (e.g., routers, firewalls, NIPS, WAF) but also about system defaults, configuration management, and encrypting nonconsole administrative access, to name a few.

Today PCI DSS makes it easier for your organization to do it! The new (in PCI DSS 3.0) Requirement 2.5 makes the organization “ensure that security policies and operational procedures for managing vendor defaults and other security parameters

are documented, in use, and known to all affected parties.” Compliance has to be achieved—and then operationalized!

Default passwords

Default passwords exist with almost every operating system and application. Requirement 2.1 states that all vendor-supplied passwords must be changed before deploying a system on the network. Requirement 2.1.1 imposes the same mandate for wireless environments. Password policies and procedures are usually dictated by the organization. Although there are several alternatives for authentication like biometrics, smart cards, and tokens, most of us use the traditional user ID and password.

Additionally, if your organization has a procedure for adding new users and granting them access to systems, there may be some default passwords that you haven’t thought about. If you can remember back to when you first received your user ID and password, you might recall that it was a preset generic password (does Password123 or Welcome1 sound familiar?). Before PCI DSS required otherwise, many system administrators used the same generic password for all new users. If your company has not changed its new user process globally to reflect the more stringent requirements for users with access to cardholder data, you may end up with some users that have generic passwords. For more information on this, see Chapter 6.

Simple network management protocol defaults

Requirements 2.1 and 2.1.1 mandate all system defaults be changed before deploying a system into production. Simple Network Management Protocol (SNMP) is associated with several known vulnerabilities—specifically, versions of the protocol before version 3—and default strings can allow someone to learn nearly everything about a device and potentially change its configuration. SNMP is a good network management tool for administrators of large infrastructures, but if it is improperly configured, it can allow hackers to do significant damage on a mass scale. Make sure SNMP defaults are changed.

NOTE

The SNMP protocol has many versions. Most modern devices now support SNMPv3 that allows for individual user authentication and encryption of the SNMP channel. Avoid prior versions of the SNMP protocol.

The most basic form of early SNMP security is the community string. There is a public community string that allows read-only access to network devices, and a private community string that allows read-write access. The default values for these community strings are “public” and “private,” respectively. Remember, community strings are not unlike passwords, and any SNMP armed with those defaults can gain access to an SNMP aware network device.

WARNING

The only thing worse than having “public” as your community string is to have a “private” community string set up with no restrictions on use. This would give anyone read access to your network devices and the ability to change its configuration. A hacker can find out a lot of information about a device through SNMP.

Delete unnecessary accounts

Systems and applications come with a variety of accounts built-in. Some are system accounts, and others are administrative accounts allowing vendors to support their products. All support accounts should be disabled or deleted immediately. These accounts are essentially backdoors into your system, and if not controlled closely, they can cause a compromise to easily occur. Many recent retail data breaches involve such insecure or compromised accounts.

Thus, all guest accounts should be deleted or at least disabled. The passwords should be set to something no one knows and you should consider renaming the account if it can't be deleted. The same goes for default administrator accounts. Rename them to something inconspicuous and change the description of the account as well. It adds a layer of difficulty for an attacker looking for the account.

NOTE

Here are some common accounts to disable that are typically available on new installations with a basic password or no password at all.

- Root account on UNIX and Linux systems
- Administrator account on Windows systems
- SA account for Microsoft SQL
- qsecosfr account on AS/400
- “Enable” passwords on Cisco routers

Develop configuration standards

All organizations should adopt a baseline that is considered to be a minimally acceptable configuration for all systems. It is a key element of security to aid a security team's efforts in reducing the vulnerabilities on their systems from the minute they are deployed, thus reducing the overall security risk to the organization. Requirement 2.2 mandates that all known security weaknesses are addressed and are consistent with industry-accepted system hardening standards. If a particular vulnerability is not addressed with specific hardening techniques, workaround solutions may need to be applied to mitigate the risk. Once you have adopted a standard, the systems should be baselined to ensure all systems are built and hardened the same every time. Creating security baselines on computers and your networks is no trivial task. It takes time and effort, but the end result is priceless. A security baseline is a standard set of security settings that are established for each type of computer or network component in your organization. The baseline configuration is a “point-in-time” configuration

and should be updated regularly as new settings are applied and new security threats emerge. Your organization's security policy should drive what security features are applied to your systems. A well-defined security policy lays the foundation for security elements that must be put in place.

TOOLS

If you are not sure where to start, the National Institute of Standards and Technology provides checklists for almost all platforms in use today that are freely available on their Web site, <http://web.nvd.nist.gov/view/cvp/repository>. These need to be modified and adapted for your organization. The Center for Internet Security (CIS) (ciseecurity.org) is another great site for checklists, such as their CIS Benchmark tools for tons of common operating systems.

Implement single purpose servers

Requirement 2.2.1 mandates that critical servers provide a single service (e.g., Domain Name System [DNS], database, e-mail, Web) to the organization. All too often, organizations try to save money by hosting multiple services on the same host. Each service brings its own vulnerabilities and risks to the table and provides a hacker with multiple choices for attack. If too many services are provided by a single server, an exploited vulnerability on one service (i.e., DNS) can bring down or cause a denial of service to the entire server. The integrity of all the services and data is questionable at that point. As a rule of thumb, increasing the number of services provided by a single host degrades the overall security of the server and the organization.

This requirement is often debated in a number of areas, and you will have to use your best judgment many cases. For example, almost all Windows Active Directory Domain controllers run DNS and many serve DHCP. Does this mean that in order to comply with PCI DSS you have to separate all of these out? The authors don't believe so. This is very a common and accepted practice. While this can be a slippery slope, we suggest that DHCP and DNS are core to the functionality of Active Directory, thus should not be separated out.

NOTE

If you are running a Web server that is interacting with a database, that Web server should always reside on its own host separated from the database server by a firewall. If the environment is virtualized, the Web server and database server can physically be on the same host but should be separated as individual guests systems.

This particular requirement is hotly debated among virtualization enthusiasts as well as small businesses with limited resources and multifunction servers. The use of virtualization in conjunction with this requirement is perfectly acceptable for PCI DSS. The main trick is to remember the host operating system, or hypervisor, is in scope for PCI, but guest operating systems (or virtual machines) can be scoped out

depending on what they have access to and what they are doing. Remember, any guest host that can see into another guest host with PCI data in it may be deemed in scope. In addition, the entire management infrastructure is included in scope as it can manipulate an in scope guest. In 2011, the PCI Council released additional guidance on the secure use of virtualization (which you can access from their Web site); however, this new document is less than useful for compliance as it is loaded with editorial information that does not belong in a guidance document. Some of the information in the best practices doesn't match generally accepted best practices, and will surely confuse you and your QSA or ISA. Remember: the underlying infrastructure is always in scope. If you make that comply, you can have non-PCI guests in the same environment. Just remember, there should be controls put in place that prevent a non-PCI guest from accessing a PCI guest.

Today, PCI DSS 3.0 states in a side note to Requirement 2.2.1 that “Where virtualization technologies are in use, implement only one primary function per virtual system component.”

The other side of this is multifunction appliances for small businesses. This is a gray area without a ton of guidance. Depending on how literally you read the requirement, you could argue it as a server with multiple services is one single function; thus, it is compliant with the requirement. If you wanted to overdo it on the narrow side, then every server type service should have its own hardware or virtual machine to run on. The true answer is somewhere in between. Black box solutions are typically viewed as compliant with this requirement, where homegrown ones may not be.

The answer here is to use good sense. You probably don't need to have your cardholder database on a machine that also acts as your Primary Domain Controller and external e-mail server.

Configure system security parameters

You might think this is a “no-brainer,” but not all system administrators know exactly which services are enabled and disabled on their systems and how the system itself is secured. Requirements 2.2.2–2.2.4 describe how system administrators must handle the services that are available and running on their servers. Requirement 2.2.2 is standard system hardening whereby all unneeded services are removed, which couples nicely with Requirement 2.2.4 that mandates the same for removing unnecessary scripts, drivers, features, and more. Any service, piece of software, and operating system feature that does not have business justification for running must be disabled or removed. Understand that things like Internet Explorer may not be able to be removed, but there are other controls you can put in place to mitigate the use of the software (like network or host-based firewall rules).

WARNING

Don't forget about your network appliances and peripherals. These should also have appropriate security features applied. Yes, it even includes today's printers and copiers that are really network servers that can print or copy! Soon, with Internet of things emergence, the same would be said about air conditioners, refrigerators, thermostats, elevators, and many other devices.

Requirement 2.2.3 mandates the configuration of all system security parameters to prevent misuse. This particular control includes both a question and answer session with the system and security administrators as well as verification that common security parameter settings are included in the standard configuration. This can be accomplished by reviewing internal vulnerability scan data, as well as interacting directly with the machines. You can expect your assessor to ask things such as “What is your security knowledge?,” “How would you verify secure configurations on your particular equipment?,” and “What kinds of services would you disable immediately after installing a new server?”.

NOTE

Remember, in most cases, default installations have numerous vulnerabilities and insecure configurations. For example, even PoS systems are often deployed by irresponsible service providers with insecure remote access tools that are strictly forbidden by PCI DSS. Today’s malicious hackers know this really well and exploit such weaknesses for their nefarious gain.

A lot of these services, features, ports, protocols, and so forth were put there by the vendor, and it is well-known information that is freely available on the Internet. Even PA-DSS compliant devices still must be configured according to the vendor-provided Implementation Guide in order to be deployed in a PCI DSS compliant manner.

Encrypt nonconsole administrative access

System and network administrators, by design, have access to everything. They “own” the network and are responsible for keeping it functioning. However, some of the tools they use are a little less than secure. Many of the tools are antiquated and actually pass user IDs and/or passwords in the clear (such as a legacy telnet service). To accommodate Requirement 2.3, encryption solutions must be used for all nonconsole administrative access. Most modern platforms have open-source solutions for this such as OpenSSH as a replacement for Telnet (see www.openssh.org for more information). Mainframes usually require licensed software to enable encryption, so other compensating controls may be considered here.

When deploying OpenSSH, be mindful of encryption key management—freely accessible private keys do not add any security compared with easily guessable passwords.

NOTE

Compensating controls can be used for anything but Requirement 3.2 (you can never ever store sensitive authentication information, no matter what!), so in rare cases, you may be able to run services such as Telnet on your internal network. If you have a valid business case and have taken the appropriate steps to design and implement an acceptable compensating control, you may be able to use these services. From a security perspective, allocate resources to upgrade those systems as soon as possible. Services such as Telnet and even older rlogin/rsh allow users to easily capture sensitive data, remotely control hosts, and even modify data in flight. Virtually every maintained platform in use today has an encrypted nonconsole administrative option. For more information on compensating controls, see Chapter 16.

Still, please leave telnet and FTP in the 1980s, where they belong!

Keep the inventory!

A recent addition to PCI DSS is a Requirement 2.4. It makes sense now and it always made sense, but now it also “the law”: the organization must “Maintain an inventory of system components that are in scope for PCI DSS.” Forgotten components and systems that touch card data have been involved in data breaches and such an inventory helps root them out.

Hosting providers must protect shared hosted environment

PCI compliance goes further than just the commercial entity providing the goods and services. Far too often your favorite store is nothing more than a “store front” with no back office, just a building or a Web site that pushes goods. Typically, Web site hosting for these operations is done through a service provider. Requirement 2.6 mandates that hosting providers protect each entity’s hosted environment and their data. It also applies to today’s cloud providers (mostly IaaS and hosted private clouds) or flexible computing platforms, such as Amazon and Rackspace. In addition, your hosting provider should provide you with the details to complete Appendix A. Refer to PCI DSS for more information, but the requirements and testing procedures are well written and easily understood by someone in the business.

WHAT ELSE CAN YOU DO TO BE SECURE?

Secure networks are often dismissed as too hard to maintain. Why spend precious cycles chasing our tails with a locked-down configuration when we can get by fine without it?

Here is a dirty little secret that many professionals don’t want you to know: security and functionality don’t have to be mutually exclusive. Many messy and disorganized networks are in fact horribly insecure (hackers usually know them better than their owners, in fact!) and will be extremely costly to make PCI DSS compliant. On the contrary, when they are set up properly, companies can achieve both a substantial amount of flexibility and the speed to market with a solid security posture.

Expanding on this requirement, go back to basics when reviewing your firewall and filtering router rules. Make them start from a deny-all in both directions (yes, it means default-deny for outbound as well, very important in this day and age of pervasive malware), and then before adding any exceptions, ask yourself if this rule is really needed. Don’t just stop at the PCI DSS environment; go throughout your entire enterprise with this same methodical review. In fact, many recent breaches involved first a compromise of non-PCI networks, use system with a subsequent “lateral jump” into the payment network.

After you have your final set of rules, go back again and examine the network protocols and traffic that you are permitting. Can you change the software to use encrypted streams? Can you go a step further and force mutual authentication with SSL certificates (or some other means) between network hosts? If this is possible,

perhaps you can further limit the types of traffic permitted through your firewall. At this point, document it and be ready to provide it to your QSA.

Finally, the best thing you can do is to separate the people from the machines. No, we don't mean electrifying computer keyboards everywhere in your enterprise (but that could be a fun experiment). We mean put access controls and firewalls between your server farms and "userland." Users are creative little monkeys that learn how to take the keys from an exhausted zookeeper to let all the animals out at night. Even with a massive investment into technologies to secure laptops and desktops, all it takes is one creative act to introduce unwanted software into the environment, potentially targeted at server platforms. Separating those environments will go a long way to build resilience and security into your network.

Sadly, at this point the attackers know that such separation is not achieved in many organizations and thus one can compromise the developer personal machine, located one jump away from the crown jewels of payment processing....

TOOLS AND BEST PRACTICES

Firewall and network administration are easy when you have only one or two devices like many small merchants. Larger companies have hundreds or thousands of devices distributed over dozens of location and even countries and must use multiple tools to automate portions of the administration.

Firewalls that accept plain text configuration such as Cisco ASA, Juniper Netscreen, and even Linux IPTables can easily have scripted solutions that allow one change in one file to propagate to a virtually unlimited number of sources. If all store configurations are the same, or at least can be grouped, you can input the baseline configuration into a database and then have a script that generates the appropriate configuration for each store based on variable data such as store IP addresses, special store cases/rules, and other custom configurations.

The database structure could be as simple as three tables: a store definition table that has custom elements such as IP address space, possibly other Boolean configuration switches, a baseline store configuration that all stores should conform to, and a table for supplemental rules for local customization. Basic setup scripts could easily dump a working firewall configuration for each store, and then normal distribution methods could retrieve and install them. Adding a new device to all stores or changing a global configuration could now be done with minimal effort and cost.

Most enterprise class firewalls that use a graphical user interface to administer them, such as Checkpoint, have the capability to administer multiple enforcement points in one central location. Many of these can be scripted as well such that you could accomplish something similar to the above but through commercially available means.

Firewalls and routers can be assessed through automated tools that review their configuration and match them up against several security standards, including PCI DSS.

COMMON MISTAKES AND PITFALLS

These requirements normally bite companies in a few specific ways. The companies requiring the most remediation under this requirement typically are companies going through PCI DSS for the first time. Small companies typically have challenges with their technology implementation (and documenting it) such that their setup does not fail safe, or they didn't have the physical or human resources to build things to comply with PCI DSS.

Documentation tends to be one of the biggest deficiencies companies face when assessing against this domain. Your best bet is to make sure that you have documented all your firewall rules as required by PCI DSS. Simply going through that process will force several issues that will help you meet your end goal of compliance with PCI DSS. Those issues are outlined below.

EGRESS FILTERING

Firewall policies tend to forget that outbound traffic should not get a free pass. For firewalls to comply with PCI DSS (and be effective security devices), they must only permit traffic that is necessary for business—both inbound and outbound. To successfully enhance your firewall policies without interrupting your business, consider adding new rules to your firewall that permit certain types of traffic and log any hits to those rules. Rules in firewalls don't just have to block things, they can help you tag and categorize traffic that you allow through. This will allow you to quickly determine which rules will work and which ones will not. Remember, fresh installations should start from default-deny in both directions. If faced with a legacy configuration, try to find an opportunity to redo it on the same principle.

DOCUMENTATION

Without fail, documentation is one of the most tedious aspects of attaining and maintaining PCI compliance. Before your assessor comes on site, make sure that all in-scope firewall rules are documented and have all the necessary approvals. In fact, it is easier to maintain the documents than to create them every time before the QSA visit (this is one of the things that PCI Council now promotes as “PCI as business-as-usual”).

Don't forget that all ports and services allowed in and out must have documentation associated with them. Consider performing a risk assessment on those rules and including that documentation as well.

SYSTEM DEFAULTS

Good internal vulnerability assessment tool finds most instances of default passwords or configuration on in-scope systems. Many payment data breaches that happen today start with a default or blank password or default to an insecure configuration.

Ensure that a vulnerability management program correctly identifies these mistakes and that the management process designed to take findings through to resolution (including the all-important feedback loop!) correctly reports progress on remediation activities.

CASE STUDY

For this section, we will explore two different cases to show how the requirements can be applied in both small and large companies.

THE CASE OF THE SMALL, FLAT STORE NETWORK

Before PCI, the notion of a firewall anywhere except for the border of a network didn't exist. In fact, wasn't the old joke about security "Hey, I'm all about security! I have a firewall!?"

Unfortunately for most companies, big or small, rapid growth and pressure to meet financial expectations have stifled security such that compliance initiatives like PCI become challenging. In nearly every company, network segmentation had to be addressed at some level.

Joe's Jumping Jerky Joint, a small company given to Todd by his father Joe 10 years ago, has four stores and an e-commerce Web site that accepts credit cards for payment. Todd was notified by one of his acquirers that he is now a Level 3 merchant and must submit an SAQ to demonstrate his compliance with the PCI DSS. As a former Level 4 merchant, his knowledge of PCI DSS was limited; thus, his stores or online site have not been validated against the controls.

The physical stores use IP-based Point of Sale (POS) terminals that he purchased off of eBay, and they share infrastructure with some nonpayment related machines. There are two PCs that are used by the manager and assistant manager of each store to browse the Web, check e-mail, and access the order fulfillment screens from the online store. There is also one kiosk in the store that allows customers to sign up for e-mail updates. The stores also have a small cafe and tasting area where you can sample some of the jerky and have a light lunch or coffee. Todd provides free Wi-Fi to customers who come to the cafe.

Each store connects to the Internet via a business class Digital Subscriber Line (DSL) for his Internet service, allowing Todd to ensure certain minimum levels of bandwidth and favorable pricing when bundling other services. The business class DSL came with an upgraded router that provides the capability to create a DMZ, but he has not used this functionality to date.

Todd knows that the PCs and the kiosk are not fully up-to-date with PCI standards and that the wireless network is a problem for PCI, as there is currently no segmentation between it and the wired infrastructure. In order to meet PCI DSS, some changes must be made. Todd does not want to invest thousands of dollars into added infrastructure to comply with PCI DSS. What options are out there?

Luckily for Todd, he upgraded to the business class DSL! The DMZ functionality on the device allows Todd to segment the POS devices onto their own network with relative ease. He needs to purchase a small managed Ethernet switch to accommodate the few POS devices per store. Finally, he needs to review the configuration of his DSL router to ensure that the firewall settings are done appropriately according to PCI DSS. He will not be able to access the POS devices or the POS controller (if applicable) from the PCs on the network, so he may have to adjust some process to go to those machines directly for end of day batch processing.

For the Web site, Todd must work with his hosting facility to ensure that they are providing a PCI compliant solution. He should either ask for their completed Appendix A for his environment (to meet PCI Requirement 2.4 if applicable) or see by checking if they are on the CISPA Compliant Service Provider list (www.visa.com/cisp). Regardless, his contract with his hosting company should comply with Requirement 12.8. See Chapter 7 for more information. If they are a compliant hosting facility or service provider, he knows he can provide documentation to satisfy his internal assessment requirements for demonstrating compliance to PCI DSS. If not, he must work with them to ensure they take the appropriate steps to become compliant. If he decided to in source his Web site, he would need to document all his firewall rules and make sure he had sufficient ingress and egress filtering. Oftentimes, firewalls will default to a minimal inbound rule set without restricting outbound traffic at all.

For examples of what Todd's network looked like before and after segmentation, refer to Figures 16.1 and 16.2 in Chapter 16.

THE CASE OF THE LARGE, FLAT CORPORATE NETWORK

Flat networks don't only appear at small retailers' store locations, they appear in the corporate offices too—oftentimes with much higher remediation costs.

Consider the case of Christine's Car Commissary, a large retailer with 2000 stores. Christine's company has recently become a Level 1 merchant and is facing fines of \$25,000 per month under the VISA Compliance Acceleration Program. She hired a QSA, and among other findings, she discovers that her internal assessors have underestimated the scope of PCI due to their flat corporate network. The store locations have large enough IT installations to make segmentation as easy (if not easier) as James's Jumping Jerky Joint. Instead, she is faced with massive costs associated with upgrading legacy systems not involved in card processing on her corporate network, and many of which are no longer maintained and cannot meet PCI DSS.

Christine knows that she needs to get as many systems out of scope as possible to keep her remediation costs under control. Two years ago, Christine had to upgrade several of the core switches that run her corporate infrastructure simply due to capacity limitations. She smartly purchased for growth and has both CPU cycles and bandwidth to spare. Her IT staff have several VLANs defined in the core switching infrastructure, and with the recent upgrades, they have the ability to place ACLs on some of the switching interfaces (or in some cases, directly on VLANs).

Christine must quickly deploy ACLs to isolate the cardholder environment such that her legacy computing systems are not included in the scope of the assessment. After consulting with her vendors and IT staff, she decides to take a two-prong approach. Several of her distribution switches have empty slots available. She will purchase firewall blades to boost the security and efficiency of her switching network, allowing her to accomplish several things.

1. The cardholder environment will be segmented from the rest of the core network, thus significantly reducing the scope of the PCI assessment, saving both remediation and assessment costs. She uses her new firewall blades to handle this at the network core.
2. IT and Management staff requiring access to those systems (both internally and remotely) are provided two-factor authentication tokens and have VPN software installed on their laptops. When they are in the office, they must use that two-factor token to access the environment just like if they were at home. This can effectively change the perimeter of her corporate network (as it relates to PCI), thus further reducing scope.
3. Like Todd does in the previous example, Christine creates segmented areas inside her store locations. Christine accomplishes this differently but with the same level of effectiveness. She directs her IT staff to place RACLs in the stores to segment her POS environment from the administrative and wireless areas in the stores.
4. She also puts additional controls on the wireless network with more stringent RACLs and deploys wireless intrusion prevention systems to further bolster the security around her wireless network.
5. Finally, Christine has her security staff review the overall architecture of her network and design additional enclaves to boost the security of the network and increase its overall resistance to worms, viruses, and other malware that propagates via weak network access controls.

Christine is able to focus her IT and Security staff on the above five items, saving both time and money and successfully passes her PCI assessment by concentrating her resources on the in-scope sections of her network.

THE CASE OF THE DO OVER

Angelina's Appliances, a family business selling appliances in the Atlanta area for 20 years, has built up a significant business with seven locations covering the entire metro area. Angelina's father turned the family business over to her last year and she has been spending some time modernizing the infrastructure. The business has largely run its backend systems on a shoestring budget, investing very little in the infrastructure that connects and enables the businesses to function well. Angelina knows that certain improvements can be made, including enabling many of the Internet-ready appliances in her stores to be fully functional for demos, and has spent the last few weeks with a consultant going through the infrastructure.

Her consultant, as they will, recommended a complete teardown and redo of the infrastructure that would protect both the stores and keep them interconnected with corporate. She wasn't ready to invest six figures into the infrastructure, but she knew the equipment was all relatively new and under service contracts with the manufacturer. She had the consultant build a sample rule set that he thought she would need, and log every packet that the firewalls processed. This all happened transparently to the employees and generated extremely valuable data on the back end. In the process, she learned that her payment infrastructure didn't need any Internet access directly, and the few times that those machines did access the Internet was to do DNS queries and access a Web-based map service for planning appliance deliveries. She also learned that several of her corporate employees used remote-access services to do work from home, and that some of her suppliers were connecting into her network insecurely.

Throughout the process, she learned how her networks functioned, and what she needed to do to bring herself into PCI compliance. As the consultant built her secure zones and started locking down the network, she regularly communicated with employees to talk about the changes and why they were important. The process was completed over a 6-month period, well ahead of the slew of Internet-ready appliances that she began selling for the holiday season.

SUMMARY

All systems must be protected from unauthorized access, whether it's from the Internet or any other source. Seemingly, insignificant Internet paths such as employee e-mail, browsers, or e-commerce services such as Web servers can prove to be disastrous if not secured properly. Throughout this chapter, we have discussed Requirements 1 and 2 of PCI DSS 3.0. Understanding these two requirements is fairly easy; complying with them and actually implementing the required security features can be somewhat overwhelming. Do keep in mind that real breaches have happened—and real merchants and their customers suffered—due to the neglect of those very requirement. This is NOT compliance for compliance's sake at all!

We discussed the types of firewalls that may be effective from both an internal and external standpoint, how to update your documentation, and the best ways to manage the enormous amount of data associated with them. We also discussed administrative access to systems and components and how to handle remote or nonconsole administrative access.

Configuration standards must take into consideration all network devices (i.e., firewall, router, switch, NIPS) and your computers, servers, services, and applications. Default configurations and passwords are almost always published on the Internet. For this reason alone, take precautions and change all default settings so as not to make the attacker's job easy. If an attacker makes attempts to exploit your environment and finds it difficult, chances are he'll move on to someone easier.

Next, baseline your standards. Once the different types of systems and components have been hardened, establish a baseline security configuration. This takes the guesswork out of building and configuring the next similar system. It will have the same configuration as the previous one if the baseline configuration is followed. The baseline security configuration should be updated on a periodic basis to include new changes to the system and should always follow what is stated in the configuration standards and required by your organization's security policy.

Finally, take a lesson from PCI DSS: do make sure that operational procedures for these and others tasks are defined, known and in active use. Predictable and documented always trumps ad hoc and firefighting!

Strong access controls

6

INFORMATION IN THIS CHAPTER:

- Which PCI DSS requirements are in this domain?
- What else can you do to be secure?
- Tools and best practices
- Common mistakes and pitfalls
- Case study

Access controls are fundamental to good security in almost any situation. We put locks on our cars and homes to restrict access to only authorized parties—presumably those with keys. We put passwords on computer accounts to protect them. In this chapter, we describe some basic security principles and approaches that should be understood and implemented any time access control systems are implemented—not just for Payment Card Industry Data Security Standard (PCI DSS). By understanding these basic principles, you will find it easier to make decisions on implementing each proposed access control. After learning a general understanding of access controls, you learn how PCI DSS addresses access controls and the requirements you must meet. Then, you learn about procedures that should be in place and how systems should be configured to enforce PCI compliance. Once you learn about logical access controls, as in locking down access on your systems themselves, you learn about the requirements to physically secure systems and media that contain sensitive information.

NOTE

Still, many times the easiest way to protect data is not to store it at all. It's a good idea to review the data you're keeping and verify that you really need to keep it. Remember, securely deleting the data is MUCH easier than safeguarding it from attackers!

WHICH PCI DSS REQUIREMENTS ARE IN THIS DOMAIN?

You will find references and inferences to access controls littered throughout PCI DSS. Implementing strong access controls is important enough to PCI DSS to have a top-level heading dedicated to them, as well as three rather beefy requirements (Requirements 7–9). Requirement 7 is the shortest of the three but is probably the

most important from a policy and procedure aspect. Requirement 8 delves into many of the technical controls in-scope systems must enforce, and Requirement 9 concentrates on physical security. Before we go into Requirement 7, let's walk through some of the basic principles of access controls as defined by information security professionals worldwide.

PRINCIPLES OF ACCESS CONTROL

To understand the goals of access controls, it's important to understand the three pillars of security: confidentiality, integrity, and availability, or sometimes commonly known as the CIA Triad. As you implement access control in your organization, you should always consider how you are meeting or violating these three pillars.

NOTE

In the last decade, much criticism has come upon the CIA Triad. Opponents of the triad cite that the three pillars do not fully address the basic requirements of the expanding responsibility of information assurance. One of the most notable is the “Parkerian hexad” proposed by information security legend Donn Parker (term coined by M.E. Kabay). Parker argues that the CIA Triad only covers half of the information security pillars. The other three proposed are Possession or Control, Authenticity, and Utility. For the sake of argument, only the basic CIA Triad pillars will be covered here, though other philosophies of information security may be relevant to your organization. CIA Triad might not be enough, but it is a very useful tool for understanding information security.

Confidentiality

The principle of confidentiality means to prevent disclosure of information to parties not authorized to receive it. For PCI DSS, we want to ensure that unauthorized users cannot access cardholder data. This data is defined as the primary account number (PAN), sensitive authentication data, and full-track data, but it also includes other information about a cardholder or credit card account that is stored near the PAN. This means that an expiration date stored by itself is not considered cardholder data, but the expiration date stored next to the PAN would.

Aside from PAN data, the main focus of PCI DSS, there are many other types of information we need to block from unauthorized eyes in order to protect cardholder data. Employee passwords or encryption keys are not considered cardholder data but may be used to grant access to such data and should be kept confidential.

Integrity

The principle of integrity is an assurance that data has not been altered or destroyed in an unauthorized manner. You must put measures in place to ensure that data cannot be altered while it's being stored or while it's in transit. For example, log data that is collected for PCI DSS monitoring should be stored in a manner that an administrator would know if it had been altered from its original form. Log files are not the only data with integrity requirements. Other data includes files that contain

cardholder data, system files, logs, and other critical application files that would be covered under Requirement 11.5.

Availability

The principle of availability means that the data will be accessible to those who need it when they need it. Although the first two pillars are concerned with locking down access, this one is concerned with allowing enough access that those who need the data can get to it. In general, PCI DSS is NOT concerned with the third triad leg—availability. It is presumed that the business would already be concerned about it. Still, availability in PCI context means that employees needing access to cardholder data and other critical information to perform their jobs are granted the necessary access as part of Requirement 7. However, if a security measure—such as data deletion—can make the card data unavailable to attackers, it is a definitely a valid and effective PCI DSS control! As we pointed out elsewhere in the book, deleting the data is often a much more effective control compared to, say, encrypting it or placing stringent role-based access controls on it.

NOTE

If you are using PCI DSS as the initial foundation for your security program, keep in mind that PCI DSS does not mandate availability of your systems and data. You should focus on the basics of PCI DSS and then progress to a more robust security program such as one based on the ISO 27000 set of standards.

REQUIREMENT 7: HOW MUCH ACCESS SHOULD A USER HAVE?

Let's put the principles of confidentiality, integrity, and availability (despite its somewhat tenuous connection to PCI DSS) into practice. Remember, we want to balance integrity and confidentiality (which both restrict access and ensure accuracy) with availability (which allows access and enable the business to actually function). To do this, we use the principle of least privilege. This means that we want to give an individual enough access, so they can do their work but no more.

Requirement 7 mandates that all access to cardholder data be restricted by business need-to-know. “Need-to-know” is used by governments to help define what access an individual should be given. Lee is an FBI agent and has Top Secret clearance. He gained this clearance by proving he was trustworthy through extensive background checks and several years of service. Lee is at the end of his career and has been given a boring desk job. Because his case load is fairly light, he decides to look for other Top Secret cases the FBI is investigating. Because of need-to-know, Lee is prevented from browsing through files unrelated to his cases even though he has Top Secret clearance. Unless Lee can convince his superiors that he needs access to such information, he will not be given access.

The same rules should apply in your organization. For example, Sydney’s job as a purchaser is to buy inventory to sell at her company’s store locations. Because her

job does not dictate that she should work with customer data, she does not need access to cardholder data and should, therefore, be denied access to it.

Your company must determine exactly what access each user needs. You need to make sure they can access things for their jobs, and they should be automatically locked out of everything else (Requirements 7.1.1 and 7.2.3). Companies typically do this by defining roles and assigning employees to those roles (Requirements 7.1.2, 7.1.3, and 7.2.2). The first thing you need to do is determine what access the role needs to do its job. Management must be involved in this process, and a manager should sign off on the access granted (Requirement 7.1.4). The easiest way to accomplish this is to start by assuming that no access is granted, and list the areas or resources that the role must be able to access to perform its job as exceptions. Once the roles are defined, the permissions should be input into the automated access control system (Requirements 7.1.3 and 7.2.1) built into the system where the data transits or is stored. For those users who are handling sensitive information, make sure that you have policies and procedures in place for restricting this access, and that the parties who do have access know that the data is sensitive and should protect it (Requirement 7.3).

As you are looking at what access a role needs to complete its job, make a note of any information it will need read access to but not write access. For example, Abigail may need read access to cardholder information to be able to process it for settlement, but she would never need write access to change it. In this case, we would set permissions that would protect the integrity of the information. You should also determine if certain data can be retrieved via other employees when needed. For example, Abigail's manager may need access to certain financial data only once per quarter. Because Abigail works with this data every day, she could provide a quarterly report to her manager as needed.

REQUIREMENT 8: AUTHENTICATION BASICS

Requirement 8 mandates specific authentication and identification controls for individuals who have access to cardholder data as a part of their normal job. This largely sets systems up to be able to comply with Requirement 10, which we cover in Chapter 10. Each user of the system must be held accountable for his or her actions, and it's virtually impossible to hold an individual accountable for his or her actions when he or she shares his or her username and password with dozens of co-workers. Without shoulder surfing, screen scraping, other monitoring technology, or high-definition cameras installed on each person's workspace, we must rely on our systems to help us properly authenticate users.

This starts with Requirement 8.1 (and 8.1.1), giving every user a unique ID before they may access systems in-scope for PCI DSS. Most companies provide unique accounts for things such as network resource access and e-mail, but things fall apart when administrator-level accounts get introduced into the mix. Nearly every system comes with a common administrator or root-level account for administrative purposes. Most security best practices instruct administrators to rename and disable that account before the server is placed into production.

The latest update to Requirement 8 includes a major overhaul on how the requirements are listed. Requirement 8.1 primarily deals with identification and 8.2 primarily deals with authentication. So even though you may see some requirements that look like duplicates, read carefully as they have different objectives. In PCI DSS 3.0, assessors can consider other methods than passwords for authentication, and the word passphrase makes an appearance. In addition, all of these requirements are valid for third-party vendors as well. Before your assessment begins, you should sample some of your users and make sure that they pass all of the requirements below. If, for example, you find that your authorization forms do not match the access granted, your assessor will probably find the same.

Identification, authentication and requirements

8.1.2–8.1.8 and 8.2.1–8.2.6

Requirement 8.1 describes much of the technical and procedural aspects for handling usernames and passwords for PCI DSS. This is a re-location from PCI DSS 2.0 in a general overhaul of Requirement 8, so while there are really no new requirements of note, there are clarifications and reorganizations that are important to discuss. Requirement 8.1.1. requires that all users have a unique ID before granting them access to the system. For Requirement 8.1.2, your assessor selects a sample of user IDs from your entire population, asks you to supply the form authorizing the access, and then validates that the access on the system is set up exactly as authorized. Be sure that you have as much automation in here as possible, as this leads into Requirement 8.1.3 for terminated users.

Another source of compromise is old or stale authentication information from users no longer employed with your company. If you terminate a user, you must immediately revoke his or her access per Requirement 8.1.3. If a user does not authenticate with his or her password for a period of 90 days, the account must be removed or disabled per Requirement 8.1.4. Reviewing logs, as we cover in Chapter 10 is one of the ways to track successful access from terminated employees. The account login IDs should never be seen in access logs after their termination.

Requirement 8.1.5 mandates tight controls around accounts that vendors may use to support systems. If the rumors are true, this particular requirement could have prevented a prominent breach. Although vendor accounts vary in their level of authorization on systems, they should be disabled any time they are not being used and monitored while they are in use. These support accounts can come in more than one form. Sometimes the original equipment manufacturer supports it, such as IBM or Cisco, and other times you may have a third party, such as SunGard or a division of a Big 4 to support it. Regardless, any common vendor accounts must be disabled when they are not in use. To monitor what happens during a session, you could have your vendor log into a Citrix portal to then access your machines and log the entire session. Command-based machines could make use of logging utilities built into common applications like sudo, or even using the history function of a UNIX shell and offloading the logs somewhere outside of the vendor's write access.

Locking users out: requirements 8.1.6–8.1.8

The first two requirements help to protect accounts against brute force attacks as well as the nefarious individual from abusing an abandoned, logged-in terminal. Requirement 8.1.6 mandates that systems automatically lock an account after six failed login attempts, and Requirement 8.1.7 mandates that systems maintain that locked status for at least 30 min for an automated system or until an administrator resets it for a manual system. To test this, an assessor may ask a user to perform six failed login attempts to make sure that the account locks, or they may just examine the system's settings to make sure it is set up properly. For Service Providers, note that there is an additional testing procedure that aims to ensure that noncustomer user accounts are locked out per the requirement.

Requirement 8.1.8 mandates that idle sessions time out after 15 min of inactivity. This requirement led to a myriad of interpretations, some of which actually broke a business function. For example, Matt manually runs some processes on a mainframe that takes just over 1 h to complete. When he types in the command, the session essentially freezes while the task runs but becomes interactive again when the job completes. Some Qualified Security Assessors (QSAs) interpreted this to mean that after 15 min of starting the job, the session should time out (forcing the process to terminate abnormally). This requirement should not be applied to every possible way a session could be started but instead should be smartly applied to the environment as a whole. If all mainframe sessions must be initiated from a Windows-based workstation, then make sure the workstation meets the session timeout requirements since the mainframe session runs inside the Windows one. This may not work in every case, but take the concept and find the best way to implement it in your environment.

Once you have all users working off of unique, individual IDs, you must add some kind of password (or password-like) authentication to it to meet Requirement 8.2! Many security administrators look at this requirement and think, “Well DUH, guys....” The intent of this requirement is to both define acceptable methods of authentication and prod companies to think about more than just a password for their authentication needs. The most common way companies meet Requirement 8.2 is by assigning a password to the unique account. The makeup of the password is described in the section “Password Design for PCI DSS,” later in this chapter. Alternatively, you could use some component of a multifactor authentication solution to access in-scope systems. Multifactor authentication might include a fingerprint reader embedded into your laptop or a certificate installed on your machine. Your assessor asks you to provide documentation on the authentication methods used, as well as performs the authentication for each method documented to ensure design matches reality. We’ll discuss some of those exact settings in the “Windows and PCI Compliance” section of this chapter.

The revision to Requirement 8.2 focuses on passwords and authentication, and has been updated to be more flexible in PCI DSS 3.0. Thus, instead of it only being focused on a password, the standard now specifies that any one of the generally accepted classes of authentication (something you know, something you have,

something you are/do) could be used to authenticate all users. Thus, if you had only a thumbprint to unlock your desktop, that would be sufficient in the eyes of PCI DSS 3.0.

Rendering passwords unreadable in transit and storage

Requirement 8.2.1 has far reaching impact to most organizations. Companies sometimes get confused on exactly what is required here and struggle with the interpretation, especially as it relates to Requirement 2.3 (described in Chapter 5). All passwords for in-scope systems and users must be transmitted encrypted to the system in question to prevent someone from capturing the password with simple network sniffing technology, and all passwords stored on a system must be encrypted at rest using strong cryptography. The easiest way to delineate which users and systems this requirement applies to is to ask the following question: Does this system process, store, or transmit cardholder data, meaning is it in scope for PCI? If so, then all users with administrative privileges or access to cardholder data must use some kind of encrypted channel for their authentication like Secure Shell (SSH) or Secure Sockets Layer.

NOTE

There is some gray area with this requirement. What if a user's authentication credential could also be used for access to systems not in scope for PCI? For example, let's say that Rob is a UNIX administrator and is responsible for both in- and out-of-scope systems. Some of the out-of-scope systems do not have SSH deployed on them, and Rob must use Telnet to administer them (as if he still lives in the 1980s and uses Telnet...). For ease of management, both types of systems authenticate to the same Lightweight Directory Access Protocol (LDAP) server. Would that mean that all systems must be upgraded to support SSH because credentials captured while Rob authenticates to an out-of-scope server could be used to authenticate to an in-scope server?

By reading the requirement alone and considering the scope, you might think you don't have to deploy SSH everywhere. The updated guidance for PCI DSS 3.0 does not specify, but it seems to suggest that those systems would need to be upgraded to require SSH. The best thing you can do is err on the side of common sense. Why protect a credential in one place and not another? You should deploy some kind of encryption for anything using that credential. Also, telnet and other remote access tools that expose the passwords should in fact be left in the 1980s....

Don't forget all the hard work we did in Requirement 7 with respect to roles and responsibilities! Requirement 8.1.2 states that All IDs must be implemented with the permissions as laid out in Requirement 7's role definitions. Expect your QSA to review those for consistency.

Requirement 8.2.2 is another one of those "Duh, guys" moments, but you would be surprised how easily companies are compromised because they didn't check to make sure that the person on the other end of the phone asking for a password reset was the actual owner of the account! One tactic used in penetration testing is to obtain the name of an actual employee, and then call the help desk posing as that employee and request a password reset. It works more often than you imagine. Social

Engineering is far beyond the scope of this book, but if you are interested in learning more you should check out Christopher Hadnagy's book called *Social Engineering: The Art of Human Hacking* (ISBN: 0470639539, Wiley). Requirement 8.2.2 aims to prevent that by making sure that if Steve requests a password reset that the help desk person on the other side of the phone or e-mail request, or the process on the other side of the browser verifies Steve's identity before doing it. Your help desk may need access to pertinent employee data such as an employee number, last four of a national ID (Social Security Numbers in the United States are essentially that), address, home phone, or other types of information known only to the employee. Help desk personnel should be trained on social engineering tactics (keep in mind that PCI DSS does mandate such security awareness training) and be prepared to deal with an outsider trying to beat the system.

Password design for PCI DSS: requirements 8.2.3–8.2.6, 8.4–8.6

When PCI DSS was really gaining steam, one big complaint from companies forced to comply was that the password controls were too stringent or could not be supported on the hardware that ran their businesses. Nearly every currently supported system has the capability to comply with the PCI DSS password complexity requirements. If during your compliance efforts you find systems that are unable to comply, check to make sure that it is still supported by the vendor and is not just horribly out of date. To simplify the subrequirements contained within PCI DSS Requirement 8.2, see [Table 6.1](#) that explains everything that your in-scope systems must enforce for password controls.

For these requirements, systems must enforce these controls. Having only a policy that describes the proper procedure for making passwords is not acceptable. All the above requirements can be met by modern UNIX and Windows operating systems. We'll show you how to accomplish this in the "Windows and PCI Compliance" and "POSIX (UNIX/Linux Systems) Access Control" sections of this chapter.

Table 6.1 PCI DSS Password Complexity Requirements

Req. No.	Control
8.2.3	Passwords must contain letters and numbers and be at least seven characters in length. Of the at least seven elements or positions in each password, at least one of them must be a number and at least one must be a letter
8.2.4	Expire passwords every 90 days. All users must be forced to create new passwords for their accounts at least quarterly
8.2.5	Password must be different from last four. When users change their passwords, they must not be able to use a password that has been used in the last four changes
8.6	No group or shared passwords. Each user should have his or her own user account and unique password not to be shared with others. Furthermore, requests for group or shared IDs should be denied

First-time passwords are often an easy way to compromise an account. For example, when Steve joined his company, he was provided with a cell phone and a laptop. His user ID was his first initial and last name, and his password was “Newuser1.” The initial password was the same for every user and would technically exceed the complexity requirements of PCI DSS. The password is alphanumeric and includes a mixture of uppercase and lowercase letters. But because every user gets the same password, compromising a new account might be a trivial operation with a little bit of social engineering. Requirement 8.2.6 mandates that all new accounts have a unique password that expires immediately after its first use. We’ll cover configuration methods to do these for both Windows and UNIX in the “Windows and PCI Compliance” and “POSIX (UNIX/Linux Systems) Access Control” sections of this chapter.

Requirement 8.4 mandates that you communicate all the password procedures in PCI DSS to the in-scope user base. An in-scope user is a user who has access to cardholder data as a normal part of his or her job. These users must be made aware of the password procedures, and your assessor will randomly sample users and ask them what they know about password procedures. Assessors may do this as a part of an interview for another area of PCI DSS, or they may specifically ask for a list of users and randomly call them for a phone interview.

Requirements 8.5 and 8.6 are relocated from PCI DSS 2.0, but the key message is the same. Do not, under any circumstances, use shared passwords or IDs. Each user or application must be uniquely identified with its own credential to comply with PCI DSS.

Two-factor authentication and requirement 8.3

Requirement 8.3 mandates at least two-factor authentication for remote users accessing in-scope systems. PCI DSS 3.0 simply clarifies who this applies to as follows: users, administrators, and all third parties, including vendor access for support or maintenance. Again, with recent breaches, this requirement would have saved one company some time and money. As of this writing, it is rare to see a company without remote access technologies deployed into its environment (although, apparently, it does exist). Administrators need remote access to machines in break-fix situations, and general corporate users may need remote access for tasks like responding to e-mail or uploading documents to collaboration products. Not all users need a two-factor solution to comply with this requirement. Remember your scope! Most companies only have a small subset of employees that need this type of authentication. Network and user segmentation is an excellent way to reduce the scope of this requirement, dramatically decreasing the cost and effort required to deploy a solution. Any user (be it an employee, contractor, or other third-party company’s employee or contractor) who is remotely accessing the cardholder data environment must perform a successful two-factor authentication before being granted access to the cardholder environment. If you want to shrink your scope, you could even consider your corporate network semitrusted, and require local users to use two-factor authentication to access the environment, thus effectively shrinking your PCI DSS scope.

Passwords by themselves are a losing battle. Check out this article that debunks many of the commonly held beliefs about passwords: <http://bit.ly/1avyTmu>

NOTE

Multifactor authentication does not need to consume your entire Information Technology (IT) budget! A cost-effective solution would be to set up a certificate authority inside your company, and issue user-based (not machine-based!) certificates that require a password to be used to “unlock” the certificate. As long as both the certificate and password are uniquely assigned (and not group-based), this is a perfectly acceptable solution to meet Requirement 8.3. When building this environment, ensure you are using good security practices around these certificates. Check NIST for some guidelines around PKI.

Databases and requirement 8.7

Databases contain lots of information valuable to a hacker, yet the security around databases is sometimes the worst in the entire enterprise. Many compromises occur because of administrator-level accounts with blank passwords. Requirement 8.7 has four testing procedures. Procedure 8.7.a requires assessors to verify that all users are authenticated prior to being granted access to the database, 8.7.b and 8.7.c require direct user interactions with the database to be done through programmatic methods such as stored procedures, and that direct queries to the databases are restricted. If you have power users that log into your database directly instead of going through an application, take any common actions they may perform and put them into stored procedures or functions, and then restrict their access to those elements. Better yet, code these actions into the application and force users to use that method instead.

Procedure 8.7.d requires that assessors verify that application IDs and their passwords can only be used by the authorized applications and not by individual users or other processes (typically meaning that the accounts do not allow for interactive login and you avoid using passwords over keys and certificates). This can be challenging depending on your infrastructure. Older versions of database servers may not be able to sufficiently distinguish users from applications. Consider the following example.

Diana is a Database Administrator (DBA) and manages two main locations where enterprise data is stored. Her business critical information is stored in various locations on a mainframe. The security added to the mainframe allows batch processes to operate under noninteractive login credentials, thus preventing those credentials from being used for an interactive session with the data. Diana’s Web farm for her e-commerce site pulls its data from a PostgreSQL database. In her pga_hba.conf, she set an Internet Protocol (IP)-based restriction on the application’s ID by adding in the source IPs that are valid from her application servers. She has four different ones in her enterprise, so all four of the IPs are in her pga_hba.conf, and the application IDs can only be used from those machines which are considerably locked down.

TOOLS

Here is a sample pg_hba.conf with IP-based limitations. Assume that the database is called “CommWebsite” and the ID used for access is “CommUser.” Your pg_hba.conf would look like this:

```
# TYPE DATABASE USER CIDR-ADDRESS METHOD host CommSite CommUsr  
10.4.30.0/29 password
```

TOOLS

Want to see how strong your passwords are? Mandylion Research Labs (www.mandylionlabs.com) created a fantastic brute force calculator that you can download (www.mandylionlabs.com/documents/BFTCalc.xls) and test to see how long it would theoretically take to break a password or key. Plugging in the elements of the password above (Newuser1), it would take the average computer just over 2½ h to break that password. Let’s say that you didn’t know that the password contained one uppercase letter, six lowercase letters, and one number and assumed an eight-character random mix of uppercase and lowercase letters and numbers. If you made this assumption, the average computer would take a little more than 6300 h using a brute force attack to crack the password (an effective key strength of 236). Adding special characters in it would take over 117,000 h with an effective key strength of 252. This is where user education is important.

TIP

In Windows (especially prior to 2008) it is a best practice to rename the RID = 500 account and set an extremely long passphrase. Activity using this account should not ever occur as administrators should use their own unique accounts. Events from the RID/SID = 500 account can be monitored with a log management system.

This is relatively easy when you have infrastructure such as LDAP or Active Directory deployed, but it can be a challenge when you have machines that stand alone, not as part of a formal policy enforcement process.

NOTE

Why do operating system manufacturers insist on continuing the trend of providing a root or administrator user account that has access to the whole system for whatever it wants to do? Poorly developed software with global administrator privileges will surely lead to a root-level compromise, whereby a system is then “pwned” by an attacker. Software that needs elevated privileges must be limited to sandboxes on the servers and should never require administrators to run it under the root-level administrator account. If your vendor tells you it is a requirement, tell them you will be taking your business elsewhere.

Educating users

Although PCI’s password requirements are not incredibly strict, they may be stricter than what your company was using before becoming compliant. If your company is

going from a very relaxed password policy to a stricter one, you will probably meet resistance from employees. Of all the changes you may have to make, this is one that affects an employee's day-to-day work. Some employees have a hard time seeing the benefits from using strict password policies—some may even grumble that this is just another way the IT department is making their lives more difficult.

Instead of forcing policy upon users and communicating it through e-mail or newsletters, meet with employees to personally explain the policy to them and answer any questions they may have. This is a great opportunity to educate them on what makes a good password and why they are important, and this could even be considered part of your security awareness training for Requirement 12.6.1 and would apply to Requirement 8.8 as well. Management should be involved in this meeting, and it makes sense to tie it to some other form of All Hands event. As we learn from our mothers (thanks Mom!), it's best to lead by example. If employees observe management interested in and adhering to the policy, they will take it more seriously. You may want to get someone from management to briefly introduce that the company will be implementing a new password policy to become more secure.

One of the things you want to cover in this meeting is the password complexity requirements that will be enforced. Many times, users get frustrated when it's time for them to change their password because they don't understand why any of their new passwords are not being accepted. Give them examples of passwords that both conform and violate the policy, with information on why they do not comply.

You may also want to go over some tricks to help them choose good, secure passwords that will be easy to remember. For example, some security experts advocate writing out a sentence and using the first or second letter from each word. One of our colleagues suggests having users pick two items always present on their desk (such as a coffee cup and a monitor) and the password might be "c0ff33Cup&m0n!tor." Another trick is to take certain letters and interchange them for numbers that look like those letters (e.g., 3 for E, 7 for T, 1 for L) or take letters and interchange them for symbols the same way (e.g., # for H, \$ for S). For example, the sentence "Bill grabbed a brewdog at the high school reunion gathering" could become the password "Bg4bathsrg," and the verse from one of the author's favorite songs "Show me, how you want it to be, tell me baby, cause I need-to-know" could become "Sm#yw1btmbc1ntk." The effective key strengths of those passwords are 2^{45} and 2^{72} , respectively. Many security professionals now recommend using passphrases given the nature of password cracking today. Be sure if you do that you still enforce uppercase and lowercase letters (for complexity) and numbers (to comply with PCI requirements). A great reference is Mark Burnett's book, *Perfect Passwords: Selection, Protection, Authentication* (ISBN 978-1-59749-041-2, Syngress). Much of the book is dedicated to helping users select passwords that are unique and easy to remember.

Users should be educated to never give out their passwords under any circumstances to anyone, including the IT staff. Researchers have studied human behavior to see how quickly people would give away their passwords to strangers, and many

of them chose to exchange it for a piece of chocolate. It would be interesting to perform a correlation or longitudinal study to see if this differs for kids who “passed” the delayed gratification studies done by Stanford (look up the Stanford Marshmallow Experiment).

Use this opportunity to help them understand how often password changes will be required (at least every 90 days), and that they will not be allowed to reuse old passwords (at least the last four). You should also review company policies about disclosing passwords. Passwords should never be disclosed to anybody for any reason. Employees should understand the process that’s in place to reset their passwords if they forget it. You should always ask users if they have any questions when rolling out a new policy.

WINDOWS AND PCI COMPLIANCE

If you work in an organization where Windows is widely deployed, you’re probably using Active Directory to authenticate users. One of the great things about Active Directory is that it is easy to roll out many of the requirements for PCI to the enterprise. Using Group Policy Objects (GPOs), you can enable password-protected screen savers and set up password policies all from your domain controller. You may also have standalone Windows computers that aren’t part of the domain (e.g., a Web server that’s at a hosting company), so we’ll show you how to configure these for PCI compliance as well.

Windows file access control

Windows Access Control Lists (ACLs), or Discretionary Access Control Lists (DACLs), are used to configure and enforce access control. ACLs contain a list of Access Control Entities, and each entity defines permissions. To set ACLs in Windows, you must have proper administrative privileges. Because Windows uses discretionary access control, the owner of the file and administrators can configure ACLs for an object. When using Windows access control mechanisms, you basically have three options: you can explicitly allow permission, explicitly deny permission, or implicitly deny permission.

When you implicitly deny permission, this means that you did not explicitly allow or deny access. By default, Windows denies all access to objects that do not have rights set on them. This is a great best practice to follow for all systems and is particularly good because it helps us comply with PCI Requirement 7.2.3 without doing anything. Because Windows implicitly denies access, explicitly denying access should only be used in special cases where you are denying permission to a subset of a group. One user you would normally never deny access to is the built-in “Everyone” group because this will deny access to all users including the administrator. The correct way to do this would be to add users and groups that should have access to the file and then simply remove the Everyone group from the allowed users. Because Windows follows an implicit deny for anyone not explicitly given permission, this will likely give you the desired result.

WARNING

System administrators are busy. Sometimes they will give all users administrative rights instead of properly reducing each user's (or role's) rights to the minimum necessary to do his or her job. This is bad for many reasons, including higher support costs when "Acts of CLOD" occur. With everyone acting as an administrator, Windows no longer follows the default deny policy required by PCI Requirement 7.2.3 because all users are allowed full access to all files.

When configuring access controls in Windows, there are several tricks that can save you time in initial configuration and later maintenance. Remember the roles you created as part of the Requirement 7, "How Much Access Should a User Have?" section earlier in this chapter? Here's where we use them! Once you have the role, you must create a group with those permissions and assign all the required users to that group. With users belonging to roles or groups, you can set access permissions for the whole group instead of each user individually. This also makes maintenance much easier because you can change permissions for the entire group and remove and add users whenever needed. It's not uncommon to have users who are assigned to more than one group. For example, one user may only need access to unprocessed cardholder information, whereas another user may need access to unprocessed and processed cardholder information. In this case, both users would be members of a group with access to cardholder information, but only the second user would also be a member of a group with access to processed cardholder data.

NOTE

The process of defining roles is not a weekend or after-hours gig. One author assisted a customer in creating a detailed set of roles for a top 10 financial institution in the United States. What started as an initial set of 900 defined roles escalated to over 3000. Although the exercise ultimately yielded a much more secure company with an easily managed set of permissions, the effort was much larger than anticipated.

Another great time saver, but a potential minefield, is to use inheritance as much as possible. When you set permissions on a file or folder, you can also specify how subfolders will inherit those permissions. This makes it much easier to configure access control on a few folders that are near the root folder, instead of needing to configure each subfolder individually. Just remember that if you set up inheritance, by default, subfolders have the same permissions. Security templates can assist with this if you find that you have common types of folders to which you grant access often. This keeps all security settings in the same location and makes them much easier to manage.

WARNING

To be able to effectively secure data in Windows, you should always use the New Technology File System (NTFS). FAT32 does not cut it because it does not have the capability to do access control.

NOTE

Remember that new password requirements will not be enforced until the next password change, so to be PCI compliant today, you would have to have all users change their passwords today.

Finding inactive accounts in active directory

One of the PCI requirements is to find all accounts that have been inactive for 90 days or more and remove or disable them. In Active Directory, there are several ways to find inactive accounts, although in many cases you have probably struggled to find one that works well. For some tips outside the scope of this book, check this TechNet article: <http://j.mp/yodXCI>.

Enforcing password requirements in windows on standalone computers

If you have several standalones Windows systems you can make a local security policy template and incorporate it's use for all in-scope systems.

In a virtualized environment you can also make your virtual machine template(s) have these settings baked in.

To set password policies for a Windows computer (including 2000, XP, 2003, Vista, and later editions) that is not connected to the domain, you should use the Local Security Settings dialog box, which is set up basically the same way as a GPO, except that it will only affect the local computer.

- *Windows Server 2003/2008:* Click on **Start | All Programs | Administrative Tools**. Inside the Administrative Tools dialog box, click on **Local Security Policy**.
- *Windows Vista:* Click on **Start | Control Panel**. Inside the Control Panel dialog box, click on **System Maintenance | Administrative Tools**. In the Administrative Tools dialog box, click on **Local Security Policy**.
- *Windows 7:* Click on **Start | Control Panel**. Inside the Control Panel dialog box, click on **System Maintenance | Administrative Tools**. In the Administrative Tools dialog box, click on **Local Security Policy**. OR Click on **Start | All Programs | Administrative Tools**. Inside the Administrative Tools dialog box, click on **Local Security Policy**.
- *Windows 8:* Under **Computer Configuration and User Configuration**, click on and expand Windows Settings to see the Security Settings. Now open the **Local Security Policy** editor.
- For Windows 8.x the majority of the documentation available suggests that clicking start, run, and typing secpol.msc (being sure to include the .msc) is the way to access the Local Security Policy.
- *Windows Server 2012:* Click on **Start | All Programs | Administrative Tools**. Inside the Administrative Tools dialog box, click on **Local Security Policy**.

Now expand Account Policies, then click on **Password Policy** (for an explanation of what these settings mean, refer to the earlier section “Enforcing a PCI Compliant Password Policy in Windows Active Directory”). Enforce password history should

be changed to at least four times to meet PCI requirements. The Maximum password age should be set to at most 90 to meet PCI requirements. The password length should be at least seven characters for PCI requirements, and passwords must meet complexity requirements and should be set to enabled. It's also a good idea to set the Minimum password age to at least 1. Otherwise, when a user is required to change their password, they could change it four times then back to their original password. When this setting is set to 1 or more, the user must keep the same password for at least that many days before they can change it again.

You should also configure the Account Lockout Policy to comply with PCI requirements. To do this, expand Account Lockout Policy. Double-click on **Account lockout threshold**. In the Account lockout threshold, Properties dialog box change number of invalid login attempts to 6. A dialog box will pop up and ask if it should also change the Account lockout duration and Reset account lockout counter after attributes as well. These should both be changed to 30 min to comply with PCI requirements, which is what the default is in this new dialog. Click **OK**.

WARNING

All these settings may be irrelevant if the users who connect to them have local administrator privileges! Do yourself a favor and remove all local administrator access from your users' accounts, or look to your specific installation to ensure this cannot happen.

Enabling password-protected screen savers on standalone windows computers

Setting screen saver options is much easier to maintain and enforce using Active Directory. If you have computers that are not connected to a domain, these options can be set on each computer individually.

- *Windows 2003 Server:* Click on **Start | Control Panel**. In the Control Panel, double-click on **Display**. Inside the display dialog, click on the **Screen Saver** tab. The Wait option should be set to 15 min at the most. Also verify that **On Resume, password protect** is checked.
- *Windows Vista:* Click on **Start | Control Panel**. In the Control Panel, click on **Personalization** and then on **Screen Saver**. In the Screen Saver dialog box, set the Wait time to a maximum of 15 min. Also verify that **On Resume, display logon screen** is checked.
- *Windows 7/2008:* Click on **Start | Control Panel**. In the Control Panel, click on Appearance and Personalization. Click on Change screen saver under Personalization. In the Screen Saver dialog box, set the Wait time to a maximum of 15 min. Also verify that On Resume, display logon screen is checked.
- *Windows 8 and Windows 2012:* Click on **Start | Control Panel**. In the Control Panel, click on Appearance and Personalization. Click on Change screen saver under Personalization. In the Screen Saver dialog box, set the Wait time to

a maximum of 15 min. Also verify that On Resume, display logon screen is checked.

Setting file permissions on standalone windows computers

In Windows Explorer, navigate to the file or folder you would like to modify permissions on. Right-click on the **file or folder** and then click on **Properties**. In the Properties dialog, click on the **Security** tab. To add a user to the list of Group or user names, click on the **Add** button and the Select Users, Computers, or Groups dialog box will appear. You can then type in the name of a **user or group**. The **Advanced** button gives you more options to help you find the correct **group or user** to add. After you click **OK**, the user or group will appear in the previous dialog box

POSIX (UNIX/LINUX SYSTEMS) ACCESS CONTROL

UNIX-based systems such as Linux use POSIX-style ACLs. This means files have three permission modes: read (r), write (w), and execute (x). These modes can be assigned either using the letters just listed or they also have equivalent numbers. Read is 4, write is 2, and execute is 1. If file permissions are being set using letters, it will be a string of letters or dashes (e.g., a file with read-only permission would show r- and a file with read, write, and execute would show rwx). When using numbers, they are added to denote permissions. Read permission would simply be a 4, and read and write permission would be 6 (4 plus 2). When using POSIX-style access controls, there are three groups or users you set permissions for. The first set is for that specific user who owns the file. The second set is for the group who owns the file. The third is for all other users who do not have any ownership over the file, similar to the Everybody group in Windows. So, a file that allows the owner to read and write, and everyone else only read access would look like this -rw-r-r- or in numeric format it would be 644.

Linux has great command-line tools for changing file permissions and file ownership. Although exploring all that these commands can do is beyond the scope of this book, we will discuss some basics here. In Linux, to list file permissions, the ls command can be used. The syntax to list the file permission and the group and user who own the file is as follows:

```
ls -lg [filename]
```

To change file permissions in Linux, you usually use the chmod command. You can run the chmod command using numbers. The following example uses POSIX permission number format to set a file to allow the user who owns it to read, write, and execute the file, and everyone else to read and execute but not write to it, similar to a standard executable file:

```
chmod 755 filename
```

Or you could use letters and specify if you are going to add them or delete them from users (u), groups (g), others (o), or all (a). For example, to allow the user who owns the file to read from it and write to it, you would do the following:

```
chmod u=rw filename
```

To take away permissions use a hyphen or minus in front of the permissions parameter. To deny read, write, and execute permission to the group that owns the file and to all users other than the one that owns the file, you would do the following:

```
chmod go-rwx filename
```

To change the file ownership, use the chown command. To change the user and group that owns a file, do the following:

```
chown newuser:newgroup filename
```

In POSIX-style systems, there are three additional attributes that affect how files are executed and accessed. These are set user ID (SUID), the set group ID (SGID), and sticky. These settings work differently when they're applied to files or directories. The SUID bit can be configured to tell the file what user it should run under when the file is executed. Many times this is used to allow a nonroot user to run a file as the root user. This is used if a user needs to run a file that requires root access, and you don't want to give their account root access or the root password. SGID for a file works the same way as SUID, but it specifies what group the file should execute as. The sticky has no effect on individual files. The SUID bit has no effect on directories. If the SGID bit is set on a directory, any new files created in that directory will be owned by the group specified using the SGID instead of the group of the user who created the file. This is sometimes used in directories where many users will share files. When the sticky bit is set on a directory, only the user owner of the file or root can delete or rename a file (the group owner cannot). This is sometimes used in shared directories where you don't want users other than the owner or root to delete or rename a file.

In Linux, there are also several mandatory access control systems. Most of them are somewhat limited to protecting only a subset of files on the system (normally only critical system files). SE Linux is an example of this. SE Linux was developed by the National Security Agency and has been incorporated since the 2.6 series Linux kernel. SE Linux uses targets to specify what files it will control and how it will control them. Other mandatory access control systems that are currently being used in Linux include Suse's AppArmor, Rule Set Based Access Control.

Linux enforce password complexity requirements

Most Linux distributions support password complexity enforcement using Pluggable Authentication Modules (PAMs). This is normally set in /etc/pam.d/system-auth.

To comply with PCI requirements, a password must be seven characters long and contain uppercase, lowercase, and numeric characters. `pam_cracklib` has parameters to help you meet these requirements. The `minlen` parameter is used to specify the minimum length of a password. The `dcredit` parameter is used to require digits, the `ucredit` is used to require uppercase letters, and the `lcredit` parameter is used to require lowercase letters. The `retry` parameter is used to specify how many attempts a user gets before the password program exits. Let's put all these together to show the entry in `/etc/pam.d/system-auth`:

```
password required /lib/security/pam_cracklib.so minlen=7 dcredit=1  
ucredit=1 lcredit=1 retry=5
```

Depending on your implementation, you may see different names for the PAM configuration files where this information is placed (e.g., in Debian, you would find this information in the `/etc/pam.d/common-password` configuration file).

CISCO AND PCI REQUIREMENTS

Cisco devices have some important settings that should be used for you to become PCI compliant. All passwords should be encrypted when stored or in transit. Most operating systems do this and do not really give you an easy way to store them unencrypted even if you want to. Cisco devices are an exception; however, it's important to check this.

CISCO ENFORCE SESSION TIMEOUT

To force Cisco devices to automatically timeout if a session is left inactive, use the `exec-timeout` configuration under the appropriate line configuration. The syntax for this command is `exec-timeout [minutes] [seconds]`. For PCI compliance, this should be set to as follows:

```
exec-timeout 15 0
```

Encrypt cisco passwords

The current best practice from Cisco is to always use “enable secret” and “username secret,” instead of enable password. Enable password encrypts the password using a very weak encryption algorithm that has been broken for a long time. The `secret` command uses Message Digest 5 (MD5) to hash the password. Although MD5 has shown some weaknesses lately, this is far better than the alternative and the best Cisco is giving us right now. A better option would be to use directory-based authentication models such as RADIUS or TACACS+ to prevent these usernames and passwords from being stored directly on the device. Even encrypted passwords can be vulnerable to attack when disclosed.

Setting up SSH in a cisco environment

By default, Cisco routers allow Telnet access to the line vty 0 4 port for remote configuration. To disable this and set up an SSH server, you must first have an IOS version that supports IOS with the appropriate feature pack (typically the crypto pack). You need to set up either local authentication or as suggested above, tie the device to a directory. When managing any more than a few devices, pointing the authentication to a directory service makes administration much easier.

If you have already directed your device to a directory service, skip to the next configuration step. Otherwise, you need to enter this into your router after entering the “Terminal Configuration” mode by typing config t:

```
aaa new-model
```

The next command generates the keys required to perform SSH encryption:

```
cry key generate rsa
```

Then finally, to disable Telnet for remote access, type the following two commands:

```
line vty 0 4  
transport input ssh
```

Then, save your configuration!

REQUIREMENT 9: PHYSICAL SECURITY

There are three basic types of physical security. The first type is obstacles such as doors, walls, and other barriers, which can help stop or at least delay intruders. The second type is detection mechanisms such as alarms, lighting, guards, and television cameras that help detect attacks. The third type is response, which includes things you would put in place to stop an attack in progress or soon after. It’s important to use all these types of physical security to protect sensitive information. For example, you may put sensitive data behind a locked door and have security cameras monitoring that door, recording everybody who goes in and out. You may also have a guard on duty who can quickly respond to stop anyone who’s trying to circumvent the lock. Security measures in plain sight act as a deterrent to attackers, sometimes preventing the attack in the first place.

Requirement 9.1 mandates “facility entry controls” for in-scope areas including computer rooms, data centers, and other physical areas where in-scope systems may live. Acceptable controls include lock and key, badge access, or some other barrier

that automatically locks and only unlocks for the people authorized to access these rooms (Hint: the President of your company should not have access). Requirement 9.1.1 mandates the use of video cameras or other access control mechanisms to monitor individual physical access to “sensitive areas.” No doubt those areas include the ones mentioned in 9.1, but arguably they would include a large physical storage area of paper records that contain cardholder data. These large storage areas still exist in many places in the United States, and the payment systems in some countries require exchanging a significant amount of paper. Those areas should be protected in the same way and should have cameras monitoring access. In addition to simply placing the camera there, you must protect the video data from modification and regularly review and correlate the data, as well as store it for a minimum of 3 months.

Before PCI DSS version 1.2 (note that we are at 3.0, but we just wanted to provide you with some historical context), this requirement has been interpreted to mean a wide variety of controls. PCI DSS 3.0 kept the clarifications from 2.0 and does a good job clarifying exactly where these cameras should be placed. For example, placing cameras over each cash register is not required, but if you store cardholder data in a server room at a store, that would need to be monitored.

Requirements 9.1.2 and 9.1.3 aim to protect inherently vulnerable areas of your environment. Requirement 9.1.2 targets publicly accessible network jacks and mandates the access to such jacks be restricted. This one can be challenging as far as its intent. If you have conference rooms or common areas with network jacks that are outside the restricted areas of your company, you should disable them or segment them away from networks where cardholder data may be processed. This requirement would not apply to a conference room *behind* a secured area where visitors must be escorted. QSAs in the past have incorrectly read this requirement to mean that *all* conference room jacks must be disabled. This is incorrect, only those areas considered publicly accessible (i.e., with no physical access such as a badge reader protecting them). Another area to look out for with respect to 9.1.2 is retail store locations where network jacks may be placed throughout the store in plain view (or otherwise unrestricted) that might also sit on the point-of-sale (POS) network (or a network where cardholder data may be processed).

It is time for a real-life example. One of the authors was working with a customer who had a chain of cafes. When sitting down at one of the tables in the cafe, a network jack was discovered slightly obscured by a plant. This jack was actually hooked up to the same network as the POS systems, and an attacker could easily hide a device that could take advantage of this major design flaw by siphoning off every transaction processed through the store.

Requirement 9.1.3, discussed in Chapter 8.

Handling visitors: requirements 9.2–9.4

First, we begin with documentation! Requirement 9.2 talks about procedures to distinguish employees and visitors, and the testing procedure 9.2.a determines if the

procedure covers granting badges, changing access, and revoking terminated or expired badges. The other testing procedure 9.2.b sees if you actually follow your own policies! Requirement 9.2.c is in place to clarify that access to the systems that manage access are limited to authorized personnel. In 3.0, you will notice 9.2.d which adds an additional examination by your assessor to make sure that visitors and on-site personnel badges are easily distinguished, but is now clarified to say that badges are not required. You just need some way to clearly distinguish these groups such that someone who does not know the individuals could understand what bucket they fit in.

NOTE

During your assessment, make sure you make your assessor follow the procedures you set! There is nothing that says FAIL more than when a company being assessed forgets to give the assessor a visitor badge.

Requirement 9.3 is a practical test for what you set up in Requirement 7. Essentially, your QSA should take a sample of individuals with physical access to sensitive areas and make sure they have authorization paperwork and that their job function requires this access. Watch out for interpretation issues here as an overzealous QSA may decide that some of your employees don't actually need access. You should work with your QSA to ensure they are educated on the business reasons for the access. The QSA will also look through some recently terminated employees who have had access and make sure that their access is revoked.

Requirement 9.4 deals with visitors exclusively. Requirement 9.4.1 is a test to make sure that the badge you give to the assessor does not open doors where sensitive information is stored. This is what the Council means by "unescorted access." You can expect your assessors to try and get their badge to open a data center door or other sensitive area. Obviously, if you issue plain paper badges to your assessors, this will be a fairly easy requirement to pass. Your assessor will also look at the badge you give them and compare it to your badge. Requirement 9.4.2 mandates that employee badges and visitor badges visually appear different and have distinguishing marks such that an employee of your company could easily identify someone as a visitor by the badge used to identify him. Finally, Requirement 9.4.3 mandates that visitor badges are surrendered upon leaving the facility. Your assessor will probably perform the required testing procedures without you even knowing, so be sure your company is following the policies and procedures you set out!

Requirement 9.4.4 is documentation-based but not in the way you might think. When visitors are allowed to visit the facility in general, data center or other sensitive areas, they must sign in. The three items that must be captured for every access are the person's name, the firm represented, and the name of the employee authorizing the access. You must also retain this log for at least 3 months (unless restricted by law), so expect to add dates and times to the above three items.

Media and physical data entry points: requirements 9.5–9.8

Up to this point, the main focus with cardholder data has been online or live data. Data is not always online or live and exists in many different places. Requirement 9.5 mandates physical protection of all kinds of media that contain cardholder data. The term “media” is intentionally broad here, and the examples they list include computers, removable electronic media (such as USB drives), communications hardware, telecommunication lines (arguably not required if all data over the wire is encrypted), and paper (receipts, sales reports, chargeback or dispute reports, faxes, mailrooms). Although this is a procedural requirement (your assessor must review your procedures to ensure that this is addressed), your assessor may validate that you follow your procedures and ask to see areas where this type of data may be stored.

Requirement 9.5.1 deals with backup media. Backups are not required to be stored off-site, however! Several companies make use of on-site tape vaults in their primary data centers to ensure that the data remains secure. If the media goes off-site, don’t send it home with one of your employees to put in her house. Be sure it is a facility that is secure and that the contracts comply with Requirement 12.8.

When media is distributed outside your company’s secure facility, you must protect the media in three distinct ways. First, Requirement 9.6 mandates a policy be put in place to strictly control the distribution of cardholder data. The fewer places you send in-scope data, the less likely you will have a breach because someone did not adequately protect the data. Requirement 9.6.1 states that all media must be classified in a manner such that it can be identified as confidential. This requirement could literally be interpreted to say that the media must have the term “CLASSIFIED” written on it, or more accurately interpreted to state that if you label tapes with a colored dot, the *red* ones are considered classified by your policies. Finally, Requirement 9.6.2 mandates that any media transported outside the facility is done so via a secure courier or in a manner by which it can be tracked. Something like your Iron Mountain driver or a FedEx package would suffice.

When media is transported off-site, you need to enter it into a log so that you know where your media inventory is at any given point. Your assessor will review several days of logs, per Requirement 9.6.3, and make sure that both the tracking information are included as well as proper management authorization. Someone capable of providing proper management authorization could be your data center manager or another person with the delegated authority (and accountability) for authorizing the transport of media.

Requirement 9.7 mandates strict control over the media such that it is only accessible to the employees who need it and periodic inventory of the media (per a policy). The testing procedure for Requirement 9.7.1 requires your assessor to review the media inventory log to make sure that periodic inventories are performed.

NOTE

What is periodic? Good question! PCI DSS has waffled back and forth on key requirements where ambiguous terms like periodic and should were present. In PCI DSS 3.0, the instances of periodic increased from 8 to 20, and the instances of should went from 27 to 103. The crowd that wanted

more flexibility won this round, but you can imagine how painful this is going to be during a PCI assessment. So what should you do? Use this as a general rule of thumb. You must have an assessment performed annually, so do any periodic requirement at least annually so you have something recent to show your assessor. Be sure to back up your choice of frequency with some authoritative source that states it as an appropriate length of time. There are plenty of sources out there that can help you, including NIST and DoD.

Requirement 9.8 deals with the destruction of cardholder data. In cases where you need to dispose of any media containing cardholder data, it needs to be destroyed in a manner by which the data is not recoverable. This means that if you are done with a hard drive, you must either electronically destroy the data or physically destroy the media—a simple delete does not work. Requirement 9.8.1 describes some methods that could be used for hardcopy media and mandates that shred bins are available for employees to use (where applicable), and those bins are protected by some kind of locking device. Expect your assessor to walk through your areas and jiggle that lock a bit. Requirement 9.8.2 mandates that electronic media is destroyed appropriately. This type of media could be electronically destroyed with something like a bulk eraser (only for magnetic-based media) or physically destroyed with a giant shredder or incinerator. One thing you do not want to do is have a party with some firearms, a DeWalt, or a steamroller to destroy your old electronic media. Do it properly with logs. Your assessor may want to review a sample of any electronically destroyed media to ensure that the data is not recoverable.

If you are now deploying Solid State Disks or use USB Flash media, you must understand that traditional secure wipe does not work effectively to destroy data on these media. In order to protect these, be sure that you are encrypting the drive or working from an encrypted volume before deleting. This can accomplish the same effectiveness of making the data unrecoverable without access to the keys.

Anti-skimming: requirements 9.9

Requirement 9.9 is a new one for PCI DSS 3.0, and is designed to combat skimming. Skimming is a massive issue as it relates to credit card payments, and skimmers can be found in any type of device. Some skimmers come preloaded into devices direct from dirty manufacturers or integrator resellers. Unattended payment terminals (think pay at the pump fuel or ATMs) are particularly susceptible because they are not watched 24x7. The bad guys are getting very creative and have some pretty incredible resources. Injection molds, lighting, and embedded electronics are all present in skimmers now such that you wouldn't necessarily be able to detect with the naked eye.

Requirement 9.9 states that you must protect devices that capture payment card data (payment terminals) from tampering and substitution. This includes periodically inspecting them and looking for tampering and substitution. You must have logs to show your assessor all of this. In small shops, this will probably be pretty easy to do, but it will become something that you will have to work into a monthly or weekly routine. For big shops, this is going to be an issue.

TOOLS

There are automated products out there, such as SpotSkim (www.termtegrity.com), that will allow you to meet requirement 9.9, 9.9.2 with automation. It leverages a smartphone app to inspect the terminals and then a SaaS portal to maintain the inventory with reporting for your assessors. You can track this via paper or the whole Excel/Sharepoint route as well. Given the challenges associated with tracking all of your devices and making sure you have accurate logs, consider going for some kind of automation here. This is a manually intensive process that you could easily supplement with tools to save time.

Requirement 9.9.1 takes the concept of the big list of devices you must maintain and applies it to these very terminals. You must keep the inventory up to date and ensure that the list is accurate for added, relocated, or decommissioned devices. Your assessor should take a sample of the devices in your list and verify that they have been inspected, that they are correctly listed as operational (or decommissioned), and are in the correct location.

Requirement 9.9.3 is interesting in that it attempts to tackle social engineering as a risk for terminal tampering. Your front-line employees are responsible for verifying the identity of any technician dispatched and the work orders to work on payment terminals. You must train your front-line employees on how to spot suspicious behavior and give them a way to report it. If they do report it, be sure you track it and follow-up with logs. Your assessor will want to review the training as well as interview front-line folks, the cashiers and POS operators, to make sure they have this training and have a method to report this behavior.

Don't forget Requirement 9.10, which is your catch all on making sure all of your documentation is in place.

WHAT ELSE CAN YOU DO TO BE SECURE?

This chapter covers how to create PCI Compliant access controls for in-scope data. One of the most effective things you can do is reduce the amount of in-scope data stored in your systems to both make it easier to comply with these three major requirements and to improve your general security. Let's explore some areas where you might store in-scope data that could be reduced or eliminated.

Retail stores are notorious for storing data well beyond their useful lifespan for various purposes. When retailers started to embrace the concept of a computer to run their POS and process credit card transactions, it appears that the equipment first deployed was unreliable. Why else would you store 90 days of transaction logs on an in-store server? From an electronic perspective, remove all cardholder data older than 2 or 3 days from your POS controllers. POS terminals should never store this information once it has been passed to the controller (which could arguably be done on a daily batch basis). If you feel that you may need this data, collect the transaction logs or electronic journals in a central location. Bringing data from 50 stores is much easier to maintain in one place versus 50 individual places.

Next, look for paper data. If you are still printing the entire card number on the receipt that a customer signs, all of that paper must be protected in accordance with PCI. Get it centralized, then possibly imaged and destroyed. Even if you make a change to your process to mask that number when it is printed, be sure you don't have any legacy data in the stores. One of the authors remembers visiting a retail location that had 10 years of paper cardholder data in clearly labeled boxes next to the bathroom used by customers, even though corporate policy prohibited keeping data any longer than 1 year.

TOOLS

Did you know that you only need four elements to uniquely identify any transaction in your enterprise, and one of those is not the full card number? These elements are as follows:

First six and last four (or just last four) digits of the card number,

Date and time of purchase,

Amount of purchase,

Authorization code.

Customers who have used this method have never reported that two transactions matched these elements identically but had different card numbers (including the largest merchants in the world).

Do your retail stores still contain knuckle-busters, those old manual credit card contraptions that used carbon paper and made a “kerCHUNK-kerCHUNK” sound as an imprint of a customer card is made? If so, you better believe there is probably someone who has used it recently and that some data is stored on hardcopy media (i.e., paper) in that store. Just like above, be sure that once you fix the policy or remove that equipment from the store that you have removed all the legacy data.

Here's another one that you may not have thought about. If you have certain business or high volume customers that phone or fax orders in, how do they pay for their orders? Do you keep a credit card on-file so that when they come in they can just sign and leave? Work on removing that data or changing how you deal with your high-profile customers.

If you run a call center, how are calls monitored? Do the phones rely on Voice Over IP (VoIP) technologies to operate? Do you record the calls? The Council has specific Frequently Asked Questions dealing with call centers on their Web site (www.pcisecuritystandards.org) that will address your particular situation.

Finally, look to your corporate headquarters. Do you really need a credit card more than a couple of months after the transaction was initially processed? Many companies tell you that they absolutely need it until you ask them why at least three times. Why is three the magic number? Who knows, but the truth usually sounds like “Well, we've always done it that way.” Then, you ask them what they might do if the data was not available after 60 days. They will usually figure out a way to either handle the dispute with a truncated number or simply realize that the cost to secure this data far exceeds the potential losses associated with not having the data.

NOTE

Propaganda is powerful. Some industry pundits think that the card brands require merchants to store data. If you have heard this, read the next sentence very carefully. Card brands do not require that you store data! In most cases, your acquirer is taking a short cut, thus transferring risk to the merchant! In the authors' experience, companies can easily deal with dispute resolution without the full number when they press their acquirer. Acquirers will compete for your business, so if your current acquirer is not willing to help you, consider another one! You'd be amazed how quickly something like this is resolved. Furthermore, law enforcement typically provides you with the full card number they want you to pull transactions for, so you can still assist them by asking for specific dates and times. There are some exceptions to this rule, but virtually everyone can be altered to off-load risk from the company trying to comply with PCI DSS.

TOOLS AND BEST PRACTICES

Aside from challenging the useful life of data and ensuring that you do not retain data longer than is absolutely required, here are a few other best practices you might consider.

Enforcement of a password policy in conjunction with other factors of authentication helps to protect systems from potential compromise. Here are some simple password rules, above and beyond changing default passwords that will provide stronger security. As this book is going to press, we are seeing more and more security professionals calling for the death of passwords. Since we may not see wide-scale adoption of alternative methods in the super near future, some of these tips may be useful to you:

- Accounts that have system-level privileges must have a unique password from all other accounts held by that user.
- Give administrators different accounts for administrative actions; do not tie the privileges to their primary domain accounts.
- Do not transmit passwords over the Internet by e-mail or any other form of communication without being encrypted.
- Allow users to craft longer passwords in conjunction with some element of password complexity to ensure strong passwords.
- Deploy a token-based, system-wide two-factor authentication solution such that any system access to sensitive information requires a token or another factor of authentication. Even using risk-based models can make a tremendous difference.
- Deploy a single sign-on solution that synchronizes passwords so that users do not have to remember multiple passwords, thus encouraging users to select more secure passwords.
- Do not share or write passwords down.

Many of these methods can help you bolster your overall security, but in many cases, they also make authentication easier on users. After all, if security professionals impose difficult requirements on users, they will come up with ever more creative ways to get around them!

RANDOM PASSWORD FOR USERS

PCI requires unique first-time passwords. There are many possible ways to do this, and a quick Google search for password generators returns many options. These may work well for you; however, if you're ultraparanoid, then you may want to use one installed on your own computer.

Here is a short Ruby (www.ruby-lang.org) script to help you create good, first-time passwords. Notice that certain letters and numbers will never be used in passwords that this program creates. For example, 1, l, and I are not included because they can often be mistaken for each other. Also 0 and O have been removed. To run this script, you must have Ruby installed (it runs on many operating systems including Windows and Linux) and have the Crypt::ISAAC module, which is a more secure random number generator than the one included with Ruby. This can be found at <http://rubyforge.org/projects/crypt-isaac/>.

```
#!/usr/bin/env ruby
require "crypt/ISAAC"
rng = Crypt::ISAAC.new
schars = "24356789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
for i in 1..10
  password = (1..10).collect { |i| schars[rng.rand(schars.length), 1]}
  puts password.join
end
```

COMMON MISTAKES AND PITFALLS

This section covers many pitfalls and mistakes related to this PCI domain.

POOR DOCUMENTATION

This domain of PCI DSS starts with documentation, and poor documentation will set you up for failure when you try to meet the rest of it. Companies that struggle the most with this domain have two major issues they fight. The first is a poor analysis as part of Requirement 7, thus poor (if not nonexistent) documentation to support this requirement. Because content in Requirements 8 and 9 often rely on Requirement 7 to be completed, companies that often miss parts of Requirements 8 and 9 have set themselves up for failure by missing Requirement 7. Do your homework, and make sure Requirement 7 is handled well!

LEGACY SYSTEMS

The other big trip up is legacy systems. There aren't as many of them out there as there used to be, but with a new group of companies looking for the services of a

QSA for the first time, there will be no doubt a ton of these out there. Legacy systems have many issues complying with PCI DSS, and this is one of the major ones.

Is there an avenue for a compensating control? There is always a possibility for a compensating control unless you are dealing with Requirement 3.2. For these systems, you will most likely need to use network segmentation and Virtual Private Networks or a “jump” server environment like a Citrix box or a Windows Terminal Server. Keep in mind, systems that are this legacy tend to be riddled with holes and struggle with just the basics of limiting network traffic to them. Most companies soon discover that this option will break their business, and they must ultimately upgrade.

PHYSICAL ACCESS MONITORING

For Requirement 9, the biggest mistakes people make are on camera coverage. PCI DSS version 3.0 continued the legacy of 2.0 to clarify where cameras are required, and what you must do with them. Don’t over cover (i.e., put a camera on every cash lane), and don’t under cover (i.e., no cameras in stores at all).

CASE STUDY

The following two case studies explore when bad things happen (or could happen) to people with good intentions. The PCI Requirements, when followed correctly, are designed to reduce the risk that companies carry by holding this data. The moral for both of these case studies is that you should store the absolute least amount possible, but protect what you store like a mama bear protects her cubs.

THE CASE OF THE STOLEN DATABASE

Alice’s Activity Atrium is a startup that aims to revolutionize how families shop with young children in malls. Alice pitched the concept of an activity center inside her local mall, with a safe, indoor playground, physical security around the play area, and pagers to reach parents inside the mall if problems occur with their children. Her goal was not to take up a store front but to make use of existing atrium space in an aesthetic manner such that it could be part of a seasonal offering for malls. The general manager for the mall granted her proposal on a trial basis, allowing her to set her business up for the back-to-school rush. Provided that things went smoothly, she would win the contract for the coveted Christmas season shopping that often occurred during November and December.

Alice bought some day care management software for her laptop based on Microsoft Access, and opened her business in late July with amazing success. Her quick rise in popularity required her to hire some additional help. Alice contracted Darlene to run the laptop during peak times so that Alice could help with keeping the kids busy and safe. The software on her laptop captured relevant information about each customer and captured a credit card for payment. Customers paid by the tenth of an

hour for the service, so the card was preauthorized for a 6-hour stay, much longer than any stay she envisioned, and then finally authorized and settled for the actual amount used. The laptop was placed at her location's entrance so that customers could check in and out quickly without having to step inside the facility, and repeat customers quickly were able to drop their kids off because the information was retrieved from storage.

One evening, one of the children under Alice's care fell at an awkward angle and sprained his ankle. During the commotion of moving the child to a safe place and calling the mall's medical personnel and the child's parents, Darlene left her post to help out. A passing thief seized a rare opportunity and stole Alice's new laptop.

Some readers may stop there and say, well, it's just a stolen laptop and that's not a big deal. Except, the laptop contained customer data on it, including credit card data. Although this is definitely a small-scale breach, the concept could also occur at a well-established business (say at a day care or jewelry store) with thousands of customers. Alice made three key mistakes when she set up her business.

First, Although Alice took every precaution to prevent a child from "escaping" from her area, she didn't lock down the physical assets that ran her business. A laptop out in the open of a crowded mall won't last long before someone picks it up. Alice should have physically secured the laptop such that it was not openly available for a thief to steal it during a crisis.

Second, Alice's desire to make it easier for kids to be dropped off had good intentions, but fell short when she stored the cardholder data of all of her customers. Instead of storing all that data, Alice should have instructed Darlene to ask for the method of payment for every transaction. Although this request would force her customers to spend another 1–2 min per transaction, she would not put her customer data at risk, and this breach's reach would be much smaller.

Finally, the software Alice used to manage her business did not protect the information it stored. Because it was written in Microsoft Access, the data is easily retrieved. The thief not only had accesses to each of her customers' credit card numbers but also had their home addresses and medical insurance information too! Should the thief figure this out, he has more than enough data to begin the process of stealing an identity or fraudulently charging transactions.

THE CASE OF THE LOOSE PERMISSIONS

Melissa works in the marketing department for Ben's Body Boutique, a regional fitness and health center. Ben charged Melissa with merging data from a mass mailing campaign to purchase records from the last 6 months. Ben wanted to determine how effective the original marketing campaign was, and if there were certain geographic areas where the response was higher, so he could concentrate future outreach programs on those areas that yielded the largest amount of business.

Melissa made a request to IT to get access to the current customer database, and through her basic database skills, she learned as a business analyst in a former job, and she was able to perform direct queries against the data. IT gave Melissa a

different username than her normal username, and she had to use a different password from her current Windows password because the complexity requirements did not match. She had a hard time remembering the password, so she temporarily wrote it down on a sticky-note and placed it inside her closed laptop.

Melissa began performing some basic queries based on the database layout provided by one of the DBAs. The data map seemed to be missing a few key elements that she needed to complete her analysis, so she tried calling the DBA. He was out sick, and nobody else was available to help her with her question. So relying on her previous experience, she ran a few commands to get a more complete layout of the schema. She noticed that most of the fields were ones she thought she would need, so to save time, she used a wildcard when listing the database columns she wanted to see, thus effectively dumping the entire contents of the table to the Excel spreadsheet she would use on her machine. Unfortunately, her wildcard selection pushed customer credit card data down to her machine without her immediate knowledge.

It was getting late in the day, and she decided to finish the analysis later that evening after she put her children to bed. Not realizing the data she had, she simply copied the customer data file and the direct mail list to a USB disk she kept in her purse, left her laptop at her desk, and headed home. After putting her kids to bed, she loaded the data on her home computer and started to perform her analysis.

Without going any farther into the story, let's stop for a minute and find the issues with this case, some of which happen every day in our companies. First, knowing Melissa's background, IT gave her a database account with unlimited select permissions on all the tables in the customer database. The IT department violated Requirement 8.7 by giving a user direct access into the database and not regulating her use to just the data she needed. Instead, they should have asked her for the requirements and performed the data dump themselves and further ensured that the data was not PCI related. Next, she wrote her password down because the authentication for the database was different from her main Windows account. This action should be prohibited by security and password policies. Finally, she took the data home and put it on a noncompany-owned or controlled PC. Although her home machine may have all of its patches and be up to date on antivirus, she may forget to delete the file and could open Ben's up for a breach further down the road if she sold the computer or fell victim to a Trojan horse or virus.

SUMMARY

Access controls are an extremely important part of protecting your data. It is important to understand your systems and the best ways to control access to your data. Once access controls are set, they must be constantly maintained to be effective. As we have talked about in this chapter, it's important to have many layers of security to be effective. Not only is it important to have strict access controls in place on computer systems, but it's also important to control physical access.

Spend time to fully define your access requirements for PCI so that you can comply with Requirement 7. Although this is the shortest of the three requirements we covered in this chapter, it may require the most amount of effort to effectively meet. Defining roles and access profiles takes time, but the work you put in will allow you to centrally manage a diverse set of permissions effectively and efficiently. Next, put those profiles into action and assign permissions to them on your systems. Put users into containers, and ensure that their access complies with the diverse requirements in Requirement 8.

Then, take a step back and focus on physical security, and don't forget stores and satellite locations! Don't defeat the corporate security at your headquarters by leaving wide open electronic access to your store or satellite locations. Finally, discover how much money you can save by destroying the data quickly after settlement. Remember, you don't have to protect what you don't store!

Protecting cardholder data

7

INFORMATION IN THIS CHAPTER:

- What is data protection and why is it needed?
- Requirements addressed in this chapter
- PCI requirement 3: protect stored cardholder data
- What else can you do to be secure?
- PCI requirement 4 walk-through
- Requirement 12 walk-through
- Appendix A of PCI DSS
- How to become compliant and secure
- Common mistakes and pitfalls
- Case study

The Payment Card Industry Data Security Standard (PCI DSS) was created to decrease the risk of electronic card transactions by mandating security controls at merchants and service providers; it is, thus, obvious that protecting the data is one of the key goals of the standard. Most of the 12 requirements cover data protection at least indirectly. We can even say, “it’s all about the card data”; however, there are two requirements that particularly apply to protecting card data that is stored (“data at rest”) or transmitted (“data in motion”) in your environment. This chapter covers those data security requirements, which are mostly related to avoiding the storage of data and use of encryption to protect the data you do store.

WHAT IS DATA PROTECTION AND WHY IS IT NEEDED?

Before we even start our discussion of data protection methods, we need to remind you that “the only good data is dead data.” Humor aside, dropping, deleting, not storing, and otherwise not touching cardholder data is the best single trick to make your PCI DSS compliance process easier. As a side benefit, the total avoidance of data will make your transaction processing less risky, reduce your liability and chance of fines, and prevent those breach losses.

NOTE

Many times, the easiest way to protect data is not to store it at all! It's a good idea to review the data you're keeping and verify that you really need to keep it. Most business processes dealing with cardholder data can be altered such that actual cardholder data isn't needed.

As mentioned above, PCI DSS requirements for protecting cardholder data encompass two elements:

- Protect stored cardholder data.
- Encrypt transmission of cardholder data across open, public networks.

The processes and activities necessary to meet these requirements and the specific subrequirements spelled out by PCI DSS are simply the implementation of some of the fundamental components of a sound information security program, just as with other PCI DSS controls. In this particular case, the controls are about protecting “data at rest” and “data in motion.”¹ Thus, it is possible to present PCI DSS data scope as a triad:

- Transmit,
- Process,
- Store.

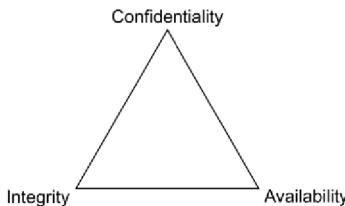
If you have already put into place the pieces of a solid information assurance program, or you are in the process of doing so, there won't be a great deal of extra work to do. Your current processes and technology may very well serve to quickly allow you to comply with these requirements without a great deal of additional effort or cost.

THE CONFIDENTIALITY/INTEGRITY/AVAILABILITY TRIAD

We defined the Confidentiality, Integrity, and Availability (CIA) triad in Chapter 6. Let's use them here to define our data protection efforts. These three tenets of information security are referred to as a triad because they are most commonly illustrated as three points of a triangle ([Figure 7.1](#)). All three principles must be considered as you manage your data. An undue degree of emphasis on one can lead to a deficiency in one of the others.

- **Confidentiality:** It strives to ensure that information is disclosed to only those who are authorized to view it. Most PCI DSS requirements apply to this leg of the CIA triad.
- **Integrity:** It strives to ensure that information is not modified in ways that it should not be. This can refer to modifications by either people or processes. While not directly applicable to PCI DSS, many operational PCI controls help to assure data integrity.

¹Sometimes “data in use” is added to this; in the context of PCI DSS, “data in use” often translated to payment application security, covered by PA-DSS.

**FIGURE 7.1** The CIA Triad

- **Availability:** It strives to ensure that data is available to the authorized parties in a reliable and timely fashion. PCI DSS does not play in this particular sandbox, with the exception of the “authorized parties” notion. The requirement to keep the data available for your business is on you and your organization. No external guidance from the PCI Council applies, even though many general security practices (such as Requirement 5 asking to deploy antivirus software) indirectly help in keeping the systems humming and data available.

In addition to the CIA triad, an auditability component is often added in. Specifically, in the case of PCI DSS logging and monitoring requirements (covered in Chapter 10) meant to provide auditing and monitoring for the infrastructure. This key tenet is about knowing who is doing what, with specific data, at any given time and being able to prove it via logging and monitoring.

Combining CIA with process-transmit-store, you arrive at the following complete structure for PCI DSS data protection ([Table 7.1](#)).

Table 7.1 PCI DSS Data Protection Mapped to CIA

	Transmit	Process	Store
Confidentiality	Don't transmit the data over less-secure networks, if possible. Encrypt data in transmission	Use secure applications (Payment Application Data Security Standard [PA-DSS], application security)	Don't store. If you have to, encrypt, mask, and truncate the data
Integrity	Don't transmit the data over less-secure networks, if possible. Encrypt data in transmission	Use secure applications (PA-DSS, application security)	Don't store. If you have to, encrypt the data. Other methods do not preserve data integrity
Availability	PCI does not apply. Your business does need card data availability to operate	PCI does not apply. Your business does need card data availability to operate	PCI does not apply. Your business does need card data availability to operate

Use this as a map for all relevant PCI DSS requirements covered in this chapter.

REQUIREMENTS ADDRESSED IN THIS CHAPTER

This chapter addresses the following PCI DSS requirements.

- Requirement 3: Protect stored cardholder data.
- Requirement 4: Encrypt transmission of cardholder data across open, public networks.
- Requirement 12: Maintain a policy that addresses information security for all personnel (only sections relevant to data protection).

Other chapters, such as Chapter 5, Chapter 6, Chapter 9, and Chapter 10, cover additional requirements relevant to data protection as well:

- Requirement 7: Restrict access to cardholder data by business need-to-know (see Chapter 6).
- Requirement 10: Logging access to data and monitoring the environment (see Chapter 10).

PCI REQUIREMENT 3: PROTECT STORED CARDHOLDER DATA

The most effective means of ensuring that cardholder data is not exposed to unauthorized parties (confidentiality) is the proper destruction of that data.

There is no mistake here! Please keep this thought in mind. The best means of avoiding that data falling in the wrong hands is not having such data at all. In fact, this is not just the best way; this is the only way that 100% guarantees it. This fact also highlights that PCI has nothing to do with data availability for your business processes (whereas the final revision of the Visa Cardholder Information Security Program [CISP] standard required business continuity and disaster recovery procedures). It reminds that if you can avoid storing and moving the data around, you can be safe from a lot of trouble.

As a result, before you engage in any data protection project involving encryption, masking, sanitization, or other processes that might scrub cardholder data, consider performing a project that investigates removing the data from as many systems as possible. The author team is well aware of the fact that it is not possible for all organizations and every circumstance; thus, this chapter continues.

By the way, we are not talking about running the commands that are as follows (both of which leave artifacts of the original data behind):

```
DEL credit_card_data.dat (Windows)
```

Or

```
/bin/rm -rf /opt/credit_card_data.data (*NIX)
```

We are talking about assessing how your organization processes payment cards and whether any parts of that business process can be simplified and improved by not storing card data, storing it in smaller number of places, or not storing full primary account number (PAN, which is allowed as per PCI, but definitely not recommended for scope considerations). Storing some types of data is expressly prohibited by PCI DSS, and thus, no protection methods are required; such data simply must not exist in your environment.

The second-most effective means of ensuring that stored cardholder data is not exposed to unauthorized parties (confidentiality) is the encryption of that data. When implemented properly, the value of encryption is that even if an intruder is able to gain access to your network and your data, without access to the proper encryption keys that data is still unreadable.

PCI standards dictate that stored cardholder data can be rendered unreadable in a number of ways, such as encrypting, masking, truncating, or tokenizing. Encryption will protect your data from being used by the malicious hackers, and thus, the spirit of PCI DSS—to reduce the risk of performing payment card transactions—will be preserved.

Should you be unable to protect the data with strong cryptography, PCI DSS allows you to implement compensating controls to meet this requirement. See Chapter 16 for more details on selecting compensating controls for encryption. Because encryption is such an effective and critical part of protecting data, we will discuss some of the details of encryption methods and the associated advantages and disadvantages.

REQUIREMENT 3 WALK-THROUGH

Let's walk through all the subrequirements of Requirement 3, "Protect stored cardholder data."

First, PCI DSS Requirement 3.1 highlights the mantra about not storing data: "Keep cardholder data storage to a minimum." PCI DSS 3.0 simplified this requirement to push the testing procedures into subrequirements 3.1.a–c. It further specifies that an organization needs to limit storage amount and retention time to that which is required for legal, regulatory, and business requirements. If you expect to use legal or business requirements to expand your retention, you should have very detailed documentation (probably with expert help) explaining why. You would be better off deleting and destroying!

Often we would have merchants tell us they needed to store full PAN data for chargebacks; this is not entirely true. Many acquirers have other ways of handling chargebacks without mandating full card data storage, and every acquirer that one of the authors worked with was able to provide an alternative method for chargeback disputes. Remember, keep cardholder data storage to a minimum. Better yet, don't store it at all.

There is a common misconception about the storage of PAN data proliferated by industry pundits. If your acquirer or processor is requiring that you store the PAN

data, keep in mind that is *their* requirement to fulfill, not a PCI DSS requirement for a merchant. If all else fails, choosing another acquirer is definitely within your rights. Before proceeding to the next part of Requirement 3, review all of the testing procedures for Requirement 3.1.a–c to ensure you are prepared for success. In PCI DSS 3.0, your assessor should verify that nothing exceeds the retention requirements set out in your policy (Requirement 3.1.c).

The next subrequirement flat-out bans the storage of some types of data (3.2): “Do not store sensitive authentication data after authorization (even if encrypted).” I don’t think this could be made any clearer in PCI DSS. Do *not* ever store sensitive authentication data after authorization. This requirement is also one of the key components of Phase 1 of the PCI DSS Prioritized Approach. Requirement 3.2 finally includes guidance for Issuers, so pay attention to the wording of the requirements so you know which pieces you have to comply with.

Both authors are really tired of hearing the issuer question at the PCI Community Meetings, so hopefully this will create an avenue for issuers to comply with PCI DSS now. There is also some more EMV-friendly language around equivalent magnetic stripe data that might be found on a chip. You can also expect QSAs to ask to poke around files containing incoming transaction data, logs, history files, trace files, database schemas, and even the contents of many database tables looking for this information. If your QSA does not ask for these things, beware, you may have a QSA that is not doing a complete job.

So, what data can *never* be persistently² stored if you are to have any hope of PCI DSS compliance for your organization? The answer is easy:

1. Full magnetic stripe data or equivalent on a chip.
2. CAV2/CVC2/CVV2/CID code, a 3- or 4-digit value printed on the card.
3. Personal identification number (PIN) or the encrypted PIN block.

For example, here is a reference from PCI DSS documents ([Figure 7.2](#)):

It shows that such sensitive data must not be stored. Remember, if you are persistently storing any of the above (full track, CVV2, PIN), you are not PCI DSS compliant and cannot be PCI-validated. Above all, in the US states where PCI DSS provisions are encoded as law, it will get you in trouble with authorities as well.³ For more information, see Visa’s famous Drop The Data site at www.visa.com/dropthedata [1].

This requirement results in a simple action before even looking into encryption technologies at all: find out if you have such data stored.

²Temporary memory storage is explicitly allowed by one of the PCI DSS clarification notes. In addition, the Guidance for Requirement 3.2 says that issuers are the only entity that may store this data if there is a legitimate reason to do so (i.e., necessary for the performance of whatever function the issuer is doing, and not for convenience).

³Not legitimately, at least! However, it is the authors’ sincere belief that lying about it can only get your organization in trouble with your acquirer, card brands and, ultimately, with your customers and the public. Above all, in the US states where PCI DSS provisions are encoded as law, it will get you in trouble with authorities as well.

	Data element	Storage permitted
Cardholder data	Primary account number (PAN)	Yes
	Cardholder name ¹	Yes
	Service code ¹	Yes
	Expiration date ¹	Yes
Sensitive authentication data ²	Full magnetic stripe data ³	No
	CAV2/CVC2/CVV2/CID	No
	PIN/PIN block	No

FIGURE 7.2 Banned Data**NOTE**

Track 1 contains the following data, as per ISO/IEC 7813:2006:

SS | FC | PAN | FS | CC | Name | FS | Additional Data | CC | LRC

The field abbreviations are as follows:

SS: start sentinel

FC: format code

PAN: primary account number

FS: field separation

Name: primary account holder name

FS: field separation

Expiration date, offset, encrypted PIN, and so on.

ES: end sentinel

CC: country code (three characters minimum)

LRC: longitudinal redundancy check

Track 2 contains the following data, as per ISO/IEC 7813:2006:

SS | PAN | FS | ED | SC | Other Data | ES | LRC

The field abbreviations are as follows:

SS: start sentinel

PAN: primary account number

FS: field separation

ED: expiration date

SC: service code

Other data

ES: end sentinel

LRC: longitudinal redundancy check

If there happens to be an active business process that results in such data being stored or that relies on having such data, adjust it:

- Destroy the data.
- Make sure that no accidental/undocumented storage is taking place.

As for a real-world example, remember that the storage of prohibited data killed CardSystems back in 2005 (or at least was a contributing factor in its demise).

TOOLS

Now you know that you cannot store sensitive authentication data, but what does it look like? The best thing you can do is search for valid PANs first and then manually verify what data is stored around it. Card verification values that appear on either the front or the back of the physical cards can be a challenge to find programmatically. Track data is easier as it is a formally defined standard. For more information, point your browser to your favorite search engine and enter “ISO/IEC 7813” for the layout of track data. Also, a brief introduction on card track data as described in ISO/IEC 78xx family of standards is presented in the above note.

PCI DSS 3.0 clarified Requirement 3.2 to require that any secondary authentication data is rendered unrecoverable after the authorization process. This might be the single best enhancement for PCI DSS 3.0 by attaching the intent to the wording. Words like secure delete, destroy, and remove can be open for interpretation, but the goal is to ensure the data cannot be recovered.

Requirement 3.3 covers processes and procedures around displaying the PAN. Given that an accidental disclosure of the number can lead to card fraud just as well as its theft, the DSS mandates that organizations “Mask [the] PAN when displayed.” Showing the first six (Bank Identification Number [BIN]) and last four digits is preferred to showing the full PAN under this requirement. It goes without saying that this guidance cannot be mandated for the employees that need to see a full account number for their business functions, but nowadays those employees are few in numbers (and getting smaller by the day).

Requirement 3.4 mandates that you “render PAN[s] unreadable anywhere [they are] stored.” This is another reminder to you that not storing the data will make PCI DSS easier for your organization; clearly, if it is not stored anywhere, this requirement would not require any action on behalf of your organization.

This requirement, by the way, does not simply mandate encryption. It allows any of the following to be used:

1. One-way hashes based on strong cryptography.
2. Truncation.
3. Index tokens and pads.
4. Strong cryptography with associated key-management processes and procedures (covered in Requirements 3.5 and 3.6).

All the above are equally acceptable. Some are harder to accomplish, some are harder to maintain, and some just flat won’t work for your business. Remember, if it is not stored, it cannot be stolen (are you sensing a theme yet?). You will gain significant ground by altering your business processes in such a way that cardholder data is not needed. We cover encryption in the next few sections in more depth.

ENCRYPTION METHODS FOR DATA AT REST

Data at rest encryption options can be broken down into three high-level categories:

- File- or folder-level encryption.
- Full-disk encryption (FDE).
- Database encryption.

Let's examine the advantages and disadvantages of each as you consider how and where they might fit into your program for protecting cardholder data.

File- or folder-level encryption

File- or folder-level encryption (or file system level) is an encryption system where specific folders, files, or volumes are encrypted by a third-party software package or a feature of the file system itself.

Advantages

- You have more granular control over what specific information must be encrypted. Card data files that you need to encrypt can be stored in a particular folder or volume, and data that does not need to be protected can be stored elsewhere. For example, some smaller organizations that do periodic billing actually use this method to encrypt all the numbers between the billing runs, thus satisfying certain PCI DSS requirements.
- Many file-level encryption products allow you to integrate access-level restrictions. This allows you to manage who has access to what, and can extend roles-based access controls making large-scale management scalable. This helps satisfy data protection and access control.
- Some file-level encryption systems offer the capability to track who attempts to access a file and when. In order to satisfy the PCI DSS logging requirements, your file-level encryption product must allow you to granularly log information about their use to satisfy Requirement 10.
- When there is a need to move the data, data can be encrypted on a file level and then moved off of the storage location. Don't forget to destroy the original data! This maintains the confidentiality of the data when it is moved to a backup medium. Remember that any media lost with cardholder data on it still constitutes a "breach" and must be reported.
- File-level encryption tends to consume less resource overhead, thus less impact on system performance. Modern operating systems can perform efficient file encryption on the fly.

Disadvantages

- Performance issues can be caused for backup processes, especially with relational databases.
- Extra resources for key management are required since more keys may need to be managed.

Windows Encrypted File System (EFS) with Microsoft operating systems is the primary example of such technology. Remember, if you deploy this type of encryption, you will need to ensure that the decrypting credentials are different from your standard Windows login credentials (Requirement 3.4.1). Additional encryption products can be used as well. Here are some of the common free or open-source file encryption products, found in wide use:

- GNU Privacy Guard (GnuPG or GPG) from Free Software Foundation can be found at www.gnupg.org. It performs efficient file encryption using symmetric and public key cryptography and works on Windows and Unix operating systems.
- TrueCrypt is another free open-source disk encryption software for Windows, Linux, and even MacOS. It can be found at www.truecrypt.org. It can perform file, folder, and FDE.
- AxCrypt (www.axantum.com/AxCrypt/) is another choice for Windows systems. It is also free and open-source.

Encrypting individual card data files is free and easy with the above tools. As with other domains, PCI DSS never mandates the use of specific tools or vendors.

Full-disk encryption

FDE or “whole-disk” encryption methods encrypt every file stored on the drive (or drives), including the operating system/file system. This is usually done on a sector-by-sector basis. A filter driver that is loaded into memory at boot encrypts every file as it is written to disk and decrypts any file that is moved off of the disk. This happens transparently to the end-user or the application generating the files.

Advantages

- Everything on the drive (or drives) is encrypted, including temporary files and swap space, increasing security of all your data, not just card data. If deployed on all in-scope systems, the card data would be guaranteed encrypted.
- Encryption of data is forced on end-user, alleviating decisions on what or what not to encrypt.
- Encryption/decryption is transparent. When information needs to be accessed, it can be saved off the system and is automatically decrypted. If a processing application is installed on the system, the use of encrypted data is also easy.⁴
- Since all data on the drive is encrypted, even if an alternative boot medium is used against an encrypted system, the data on the drive is unreadable and therefore useless to the thief. Thus, card data is protected even when the system is turned off.

Disadvantages

- Some FDE programs can cause an increase in data access times. Slight delays in writing and reading data can occur, especially with very large files and high transaction volumes.

⁴However, in this case, the unencrypted card data may be stolen while in use. Such data theft has been reported to be used by attackers for some recent card processor breaches. Neither “data in motion” nor “data at rest” encryption techniques could have helped for that case.

- System password management and key management processes have to be defined and put into place. If a user loses his password that grants access to the encrypted system, he has no access to his data at all. Key management procedures defined in Requirement 3.5 are more critical for FDE. By the way, as per 3.4.1, “Decryption keys must not be tied to user accounts!”
- For data centers, this technology is largely useless for security (as opposed to laptops that may be stolen or lost in the field) since most data centers have significant physical controls keeping their hardware safe.
- FDE does not necessarily protect data on a laptop if the system is compromised while in use. It primarily helps to prevent data disclosure resulting from physical theft.
- Some FDE implementations leverage Windows AD credentials. Be careful deploying such systems as it would violate Requirement 3.4.1.

FDE is more suited to protecting data on workstations and mobile devices, whereas file-level encryption is more useful as a method on large-volume storage devices. The much publicized cases of database managers or analysts putting thousands of clients at risk because a laptop was stolen that had been used to download large volumes of sensitive data from a storage device only serve to demonstrate this fact.

In [Figure 7.3](#) illustrates the difference in architecture between file-level encryption and FDE.

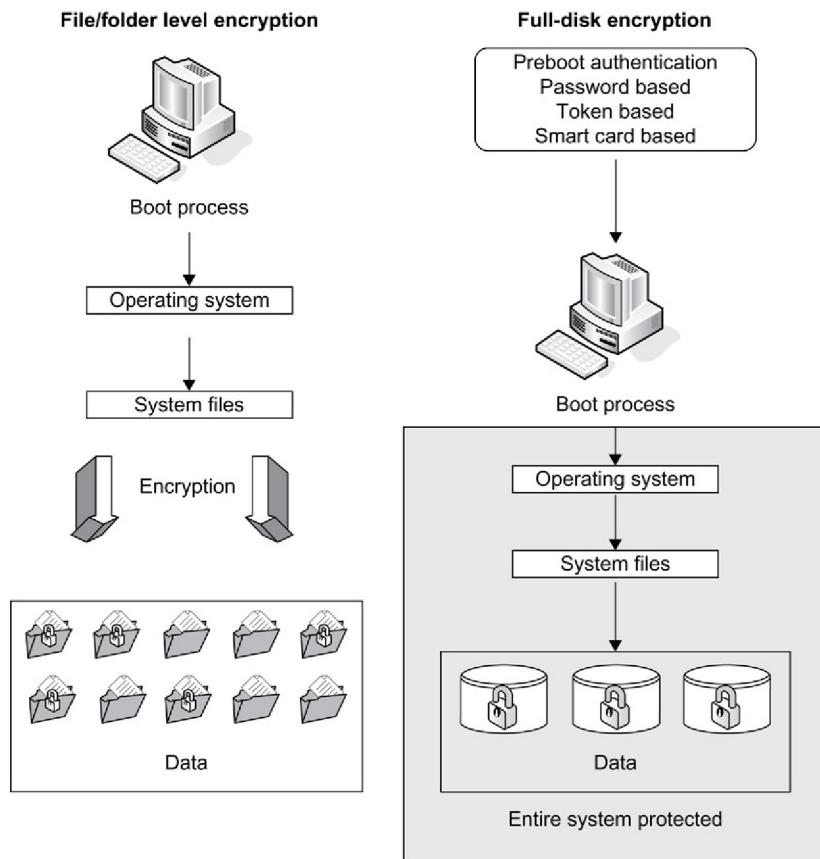
BitLocker Drive Encryption, included with the newer Microsoft operation systems such as Windows 7, is the primary example of such technology. Additional encryption products can be used as well. For example, TrueCrypt is a free, open-source disk encryption software for Windows, Linux, and even MacOS (which also natively includes FileVault), which can perform FDE. It can be found at www.truecrypt.org. The latest Pretty Good Privacy (PGP) Whole-Disk Encryption (www.pgp.com) is not free but is found in frequent use.

NOTE

Before you reach out for the encryption tools, remember and repeat the mantra: “Do I need to keep this data?” Even with “free” tools there are management costs to operate and maintain them in your environment. The best bet is to never carry the data in the first place.

Database (column-level) encryption

The most sensitive piece of cardholder data that is allowed to be stored is a PAN. Think of this as your crown jewel. This is the full card number that identifies both the issuer of the card and the cardholder account. PCI DSS 3.4 states “Render PAN unreadable anywhere it is stored.” If PANs are stored in a relational database and not in files, the column-level encryption becomes the only approach for rendering the key cardholder data unreadable.

**FIGURE 7.3 File-Based Encryption versus Full-Disk Encryption**

Advantages

- When a table is queried for data in an unencrypted column, no performance impact is seen. Since no decryption activity is taking place, no delay in reading/writing and no performance hit by system because encryption software activity is seen.
- When a query for a record with data from an encrypted field is performed, the overhead is minimal. Since the decryption activity only has to take place on the individual field or fields that are encrypted, there is much lower overhead.
- It can be used in conjunction with other controls to protect data from administrators. Separation of duties between security and database administrators (DBAs) reduces the risk presented by allowing a DBA unlimited access to the data you need to secure for PCI compliance.

Disadvantages

- Database encryption requires tight integration with the database and may need to be purchased separately from a database vendor.

- It is often highly invasive to the database design. To implement column-level encryption protection after the fact, you may have to change the following (depending on implementation):
- Data type of the field being encrypted.
- References to and queries of the encrypted field(s) will have to be modified to limit access. Middleware and other applications that interact with the database will have to be comprehended and possibly reconfigured.
- Key management has to be well planned; if the encryption key is hard-coded into scripts, it defeats the purpose of securing the data and violates Requirement 3.5. Keys themselves must be stored in an encrypted state and access controls placed around them.
- Merchants and service providers who perform batch processing will commonly end up storing sensitive data in flat files exported from a database. In this case, database encryption has to be combined with file or folder encryption.

As a result, column-level database encryption might be the answer for a piece of your overall plan for compliance to protecting cardholder data, but it is unlikely to be the entire plan.

At the time of writing, most major relational database vendors offer some form of database encryption. In particular:

- Oracle database (www.oracle.com) offers multiple type of encrypted tables, including “transparent encryption” that can be integrated with applications. See [2] for more information.
- IBM DB2 database (www.ibm.com/db2) offers data field and column encryption as well. See [3] for more information.
- Microsoft MS SQL Server (www.microsoft.com/sqlserver) offers data encryption as well.

Free open-source database MySQL (www.mysql.com), now owned by Oracle, offers nontransparent data encryption using the Advanced Encryption Standard (AES) cryptographic algorithm, and free open-source database PostgreSQL (www.postgresql.org) offers a multitude of options powered by the pgcrypto() function.

As before, choose the solution that fits your overall IT strategy; you will likely not need to switch database vendors to fulfill your PCI obligations.

WARNING

Don't forget about portable storage devices that attach to laptops or desktops. There are some software-based solutions that can be configured to enforce encryption on any attached USB device, sometimes even based on the type of data being copied. Other USB devices have built in biometric readers tied to the issued-user's thumbprint. While some solutions may be difficult to manage, this can also protect you from having your expensive encryption solution undone by a careless employee who uses a nonprotected USB drive to transfer or store payment data. Overall, minimize the use of portal devices to transport card data.

PCI AND KEY MANAGEMENT

Apart from making data unreadable, PCI DSS mandates certain key management practices if encryption is your chosen method of rendering data unusable. After all, the only thing that makes encryption a data protection mechanism is an encryption key. Let's continue to review PCI DSS Requirement 3.

Requirement 3.5 discusses the protection of encryption keys. PCI DSS details several different items for the proper management of encryption keys. They include processes, procedures, and the custodian of these keys. The management of encryption keys is probably the most resource-intensive aspect of encryption as well as the most error prone. Critical key management practices to keep in mind are as follows:

- “Restrict access to cryptographic keys to the fewest number of custodians”—only let those who need to see the key access it; this means that not everyone who needs to see card data needs to be in possession of an encryption key.
- “Store secret and private keys [...] [securely] in one or more of the following forms”—go read the requirement to understand those forms, but essentially, you need to store keys encrypted on drives or use some kind of hardware security module to protect the keys. Keys must be protected from online and offline attacks. Requirement 3.5.3 now specifies that keys must be stored in the fewest possible locations.
- “Decryption keys must not be tied to user accounts” (Requirement 3.4.1)—your system password must not be your key to decrypt all the data in case of FDE.

The next subrequirement (3.6) addresses the policy angle of the encryption; namely, “Fully document and implement all key-management processes and procedures for cryptographic keys used for encryption of cardholder data.” PCI DSS explicitly states what practices must be documented:

- Generation of strong cryptographic keys.
- Secure cryptographic key distribution.
- Secure cryptographic key storage.
- Periodic cryptographic key changes.
- Retirement or replacement of keys as deemed necessary when the integrity of the key has been weakened or keys are suspected of being compromised.
- Split knowledge and establishment of dual control of cryptographic keys—keep in mind, this is for manual clear-text crypto key management as automated key management systems have their own methods for doing this.
- Prevention of unauthorized substitution of cryptographic keys.
- Requirement for cryptographic key custodians to sign a form stating that they understand and accept their key custodian responsibilities.

However, detailed coverage of cryptographic practices goes far out of scope of this book. For more information on key management best practices, see NIST’s special publication SP 800-57, Recommendations for Key Management (as well recent addendums) at <http://csrc.nist.gov/publications/PubsSPs.html>.

WHAT ELSE CAN YOU DO TO BE SECURE?

While encryption is one of the natural ways to protect stored data, it is not the only method to protect data when stored. In some special cases, mainframe systems with strong access controls may be permitted to meet Requirement 3.4 without deploying encryption technologies. These circumstances are specialized, and typically require no direct user access, encrypted connections into the mainframe, and storage limited to only a few small areas.

If you think you may fall into this category, contact a QSA and walk through the scenario to see whether they believe that it stands up to the definition of a compensating control.

PCI REQUIREMENT 4 WALK-THROUGH

As in the case of protecting stored data, the most reliable and efficient way to ensure that your transmitted data is not intercepted (confidentiality) or modified (integrity) is to not transmit it anywhere. At the very least, avoiding public and insecure networks will go a long way toward achieving compliance and risk reduction.

The next most reliable and efficient way to ensure that your transmitted data is not intercepted (confidentiality) or modified (integrity) is to encrypt it during transmission.

PCI Requirement 4 spells out some specific details as it relates to these procedures for communication. Let's take a look at some of the specific PCI DSS sub-requirements to illuminate some of the terminology and the implications.

Requirement 4.1 states “Use strong cryptography and security protocols” such as Secure Socket Layer (SSL), Transport Layer Security (TLS), Internet Protocol Security (IPSec), or Secure Shell (SSH) “to safeguard sensitive cardholder data during transmission over open, public networks.”

An open, public network is essentially any network that contains a gateway to the Internet at large. This describes the interchange in pretty much every business network. Anytime your cardholder data is transmitted over the Internet or any network you are unsure is secure, that data must be protected. PCI DSS documentation explains that the Internet, any wireless network, Global System for Mobile Communications (GSM), and General Packet Radio Service (GPRS) network are to be considered “open and public,” while dedicated network relays are not.

Let's take a look at the specific protocols mentioned in PCI DSS and used for securing card data when transmitted over these various types of networks, and the way they are applied.

TRANSPORT LAYER SECURITY AND SECURE SOCKETS LAYER

TLS and the original Secure Sockets Layer (SSL) are cryptographic protocols that are used for transferring information over networks such as the Internet. They both

encrypt the data transferred between communicating endpoints, such as a Web browser and a Web server. The use of SSL or TLS is mandated by PCI DSS Requirement 4.1. Describing the technical differences between TLS and SSL is beyond the scope of this chapter.

NOTE

A merchant asked one of the authors some time ago: if I have an SSL certificate on an e-commerce Web site, am I PCI DSS compliant?

After recovering from shock (after all, there about 250 more requirements in PCI DSS!), the author was able to explain that SSL obviously does not guarantee PCI compliance. However, transmitting card data without SSL or TLS will certainly guarantee the absence of such compliance.

All modern Web servers such as Apache (www.apache.org) and Microsoft IIS (www.iis.net) have long offered native SSL support. Configuration of SSL on each individual server is beyond the scope of the book. Google offers many pointers to enabling SSL on your Apache Web server as does virtually every Certificate Authority (CA).

At the time of this writing, the Heartbleed (www.heartbleed.org) bug is causing havoc across the globe. As a reminder, software is written by humans and we are not infallible. Be sure to use as many layers of protection as you can to keep yourself safe.

Don't forget, per the Council, SSL v2.0, SSH v1.0, and TLS v1.0 are no longer acceptable and cannot be deployed on systems that are involved in card processing and on external systems. You must at least use TLS v1.1 or later and SSH v2.0 or later. Keep up with the emerging security protocols to make sure you are not caught by surprise.

NOTE

A PCI DSS requirement clarification from the PCI Council now mandates that all approved scanning vendors (ASVs) detect the use of older cryptographic protocols and identify this as a failure to PCI validation via scanning, leading to loss of PCI compliance. We cover PCI ASV scanning in Chapter 9. Specifically, the PCI Council stated it is imperative that an ASV identifies the use of older versions of SSL (version 2.0 and older) to transmit cardholder data as a failure. The PCI Council also clarified that the merchant can enable SSLv2 or even an older version for an initial handshake only to notify the user of the outdated browser that it needs to be updated, and then disallow access to the site until and unless the user updates his browser.

To resolve these failures for PCI validation, discontinue the use of SSLv2 and TLS v1.0 on all systems within the PCI scope. Having a user browse your site without that capability is not only rare, but you would be doing him a favor by directing him to a place where he can update his browser. There are some very specific instances where you may not be able to phase out the use of SSLv2 for in-scope systems (note, systems not in scope would be exempt from this requirement). Those should be discussed with your Acquirer.

IPSEC VIRTUAL PRIVATE NETWORKS

IPsec is technically not just a protocol, but a framework or set of security protocols for data protection with cryptography. IPsec is often used for client-to-site or site-to-site virtual private networks (VPNs). A VPN can be described as a network that uses public infrastructure (like the Internet) to create a connection between a remote host or network and an organization's main or home network. This is a much

less expensive proposition than using dedicated leased lines to provide this kind of privacy. VPNs set up a private “tunnel” using certain protocols, which causes the data to be encrypted at the sending end and decrypted at the receiving end. It can be configured in different ways, but typically involves the installation of connection software on the client, which establishes the secure tunnel to the home network and network devices on the home network end to serve as the secure gateway.

Another option for a VPN is an SSL VPN. The main advantage of an SSL VPN solution is that it does not require any additional or specialized software package on the client end. A modern, standard Web browser is all that is needed, which utilizes a small plug-in to the browser to configure it. Implementations of SSL-based VPNs have come a long way since our last edition. These VPNs now exist in consumer or SO/HO class devices as a matter of differentiation.

WIRELESS TRANSMISSION

PCI DSS 3.0 clarified more types of public networks and also lists a few types of wireless networks that you might not have thought of. Remember, security by obscurity is a victim’s security strategy.

In addition, Requirement 4.1.1 covers the transmission of card data via wireless networks. We cover wireless security in a dedicated chapter—Chapter 8. For now, it would suffice to say that wireless networks are becoming much more common for transmission of card data within stores, hotels, and other environments. At the same time, several of commonly used wireless encryption protocols such as WEP have been shown to be broken and must not be used.

Section 4.1.1 of the PCC DSS specifically states that the merchant must “ensure wireless networks transmitting cardholder data or connected to the cardholder data environment, use industry best practices to implement strong encryption for authentication and transmission.”

NOTE

Other protocols are mentioned in PCI DSS as well. GSM refers to the communication system that is used to support mobile phone networks. GPRS is a low-bandwidth wireless communication service that provides a connection to the Internet for data transfer for mobile phones and computers. Where this might affect a wireless network and transmission of card data would be the circumstances of using a GSM/GPRS modem card in a laptop, sled, or purpose built device for a connection to the Internet. If the requirements for the implementation of a VPN and wireless protocols have been observed, it will satisfy issues related to these cards as well.

The PCI Council released the “PCI DSS 2.0 Wireless Guidelines” information supplement in August 2011 (still current as of this writing) which updates the original 2009 release to both line up with PCI DSS 2.0 as well as add new guidance for some emerging technologies to help merchants manage the security of their wireless requirements. Not too much has changed around wireless from 2.0 to 3.0, so until the

council updates all of their supplemental documentation, you can use this document. The document can be found at www.pcisecuritystandards.org/pdfs/PCI_DSS_v2_Wireless_Guidelines.pdf.

NOTE

One of the largest cardholder data breaches at TJX involved insufficient wireless security! In January of 2007, TJX Companies, which is the owner of several retail stores including TJ Maxx and Marshall's, reported a very large data breach of customer credit and debit card numbers that occurred as early as 2003 through the beginning of 2007. TJX reported the theft of at least 94 million credit card numbers. Attackers were able to steal the data through an insecure wireless network at a Marshall's store in Minnesota. The Marshall's wireless network, which connected their credit card processing hardware to the company's back-end systems, was not protected with Wi-Fi Protected Access (WPA) encryption, but rather was still using the unsafe and outdated WEP standard. Despite the fact that the WPA standard was introduced in 2002 and TJX had their back-end systems protected, this vulnerability gave TJX a very dubious distinction of facilitating one of the largest cardholder data breaches on record.

MISCELLANEOUS CARD TRANSMISSION RULES

Finally, our Requirement 4 walk-through ends with what is simply an embodiment of common sense—Requirement 4.2: “Never send unprotected PANs by end-user messaging technologies (e.g., e-mail, instant messaging, chat).”

This one is a true “no-brainer” requirement with nothing else to add. Just remember and follow this one. Never e-mail plain card numbers—and never deal with anybody who does. By the way, this applies to attachments as well, not just to plain text e-mails.

Some merchants have asked about fax data transmission. It is obviously allowed but received faxes, especially if electronic, are subject to all the data protection rules, including physical security that governs data protection and access control.

Don’t forget to be sure your documents are in order for Requirement 4.3!

REQUIREMENT 12 WALK-THROUGH

Requirement 12, “Maintain a policy that addresses information security for all personnel,” indirectly plays toward protecting card data from the “softer” side via policies and procedures. Given that Requirement 12 is covered in PCI DSS document with plenty of detail in both the requirement text as well as the testing procedures, we are only providing a brief overview of all the requirements.

Requirement 12.1 starts us off, mandating we “Establish, publish, maintain, and disseminate a security policy.” Details about the policy are specified in the document; the main idea for compliance is that the policy exists, contains all the needed components, and is actively being used and updated on a periodic basis. Needed components should at least include or map to every policy required by PCI DSS (which is now essentially the last subrequirement in every major requirement).

Requirement 12.2 mandates an annual risk assessment, and an annual review process based on the output of the risk assessment, or at least as a part of a normal review process, to alter the policy based on changes to the business or the risk environment (Requirement 12.2.b). Remember, this policy may not need to be one massive document—it should be part of the policy framework in your governance structure. To make the document set more manageable, refrain from adding standards, guidelines, and procedures here. Stick to policy only.

While a deep dive into governance structure is out of scope of this book, let's take a moment to lay out some basic terms and what they mean. Policies are overarching documents that define roles, limits of governance, and other ways that a company or group may act. They should not change often and should remain relatively constant. If you are changing them more than yearly, you probably have tactical or operational details buried in the policy. Next is standards, which set forth specific mandatory controls or actions for how to operationalize a procedure. These can change with the times, but should further define the sandbox in which you can play. Next is guidelines which provide additional best practices for operations. These may change often depending on external forces that act on the company. And finally, procedures, which are step-by-step instructions on completing certain tasks. For more information on how these governance structures work, consider looking at CoBIT.

NOTE

Requirement 12.3 covers the need to “develop usage policies for critical employee-facing technologies.” For more information on Requirement 12.3, see Chapter 8.

The policy and procedures are further defined in the next few requirements. For example, Requirement 12.4 mandates you “ensure that the security policy and procedures clearly define information security responsibilities for all personnel.” The first step is to make sure that you don’t only account for employees and contractors in your policy. Every party should be included. Do you have an Acceptable Use Policy or an information security policy disclosure notice for visitors when they come on-site? That is one perfect way to make sure that visitors understand your policies and is a pretty easy way to show to an assessor that you are meeting this requirement.

Requirement 12.5, and its subrequirements, mandates assigning a particular team in your organization the responsibility for managing information security. You should have a formal chief security officer, or equivalent, with the proper authority to carry out information security management (Requirement 12.5). The delegated authority and responsibility should include creating information security policies and procedures (Requirement 12.5.1); monitoring, analyzing, and reporting information security alerts and metrics, and sending relevant information back to the business (Requirement 12.5.2); creating and managing the incident response and escalation procedures, with proper escalation to management or authorities as needed (Requirement 12.5.3); responsibility for administering user authentication and

authorization (Requirement 12.5.4); and monitoring and controlling all access to data (Requirement 12.5.5). Once companies start to get the hang of information security, this requirement is usually not a problem. How you operationalize these requirements may vary (e.g., compliance may design controls and IT/IS may implement them), but it's important that you have the basics covered somewhere in your organization.

Part of complying with PCI DSS means that you will complete information security awareness training annually for your employees (Requirement 12.6). You can use multiple methods for accomplishing this, but at a minimum, you must perform some kind of awareness training annually, and make your employees acknowledge that they have read and understand your security policy and procedures.

Human resources have their very own requirement! Requirement 12.7 mandates employee background checks (within the constraints of local laws) be performed on all employees *prior* to hiring them. PCI DSS gives some examples of what type of checks could be performed, but keep in mind this may vary from country to country, depending on the laws or customs in place.

One of the most critical requirements is 12.8, which covers your work with service providers. If any cardholder data is shared with service providers, or you have service providers that could affect the security of cardholder data, you must maintain and implement policies and procedures to manage those service providers. This is one of the few requirements that is always present in your validation scope, even (especially!) if you outsource *all* operations with card data but still maintain your own Merchant ID.

One way to determine who a service provider is in the context of this requirement is to ask one question about their activities. Could Rob's IT Shop affect the security of cardholder data? If the answer is yes, add them to the list of companies covered under Requirement 12.8. Once you have your list of companies, you must maintain the list you just created (Requirement 12.8.1), ensure you have written agreements whereby the service provider acknowledges they are responsible for cardholder data security (Requirement 12.8.2), any *new* service provider that is brought on board must go through a diligence process before engaging them (Requirement 12.8.3), make sure you have a program to monitor your service providers' compliance to PCI DSS (Requirement 12.8.4), and finally, maintain information about which requirements are managed by each provider (Requirement 12.8.5).

NOTE

Did you notice that PCI DSS version 3.0 mandates that service providers must acknowledge in writing they are compliant with PCI DSS? Requirement 12.8 is mostly a protection requirement for you. If you are a merchant and a breach of your data is caused by one of your service providers, you are liable. This requirement helps make both the merchant and service provider aware of the security requirements. If you are a service provider that uses a service provider (whew!), things may follow the same path.

Requirement 12.9 is a new requirement for service providers stating they must have these agreements in their contracts. It's now along the lines of ensuring those protections are present on both sides. Thus, the pressure is now on service providers who want to service compliant merchants to get their contracts in order.

Despite all protections, incidents will happen. In light of this, Requirement 12.10 requires that your organization “Implement an incident response plan. Be prepared to respond immediately to a system breach.” Expect your assessor to thoroughly review your incident response plan while validating your compliance on-site. Requirement 12.10.1 lists all the things that your incident response plan must include, most of which would be found in any good incident response policy. The only exception is the last bullet, which mandates that you include the card brands in your incident response process. You must test the plan annually (Requirement 12.10.2). You don’t have to hire someone to steal data to do this! A tabletop exercise should suffice, but only if you take it seriously. Remember, your breach will continue until it is 100% contained. A disorganized breach response only hurts you. You must have 24/7 incident response capabilities, and you must monitor critical systems at all times (Requirement 12.10.3). Chapter 8. reminds us that we must include any security event monitoring solution (such as wireless intrusion detection systems) in our incident response plan (also as Requirement 12.10.5). You must train your staff with incident response capabilities at least annually (Requirement 12.10.4). This could be a component of your tabletop exercise but should go beyond just the testing of your plan. Finally, you need procedures to update your plan based on things you learn and changes in the industry or threat landscape (Requirement 12.9.6).

The book Web site www.pcicompliancebook.info will contain sample policies and procedures, as well as links to resources that you can use to help jumpstart your Requirement 12 efforts.

APPENDIX A OF PCI DSS

The last set of requirements to discuss is a part of Appendix A. These requirements only apply to shared hosting providers. If you use a shared hosting provider for any process in your company dealing with cardholder data, be sure that hosting provider complies with Appendix A. Either your shared hosting provider should have this filled out by a QSA during their normal PCI DSS assessment or you may have your assessor go to your shared hosting provider to complete this portion of the assessment for you. If your QSA has to perform this, it most likely will add to the cost of your assessment from both a time and expenses basis.

For your hosting providers, here’s what you need to do. Appendix A consists of eight testing procedures: one major rollup requirement, and four subrequirements. The main focus of Appendix A is keeping every company’s environment separate by way of access controls. Requirement A.1.1 mandates that every merchant has their own ID, no shared IDs are permitted, and any Common Gateway Interface (CGI) scripts are run under the lower privilege merchant account, and not a shared account or as the same user that the Web server runs under.

Requirement A.1.2 further expands upon A.1.1 and has five unique testing procedures. Expect your assessor to verify: applications do not run under an elevated

privilege account like *root* or *Administrator* (A.1.2.a), each company's user ID has only permissions to its own files—none of which are group readable/writable—and suggest that these files exist in a *chroot* type jail (A.1.2.b), the company's user ID cannot overwrite system binaries (A.1.2.c), the company's user ID cannot read log files except those generated by its own applications (A.1.2.d), and that each hosting environment cannot take over the rest of the machines' disk space, bandwidth, memory, or CPU (A.1.2.e).

Requirement A.1.3 mandates that assessment trails or logs are enabled and consistent with Requirement 10. Finally, Requirement A.1.4 mandates that the hosting company's policies require timely forensic investigations in the event of a suspected compromise.

Your hosting provider should already have done these for you, but if they are not PCI DSS compliant and cannot supply this information for you, your QSA will need to validate it.

HOW TO BECOME COMPLIANT AND SECURE

Now that we've looked at the particulars of the PCI requirements for protecting cardholder data and discussed some of the technologies and methods available to achieve compliance, let's take a step back and briefly discuss your approach.

In many cases, organizations involved in handling PCI data existed and were involved with it before PCI DSS came out. So, networks and architecture processes already existed. If you were designing your network and your plan from the ground up with PCI DSS in mind, you'd likely do it differently. For example, you would probably minimize the use and transmission of card data on your network, and try to avoid storage altogether. Attempting to apply specific security standards after the fact is a different and difficult proposition.

By thinking logically through the requirement, both the letter and spirit, your business processes, and your IT environment, you can avoid a haphazard approach that can lead to problems such as inefficiency, unnecessary cost, insufficient controls, as well as unidentified risks or controls that are more restrictive than necessary.

We'd like to propose a process to satisfy these requirements.

STEP 1: IDENTIFY BUSINESS PROCESSES WITH CARD DATA

The first step is to simply create a list of all business processes that involve payment cards, electronic payments, card data, and others. Look at how those processes are implemented and what possible card data exposures there are. While doing it, start analyzing things such as follows:

- Is card data being stored? Where? Why?
- What is my card data flow?
- Can the process be improved by reducing card exposure?
- Is there any prohibited data—CVV2, PIN, and others—storage?

STEP 2: FOCUS ON SHRINKING THE SCOPE

After creating the list of processes that involve cards and card data, even if you outsource, focus on improving by reducing card exposure (see Chapter 4 for more tips). This step needs to happen before any talk about encryption, truncation, or any other data masking begins. Also, at this stage, focus on eliminating the storage of prohibited data as this will block any chances at PCI DSS compliance.

STEP 3: IDENTIFY WHERE THE DATA IS STORED

Databases might house cardholder data, but where else might it be found? Flat files that are results of batch processing, log files, backup tapes, and storage networks may all house sensitive information.

Ask the following questions:

- Where is the data located? Should it be?
- What format is it in (e.g., database, flat file)?
- What is the size of the data?

Answers to these questions will determine whether you have to make changes in your architecture to minimize the cost and work to protect the data. You will also probably need tools to validate your assumptions here. See Chapter 4 as well for tips.

STEP 4: DETERMINE WHAT TO DO ABOUT YOUR DATA

For each found location, consider what happens with it to bring it into compliance and reduce risk.

- Data is removed, and exposure to PCI DSS is eliminated.
- Data is truncated or hashed, thus reducing the risk of theft or loss.
- Data is encrypted using one of the discussed methods.

Only resort to encryption after you have tried other methods.

STEP 5: DETERMINE WHO NEEDS ACCESS

Too often, data breaches take place simply because people and applications have access to data they do not need. You have to balance the need for access with the proper control on that access to keep doing business.

Answer these questions:

- Who currently has access to the remaining repositories of sensitive data?
- Do they need access to do their job?
- What applications, such as backup applications or Web sites, need access?

Plan changes based on what is found.

STEP 6: DEVELOP AND DOCUMENT POLICIES

Now that you have identified what data you have, where your data is located, how it is protected, and who and what needs to access it, you can define, document, and

implement information-handling policies based on what, where, who, and how. This is where you establish such things as policies, standards, guidelines, and procedures. Such policies and procedures, as well as their implementation, are what a QSA would review to substantiate your claims of PCI DSS compliance.

COMMON MISTAKES AND PITFALLS

If you follow the media today, you might conclude that data encryption is everywhere. However, is this “good” encryption? Does it help you to protect the data from attackers while allowing for easy PCI compliance? A classic saying “Encryption is easy; key management is hard” illustrates one of the pitfalls that await those implementing encryption, and it is reflected in Requirement 3. While Requirement 3.4 simply states the need for encryption, the complexity of key management shows when you are reviewing Requirements 3.5 and 3.6 with their multiple subsections. Let’s look at some of the common mistakes that often occur when organizations try to use encryption to protect data at rest and data in transit, as well as achieve PCI compliance.

Before we start with the first mistake, how about the one numbered 0, thinking that PCI DSS data protection requirements are about encryption. No, they are about not letting the hackers get the data. As you’ve learned, *secure data deletion* is more reliable than *data encryption* as a mean of denying your opponents access to data. So, don’t encrypt what you can simply delete or not store.

The first mistake is not using encryption when it is both easy and mandatory. For example, why continue using those pesky plaintext protocols such as Telnet and FTP? One can argue that people should have abandoned the above protocols for a host of other reasons, not just PCI DSS, but as the Solaris Telnet 0-day vulnerability fiasco from a few years ago indicates, enough people are still using them. If you insist on using them, make sure that card data never travels over them.

Similarly, not using SSL encryption for online purchases using a payment card is not common, we still see such instances on lesser-known e-commerce Web sites. Exposing card data information to known, actively used attacks, such as sniffing, is inexcusable in the age when anybody can configure SSL encryption and certificates are cheap. While risk of sniffing is typically overshadowed by the risks to stored card data, there is indeed no excuse to not encrypting the data in transit when it is easy and does not cost any extra.

The second mistake has been mentioned by most cryptographers out there: inventing your own cryptographic algorithm. Cryptography is a science just like physics and mathematics; in fact, we all know that it is based on the latter. This is an area where amateurs have no place. Use publicly vetted algorithms such as AES or others with many secure implementations that can be used for free.

An interesting extension of this mistake has to do with failing to correctly implement a well-designed cryptographic algorithm. Indeed, algorithm design is hard, but quality implementation isn’t an easy job either. As a result, people who chose to

re-invent a “known good” crypto algorithm might be doing themselves a disservice (see WEP), if a proven implementation exists (as a cryptographic library, for example).

Every reader will probably recognize the third mistake: “hard-coding” secrets, such as passwords and keys. As we know, security of a quality cryptographic algorithm does not depend on its secrecy, but on its key or password. If you inadvertently make such passwords available for attackers, the game is over. Embedding passwords in code (binaries), configuration files, Web pages, style sheets, or other “hidden” files is just providing your secret to attackers. And, no, your XORing the password with a string of characters does not count, it just replaces your credible “secured by AES” label by a purely humorous “protected by the power of XOR.” You can only secure a password by encrypting it, and then your problem does not go away since you have to deal with a new password.

Hard-coded secrets led to many disasters in the recent history of information security. DVD and now HD-DVD encryption breaches are just the most famous of them. The extensions of such error, where passwords are “hidden” in files to enable scripting or automation, have helped attackers to extend their control over the compromised networks. One cannot argue that system administrators need to automate routine tasks, but “root” passwords in scripts and configuration files should be as archaic as double-digit years. Finally, the question of whether having passwords hardcoded in JavaScript code, visible on the website using “view source” command is left as an exercise to the reader.

As we learned in this chapter, database data encryption is becoming more common and more database vendors natively support it—their claim is that it is easy to just protect your database by encrypting it. Great! The task is done and the data is secured by a well-implemented encryption algorithm. Now, where do we put the encryption keys? Ah, why not in the same database, just another table? Thus, the fourth mistake is manifested in the form of storing keys with data. One author is aware of a few organizations who sought to protect their credit card databases by encrypting the tables with sensitive data and then storing the key in another table on the same database server.

Sometimes, one hears a claim that such “protection” works against fools and low-skill attackers—they would see a string of random binary data where a credit card number should have been and then go away. However, a more correct way of putting it is that it works as a “checkbox encryption” against your own management—they can now claim that cryptography protects their organization’s crown jewels, whereas in reality, it protects nothing and achieves no compliance benefits either.

While we’re at it, think about the following as well: while you might not be leaving the keys in an obvious place such as a database table, do you prevent key leakage into swap files, crash dumps, logs, and other areas that might be seen by attackers? This is much more insidious, and a detailed discussion goes beyond the scope of this book.

Finally, the fifth mistake turns encryption again into the very entity that is supposed to benefit from it (i.e., your organization): not handling data recovery. Ask

yourself if your crypto implementation passes the “lottery test.” If whoever knows the keys to data wins the lottery and disappears forever, will you be able to get your data back? PCI DSS does not mandate that you create procedures to ensure the availability of decryption keys, but it does not mean that you can avoid it.

If you implemented cryptography correctly such that there is no way to bypass the security it provides, and at the same time, you didn’t think about data recovery, your implementation will likely not pass the lottery test. As a result, the data would be as good as gone in case of a key loss. Did we mention that cryptography is a science? Handling key revocation and data recovery is a critical piece of the security puzzle. For example, Windows EFS has support for such features. Thus, protecting data from theft is only half of the challenge—you need to protect the data from loss due to “good” crypto.

To conclude, data protection is a critical piece of information security, and encryption plays a major role in it. However, one should try to avoid the mentioned pitfalls and should consider data encryption to be that long-sought security “silver bullet.” Finally, don’t encrypt what you can destroy!

CASE STUDY

The case study below illustrates how removal of card data from the environment, as well as various security measures, help achieve compliance and security.

THE CASE OF THE LEAKY DATA

Kristin’s Komix, an antique cartoon and paraphernalia store, started selling stuff online to fans back in 1997. In that ancient age of e-commerce, when the terms *new economy*, *dot-coms*, and *e-commerce* were new and cool, most Web site owners who wished to sell something from their sites had to find shopping cart software and install it on their sites. Often, especially for free and open-source shopping cart tools, they had to get into the code and change a few things here and there to customize and adapt the tool for their sites.

This is exactly what Kristin did: she grabbed the Bob and Garry Epic Fail of a Shopping Cart, 4.1 software package from www.bobngarryepicfailcart.com and dropped it on her site. After the package was installed, she checked whether the visitors of her site can click on the links and buttons, she went into the Perl code of the shopping cart to change variables, and finally configured the card processing capability.

While this scenario is unthinkable in the enlightened Internet age of 2012, remember that this was 1997!

After all required configuration “magic” was done, she advertised to her customers and to a local community that she was the first to do e-commerce in the entire industry.

For a few years, Kristin was happily selling old cartoons and charging her customer’s cards, totally oblivious to the threats and the emergence of PCI DSS in later

years. However, as she discovered later, someone else was charging some of those cards as well. Unbeknownst to Kristin's IT team, her shopping cart was deployed in debug mode and actually logged all the cards into a big text file, located in the same directory, and anybody browsing to www.kristinskomix.com/epicfailcart/debug_card_log.txt will be able to download an ever-growing card log that contained the following:

```
date, time, name, PAN, exp, amount.
```

In other words, most of the details on the cardholders, with the exception of their address, which was not commonly needed for the transactions in the 1990s.

Unfortunately for Kristin, this file location was well-known to more than a few malicious hackers, and the cards were pilfered and used—in moderation, of course, to avoid the suspicions—for various purchases worldwide.

When Kristin came to grips that her data was being used for fraud as well as the reality of PCI, what was the first step she took?

Did she reach out for those encryption tools? Data masking? An advanced cart software?

No, she picked the right choice: she went with PayPal Checkout (see <https://merchant.paypal.com> for more information) for all payment processing. No more cards—far less risk!

THE CASE OF THE SATELLITE LOCATION

Barbie's Booze, a specialty spirits shop in downtown Chicago was running out of room. Barbie got a fantastic deal on rent for her store and regularly had a packed location whereby she moved impressive amounts of inventory to local residents. She was beginning to need some support staff to handle all of the back office administrative requirements, and she couldn't justify removing floor space to create workstations for her staff. She decided she would open a satellite location roughly 2 miles away in a space purpose built for offices.

She knew she would need to connect the offices so that her electronic systems would be able to function over the distance. She found a deal on a used Checkpoint firewall set and promptly had an IT person set up a VPN for her sites to use. Unfortunately, her IT person was not well versed in this version of Checkpoint, and the VPN that was set up between the two sites did not have any encryption deployed over the link (yes, it is possible to do this). As information was being transferred from site to site, all of her daily accounting information that included credit card numbers was exposed to a public network.

Barbie hired a QSA to do a quick review for PCI Compliance as her acquirer informed her she would soon become a Level 2 merchant. She researched and found a good QSA that really understood how her business works as well as had a background in systems and network administration. He quickly informed Barbie of the

issue and gave her tips to encrypt the link between the two locations. She contacted her bank and informed them of the issue. An investigation ensued, and she dodged a bullet. Her service provider had coincidentally connected both of her lines from a single Central Office (CO), and the telco was able to walk the bank through the security controls contained inside that CO. The bank felt satisfied that no breach occurred during the 30 days where the link was used without encryption, and Barbie dodged a bullet.

Barbie learned that when she was deploying technology to support her business, she needed to ensure that the people installing and configuring the technology were in fact experts, and not just generalists that could make the product work, but not work securely.

SUMMARY

PCI DSS data protection requirements are among the most challenging parts of PCI as they deal with technology subjects such as encryption algorithms. Many organizations are still not compliant, and risk fines and data breaches as a consequence.

Here is a deceptively simple answer to your encryption worries: don't do it! If only you eliminate data storage, you eliminate the need to protect data at rest, which takes care of a massive amount of complexity in Requirement 3.

Similarly, if you eliminate the movement of card data over insecure networks, Requirement 4 will become simpler. With the proper preparation and execution of your plan, you can protect the information you have been entrusted with.

REFERENCES

- [1] Visa, Inc. DropTheData website. <www.visa.com/dropthedata>; 2009 [accessed 12.07.09].
- [2] Transparent Data Encryption. <www.oracle.com/technology/oramag/oracle/05-sep/o55security.html>; 2009 [accessed 30.07.09].
- [3] Encrypting Data Values in DB2 Universal Database. <www.ibm.com/developerworks/data/library/techarticle/benfield/0108benfield.html>; 2009 [accessed 30.07.09].

Using wireless networking

8

INFORMATION IN THIS CHAPTER:

- What is wireless network security?
- Where is wireless network security in PCI DSS?
- Why do we need wireless network security?
- Tools and best practices
- Common mistakes and pitfalls
- Case study

Wireless technologies continue to advance in both speed and proliferation. It seems like every spring season we see merchants get frisky and revisit the prospects of pushing wireless technologies down to their stores. Most big box retailers already use Wi-Fi® for inventory management and have dealt with upgrading portions of their outdated technology to comply with Payment Card Industry Data Security Standard (PCI DSS). Smaller retailers have toyed with implementing the technology to cut costs on store build-out or even to add additional registers in parking lots or other outdoor areas. To add to the capabilities of radio frequency (RF) communication, you could even build a point-of-sale (POS) network that functions over Bluetooth® or Zigbee®. We shudder to think about those kinds of networks, but it is possible and more likely to be used as the technologies mature. Wireless networks don't stop at Wi-Fi or Bluetooth, however.

Recent advancements in cellular data networks created an opportunity for a new class of card processing terminals that can process cards without Wi-Fi or hard-wired Internet connections. As we saw the effects of convergence in our cellular telephones over the last decade, we are beginning to see the same effects in the payment terminal market. Not only are the terminals becoming smaller and more functional, but some companies have gone completely paperless with their field technicians. Many companies are now giving both purpose built devices and tablet computers to their field technicians to prioritize work lists, give directions on where to go, provide traffic updates to keep their schedules efficient, keep track of spare part inventory while sending that information back to headquarters, and finally accepting a credit card for payment either via a sled or via a built-in reader.

Of course, terminals don't need to be that sophisticated to be in-scope! Think about the last time you went to an outdoor festival or arts and crafts fair. If you ran out of cash, you had two options. You could either run over to that ATM machine on the street

that only seemed to need a long orange extension cord to operate or pay for your wares directly with the merchant through a credit card terminal. Depending on how hot the sun was or how many adult beverages you may have consumed, you probably didn't think too much of it. Both of those options look and feel like the ones you see indoors but with one major difference. They are not connected to any sort of wired network!

The intent of PCI DSS with respect to wireless technologies is to impose a subset of the requirements on any communication between two devices that does not occur over a wired network.

This chapter covers some of the basics in wireless payment processing as well as the pitfalls for which you need to be aware. Although not foolproof, the basic concepts here keep your assessors and acquirers happy as well as placate the business development and security professionals at your company.

WHAT IS WIRELESS NETWORK SECURITY?

When one of the authors thinks about his first wireless network, he remembers that his access point (AP) was the only one within range of his laptop's Wi-Fi card. You know, the big fat credit card-sized ones that plugged into that fancy PCMCIA slot on your laptop? Or even better, that brick you carried around that plugged into the USB port?

Technology has come a long way in the last 15 years.

Not long after his first foray into Wi-Fi, he noticed another wireless AP suddenly appearing as an available network to join. He protected his Wi-Fi network with a 64-bit Wired Equivalent Privacy (WEP) key (128-bit keys were not available on the hardware at the time) just to see how it worked, but he noticed that his new "target network" did not. Curious, he joined the other network.

Although the signal strength was not fantastic, he was still able to browse the Internet at a slow pace, no doubt riding the same cable modem service that was coming into his own house. He decided to probe a little bit further, and sure enough, an open file share was available on one of the machines connected to the network!

Curiosity aside, the author's neighbor clearly believed that he was either the first person to install wireless in his area or that his signal did not extend beyond his four walls. The latter is the most common misconception carried by individuals, even though they can receive a cellular telephone, television, and radio signals inside the very same four walls that they expect will block Wi-Fi.

Now imagine that same individual several years later having a brilliant brain-storm that includes deploying Wi-Fi into his store location, so he can sit out among the patrons with his laptop but still get his company business done. You can probably see where this is going. He made the fatal mistake of assuming that nobody would want to tinker with his little store. Not long after putting up his cheap AP that he purchased from a local electronics shop, he received a phone call from his acquiring bank notifying him that he may have a problem.

Wireless networks, and specifically Wi-Fi networks, are frequently the target of both nuisance and sophisticated attacks. Wi-Fi networks in particular are attractive

to attackers because the cost to acquire the equipment used in the attack is minimal. Cellular network hacking has continued to take a massive leap forward largely due to hacked basebands and femtocells (those little cellular extension boxes you plug into your home network when you can't get good cellular coverage in your home). What used to cost into six figures to acquire now can be had for well under \$200 with some creative hacking and know-how.

It just goes to show that you should never trust the actual transport mechanism of your communications.

Technology has come a long way and the chips our technology uses are quite capable of encrypting data before dropping it to a network connection with minimal impact to performance. Satellite networks still have an advantage over Wi-Fi and cellular networks, however, as the equipment required to go after satellite networks is typically expensive and requires specific training or knowledge to carry out a successful attack. As we saw with cellular networks, satellite networks will too fall victim to hacking (and in some cases already have) and should not be trusted either. These networks should never ignore things like good security and encryption, but often, the cost to acquire the equipment necessary to hack those networks are seen as security controls (financial constraints).

To clear the air, they are not.

Early implementations of Wi-Fi only offered WEP encryption. Although the underlying algorithm was solid (RC4), the implementation of the algorithm caused key information to be leaked with each packet. When used in a certain way, this information leads to the compromise of the WEP key (see the "Common Mistakes and Pitfalls" section of this chapter). With the key in hand, attackers could decode every single packet over the air, dumping information in a manner not too unlike sniffing traffic via a network span port on a switch. Usernames, passwords, company secrets, and customer information were now all available until the key was changed. Because the keys could be compromised in a few minutes, key changes did not stop the attacks.

But don't totally feel comfortable with your WPA and WPA2 setups either. If you so choose, you can take some packets from a capture and upload them into a number of cloud services that will crack the password for you for a fee (use your favorite search engine to find them). The point is, you should never trust the link layer encryption when packets are moving through the air.

Basic security functionality like disabling Service Set IDentifier (SSID) broadcast and Media Access Control (MAC) address filtering adds to the illusion of security. These features are easily overcome by anyone with a basic understanding of wireless networking and the proper tools. Remember, wireless encryption only protects the payload, but it does not encapsulate the entire packet from a laptop to the AP. This means that both the SSID and MAC addresses can be seen by a casual observer regardless of the encryption technology deployed, and both of these values can be configured on any Wi-Fi card to perform a successful attack.

Worse yet, now that the casual observer has examples of legitimate hosts (and their hardware addressing) as well as the key, he could join the network and poke around looking like a legitimate device on the network. He could easily map the defenses (if any) deployed by watching for a firewall to stop his probes. For the most part, these dropped packets are not logged, or if logged are never analyzed, so he

could poke around undetected for as long as he wanted. Store and internal networks in general used to be completely devoid of firewalls, but key IT investments for PCI compliance has changed that dramatically over the last few years. It's not anywhere close to the level it needs to be, but attackers are more likely to encounter a firewall today than they were 5 years ago. Thus, an attacker joining a remote store's Wi-Fi network generally gave him free rein of the corporate network and a launch pad for attacks on servers inside the "secured" area of the network. Depending on how the Wi-Fi technology is configured, this may not be any different today. Many cardholder data compromises start like this.

WHERE IS WIRELESS NETWORK SECURITY IN PCI DSS?

For the most part, PCI DSS only sets the stage for a baseline of wireless security. PCI DSS's handling of wireless network security is a prime example of how PCI DSS compliance does not necessarily mean you are secure.

Little has changed upfront in PCI DSS 3.0, but there is a small section on wireless that is helpful to review. Notice this revision still includes the acronym WLAN (Wireless LAN) in reference to wireless networks. It's pretty complete as far as giving you a definition of what is in-scope for PCI DSS. When in doubt, assume it is in-scope.

WARNING

Manufacturers of cellular or satellite products tell you they are safer to use because of the difficulty in intercepting the traffic, just as the ones that manufacture and sell Frequency-Hopping Spread Spectrum radios will. Using more obscure spectrum like General Mobile Radio Service (GMRS), which may already be deployed in larger locations to facilitate communication among the staff doesn't add any security to your payments either. Security by obscurity is a foolish way to protect yourself against the bad guys. Eventually, someone will (or in many cases already has) figure out a cheap way to intercept communications like this and the game will be over. The technology may lend itself to a lower risk of compromise today, but that doesn't mean it will be that way forever. Be sure you are using industry standard stream-ciphers over these networks. And remember that any wireless technology in use must comply with these requirements!

Companies wishing to comply with PCI DSS must minimally address several requirements, even if wireless is not deployed in the target environment. Those are 1.1.2, 1.2.3, 2.1.1, 4.1, 4.1.1, 9.1.3, 10.5.4, 11.1, 11.1.1, 11.1.2, 12.3, 12.10.3, and 12.10.5. Let's explore how companies can meet those requirements.

NOTE

If you think that you have no legitimate wireless in your production environment, at a minimum you still must address the actions mandated by Requirements 11.1, 12.10.3, and 12.10.5. "Rogue," or unauthorized, APs in your cardholder environment can be lurking without your knowledge.

REQUIREMENTS 1, 11, AND 12: DOCUMENTATION

The first step, as is with most parts of PCI, is to document! 1.1.2 was referenced in Chapter 5, but it has applicability here specifically for wireless networks. Any wireless networks that are permissible in your environment should be documented in the network diagram you present to assessors. As an example, let's say that the only wireless permitted in your environment is a vendor wireless network that only has limited Internet access. Your network should already have a firewall between the unsecured wireless network and the corporate network. That action alone helps you meet most of PCI DSS. Even though it is not connected to any card processing networks, nor does it process cards itself, placing it on the diagram helps to illustrate that you have your ducks in a row. You can also expect that your assessors will cross reference with 11.1 and 11.1.1 for authorized devices.

NOTE

Good Qualified Security Assessors (QSAs) are trained to look for inconsistencies and oversights. If a QSA asks for a network diagram and you provide one that doesn't include Wi-Fi, yet the QSA is connecting to their corporate network using a Wi-Fi network from your company, they will start to ask themselves, "What else are they not telling me?"

Requirement 1.2.3 mandates that firewalls be installed between any in-scope networks and the wireless network. This is pretty self-explanatory, but the part that can trip companies up is defining what is acceptable as a firewall. Because PCI DSS uses the word "perimeter" to describe the kind of firewalls you should use, many QSAs interpret that to mean a stateful inspection firewall like what you see in Requirement 1.3.6. Some QSAs might consider stateless packet filtering firewalls as a way to meet this requirement. With the flexibility that Reflexive Access Lists afford you as a stateful inspection access list, consider deploying those instead of new firewall hardware. Pay attention to your resource management consoles as any time you add filtering like this to your switches or routers you will be adding both overhead and memory usage. Routers and switches at or near capacity should be upgraded before considering this type of deployment. The main point here is to put some kind of enforcement point between the wireless and wired network—preferably a stand-alone firewall—with the wireless network being on the untrusted side of the device.

With the new penetration testing requirements, you can expect that your QSA will want to see the report that shows how someone tested this firewall as well. It's part of requirement 11.3.4 in PCI DSS 3.0.

Speaking of Requirement 11, 11.1.1 has been updated to require documentation on the list of authorized APs with their business justification. Your QSA should ask to see a few of the APs around the office for verification that this is happening.

Zooming to Requirement 12, we have more documentation-related items to address. Requirement 12.3 now includes wireless technologies. Remember, this is a policy document. If you set a company policy, your internal audit group should

conduct periodic reviews to ensure that the policies are being followed. Your QSA is not required to dig that deep, but if a corporate policy that has not been followed leads to a breach, you probably won't receive safe harbor protection under the various card brand operating rules or applicable state or federal laws. The policies for Requirement 12.3 should address all the following before deploying wireless in your environment:

- Explicit management approval for the use of wireless,
- An authentication scheme for anyone or anything that wants to use the technology,
- A method to accurately and readily determine owner, contact information, and purpose as with a label,
- Acceptable uses,
- Acceptable network locations for wireless,
- List of company-approved products,
- Automatic disconnect of sessions after a specific period of inactivity (think more client-to-site virtual private network [VPN] access, less standard Wi-Fi)—consider 30 min of inactivity a good benchmark with 24-h of maximum connection time before a forced disconnect,
- Activation of remote-access technologies for vendors only when needed by vendors, with immediate deactivation after use,
- Prohibition of copying, moving, or storing of cardholder data onto local electronic media when accessing such data via remote-access technologies.

Most companies that deploy this type of technology can address the wireless components as part of their broader policy covering Requirement 12.3.

Next in Requirement 12 is 12.10.3, which requires that your incident response personnel are available 24/7 for both incident response and monitoring looking for any evidence of unauthorized activity, unauthorized wireless APs, and able to manage critical intrusion detection system (IDS) alerts or reports of unauthorized file changes. If you follow PCI DSS to the letter, you might be thinking that the possibility of activating this clause in your incident response plan seems fairly remote. Here's a hint: it is. We'll get into that more later in the "Testing for Unauthorized Wireless: Requirement 11.1" section of this chapter.

Finally, Requirement 12.10.5 needs some documentation to validate. Essentially, all of the monitoring that is supposed to happen with these devices should be centralized and processed. Your QSA should ask to see how you monitor and respond to alerts from these devices.

ACTUAL SECURITY OF WIRELESS DEVICES: REQUIREMENTS 2, 4, AND 9

By now, you are probably wondering when we will get to those fancy encryption requirements! It goes without saying, but building a secure environment where you operate any sort of technology starts with good and complete documentation. Part of

your wireless usage and deployment standards should include select elements from Requirements 2, 4, and 9.

Wireless encryption technologies have come a long way in the last several years. Just nearly two decades ago, the only options for wireless encryption were WEP or tunneling encryption inside your wireless connection such as a VPN or Secure Socket Layer (SSL) connection. Now, there are a multitude of options for both encryption and authentication. As of PCI DSS 2.0, WEP is no longer permitted as an acceptable protection technology. That doesn't mean, however, that you might use WEP in combination with SSL or a VPN to provide additional technology. Keep in mind, this is going to be a hard sell to both your QSA and your acquiring bank. You are most likely better off updating your systems or changing your business process to avoid WEP altogether.

Requirement 2.1.1 lists five items (a–e) that QSAs must check for to validate compliance. Your wireless installation should (at a minimum):

- Have unique encryption keys (i.e., not default) that are changed anytime anyone with knowledge of the keys leaves the company or changes positions (for shared keys only);
- Change default Simple Network Management Protocol (SNMP) community strings;
- Change default passwords/passphrases used for administration on APs;
- Support (and deploy) strong encryption for authentication and transmission over wireless networks—for example, Wi-Fi Protected Access (WPA) or WPA2 (802.11i); and
- Change other security-related wireless vendor defaults.

This dovetails nicely into Requirement 4.1.1 that simply reiterates that industry best practices should be used for encryption and authentication of wireless devices. This requirement overlaps with Requirement 2.1.1 somewhat, but you can consider 2.1.1 as a configuration and design requirement while 4.1.1 is an operational requirement. This may help you determine which teams are responsible for which elements.

NOTE

WEP may no longer be used for new in-scope networks. Per the last revision of PCI DSS (version 2.0), Requirement 4.1.1, alternative schemes like WPA (WPA and WPA2/802.11i) must be used on these networks instead.

Some of the attacks against wireless networks start by gathering lots of traffic, either by performing injection attacks or by selectively targeting users and having them download large attachments or stream media. The more traffic you have, the more cryptanalysis you can perform, and the more likely the attack against the key will be successful. On top of that, shared keys are just that—*shared*. Everything you learn about information security screams “Don’t do that!”

WPA2 or 802.11i provides networks with a significant boost in security by authenticating individual users through certificates or usernames and passwords. Additionally, devices that use WPA2 or 802.11i benefit from mutual authentication, meaning that the device itself can authenticate the AP it uses making evil-twin type attacks much more difficult to perform.

What constitutes an industry best practice for wireless security? For Wi-Fi installations, WPA or WPA2 should be deployed. WPA-TKIP is increasingly coming under attack due to its reliance on WEP to function. WPA was originally designed as an interim fix to WEP until the 802.11i standard was finalized (also known as WPA2) and has recently demonstrated vulnerability to dictionary and “chopchop” like attacks due to its reliance on WEP. WPA-AES is a different mechanism, and does not fall victim to those specific types of attacks. Details of these attacks can readily be found via your favorite search engine. New installations using Wi-Fi should absolutely use WPA2 with some form of unique authentication, sometimes called WPA2-Enterprise. Don’t use shared keys (sometimes called WPA2-Personal). They are a pain to deal with and, for large installations, virtually impossible to maintain according to PCI DSS.

Given these constraints, your security is only as strong as the users that use it daily. Basic passwords may be easily cracked through cloud cracking services with only a few packets captured. Consider other methods of authentication and key derivation, and consider additive security measures to protect information going through the air.

For other wireless technologies such as satellite, cellular, or microwave, encrypt transmissions with a current stream cipher such as the Advanced Encryption Standard (AES), or Elliptical Curve Cryptography (ECC), or an industry-accepted algorithm of equivalent or better strength. New algorithms are coming out all the time, so you have plenty to choose from. Don’t rely on the cost of communication interception equipment to secure these increasingly popular forms of communication. Such reliance is both risky and could lead to a false sense of security, further putting your company at risk.

Requirement 9.1.3 mandates physical protection for wireless devices, as well networking and communications hardware and telecommunication lines. APs should be kept under lock and key, behind badged access doors, or in some cases, it should be protected with a cage. The intent of this requirement is to prevent an unauthorized user from tampering with the device. Don’t rely on a 12-feet ceiling to protect the APs deployed on or above it. Ladders are readily available here in the 21st century. For that reason, don’t rely on other physical hiding techniques, such as disguising your AP as a smoke detector, to secure your hardware.

LOGGING AND WIRELESS NETWORKS: REQUIREMENT 10.5.4

Wireless gets a quick mention in the dreaded logging requirements for PCI DSS. Be sure to include wireless logs from your AP in your centralized logging solution. Different vendors have different ways of communicating logging data but most can dump data via syslog(). Piggybacking on the same infrastructure that collects logs from routers and switches is trivial to do.

TESTING FOR UNAUTHORIZED WIRELESS: REQUIREMENT 11.1

When it comes to wireless, there is no requirement more debated than 11.1. Security and compliance may not be farther apart anywhere else in the standard than they are

right here. On one side, merchants are equipping district managers with basic wireless tools and making sure they hit each of their stores at least once a quarter. These merchants rarely are able to be compliant with the standard all year long as invariably stores are missed and equipment fails. Managers don't understand why they have to do it, and every merchant has at least one maverick out there that would opt to buzz the tower instead of respecting the controller's wishes.

NOTE

The intent of Requirement 11.1 is to discover unauthorized wireless devices. Unauthorized devices can show up in your environment even with a "No Wireless" policy. As the "Property of IT" example in the "Why Do We Need Wireless Network Security?" section of this chapter illustrates, breaches can easily come from the wireless device you don't know is there.

On the other end of the spectrum, you have wireless defense vendors who tell their prospects that they cannot comply with PCI DSS unless they buy and deploy their wireless IDS or intrusion prevention system (IPS) solution. One author knows he has ruined a few sales people's quarters by giving merchants alternatives to deploying wireless IPS. Early deployments were often costly, and retailers of any substantial size face mounting costs in deploying and operating the technology in each store. A \$2,000 cash outlay for one location is easy to swallow, but that same outlay for a thousand locations is significant. Add to that the ongoing operational costs of maintaining the infrastructure, and things start to get out of hand.

In extremely rare cases, some merchants have sophisticated enough network equipment to positively identify every device plugged into their network with automatic quarantine capabilities when devices that should not be active are plugged in. The number of ways you can attack this particular requirement are numerous, and the effective security of these solutions varies greatly.

The authors would like all those wireless IDS and IPS vendors to cover their ears for the duration of this paragraph. Just skip the rest of this paragraph, and go to the next one. Neither author wants to see this show up in a marketing slick, seriously. For the rest of you, the wireless vendors really do have your best interests in mind when they are pushing their products as a method to meet this requirement. One vendor in particular has a great analogy about scanning each store once per quarter (as the requirement states). It's equivalent of turning your firewall on for a few hours on 1 day each quarter, then assuming nobody would want to come in and attack you until you turn it on for that 1 day next quarter. This analogy is fitting because it helps put things into context. It's also a great illustration on the difference between compliance and security.

With PCI DSS 3.0, the Council has made the language here a bit more generic, which offers you more options on how you wish to comply. While compliance with Requirement 11.1 could mean that at a minimum, you must scan each location with a wireless analyzer each quarter to identify all wireless devices (11.1.a),

they have changed the language to basically make it your decision on the how of the detection. Should an unauthorized one show up, it should be traced down and efforts made to ensure that it is not affecting the security of the cardholder network. Alternatively, you can use a Wireless IDS or IPS to identify these devices in real-time, and in some cases, take action against them to prevent them from functioning on the network. Here is a quick breakdown of the subrequirements:

- You must have a documented process to detect and identify wireless APs on a quarterly basis.
- Ensure the methodology used above is adequate to detect and identify any unauthorized devices, including at least the following:
 - WLAN cards inserted into system components,
 - Portable or mobile devices attached to system components to create an AP, and
 - Wireless devices attached to a network port or device.
- The documented process to identify both authorized (this is new for 3.0) and unauthorized wireless devices is performed at least quarterly.
- Configuration for automated technologies like WIDS and WIPS must generate alerts (and check to see if they are followed up on).
- Your incident response plan includes a response in the event unauthorized wireless devices are detected (overlap with 12.10.3).

If you are using a Wi-Fi POS system for your stores, do yourself a favor and deploy an AP that has IDS and IPS functionality out of the box. Then, enable it and ensure that it meets Requirement 11.1.b. If your plans include Wi-Fi POS, you should do everything you can to defend those devices. Make no mistake; if you deploy it, the attackers will come.

Quarterly sweeps or wireless IDS/IPS: how to choose

As with most parts of complying with PCI DSS, there is no clear solution or silver bullet. Let's explore where one might be better than the other.

Automated solutions are slick. They provide scalability (usually) and do much of the thinking for us. If you have Wi-Fi technology in your locations, using a Wireless IDS or IPS solution is probably going to be the best way to handle security and compliance with PCI DSS. If you have a proven rapid response time in the field, a Wireless IDS may work well for you. The difference is similar to network IDS and IPS technologies we discussed in Chapter 5. Wireless IDS will only tell you about the problem, and then, you must take action. In order to make IDS an effective solution, you must have some kind of 24/7 response and appropriate staffing.

Wireless IPS solutions typically come on the same hardware and carry incrementally insignificant costs over the wireless IDS solutions. If you don't have a proven response time or don't want to staff up accordingly, go with the IPS solution instead. Let it alert, but also let it take action. You will spend more time upfront configuring it to not interrupt normal business activities, but overall IPS can carry a lower cost to your organization (when properly tuned and maintained).

So with all this fancy, whiz-bang technology, why would we go the manual route? For a couple of reasons, but the number one reason typically being cost. Companies considering this option will only have limited network capabilities in their locations, such as a store front. If all of the network connectivity comes in through a Digital Subscriber Line on a modem with four ports on it, and all four ports are in use by equipment required to run the business, regular visual inspection of this equipment (shift change is a good time to do this) may be sufficient enough to protect the enterprise and only rely on the quarterly sweep to identify any rogue devices. It's not foolproof (as any number of security pundits could no doubt come up with a list of ways to defeat this control), but based on the risk, it could definitely be both an acceptable compliance and security control.

This is where training is key. Good security includes all the principals of technology, process, and people. The impact that skimmers have on credit card fraud committed against fixed devices (such as unattended fuel pumps or cash machines) will be dramatically reduced by a thorough visual inspection at a shift change and other random times during the day. Leveraging technologies like SpotSkim to assist in this will not only make this easier, but keep all your documentation for when PCI Requirement 9.9 is no longer a best practice. This is another argument for keeping your networking simple. A shift manager or even an individual contributor taking a few minutes each day to visually inspect all equipment and network jacks can be an effective control against unauthorized wireless devices.

WHY DO WE NEED WIRELESS NETWORK SECURITY?

Corporate networks are protected by many layers of security, one of which being physical security. Think about how difficult it is to get into the data center at your company. It probably includes going through multiple layers of physical security controls such as parking access gates, fences, and security guards. Employees can easily get access to the facility, but getting access to the data center is usually limited to a select group of individuals.

Wireless networking cannot rely on physical security to completely secure it. Yes, it is possible to use directional antennas to contain the signal inside your four walls or even use specially designed mesh surfaces inside your walls to create a Faraday cage for Wi-Fi signals, but that is neither foolproof nor 100% secure. Worse, those techniques generally don't work with all wireless technologies, and it won't protect your network against a hot-shot user who puts a generic AP in his office, so he can work on his laptop from the conference room.

Because we lack physical security controls, we must rely almost entirely on technical controls to protect our wireless networks. Defense technologies have come a long way since the first corporate AP was deployed, but companies still need to install and configure these technologies properly in order for them to be effective.

One popular story that is part of QSA lore is a company that was in the middle of a PCI assessment and the subject of wireless technologies came up. The company's

representatives were adamant that wireless technologies were expressly prohibited by policy, yet the observant QSA looked over to the head of the conference table and saw a suspicious Linksys device (with antennas) that had a sticker that said “Property of IT. DO NOT TOUCH” placed in plain sight across the front. When the QSA asked about that device, the company’s representative just shrugged her shoulders and said, “That showed up a few months ago. Our relationship with IT is not great, so we just ignored the device and didn’t touch it.” Not only did IT not deploy the device, but someone had been accessing the corporate network from a neighboring parking garage for months.

PCI DSS only requires a minimum baseline for wireless security. In the authors’ opinion, companies relying on wireless technology for their business should go beyond PCI DSS and choose the appropriate defensive solution to protect their networks.

OTHER WIRELESS TECHNOLOGIES

We’ve spent the majority of this chapter going over Wi-Fi and cellular network communications, but there are a multitude of other common network technologies that you need to watch out for and secure. The easiest way to think about this problem is to consider how a device gets and uses its data. If you don’t see a network or USB cable coming out of the device, it is using some kind of wireless technology and you need to secure it. Here is a list of some of those technologies:

- Bluetooth is included in almost every smartphone and every new laptop that sits on a retail shelf. It was originally designed in 1994 by two individuals working for Ericsson in Sweden. As of this writing, the current specification is now up to version 4.1. Version 3 included an interesting possibility to facilitate high-speed Bluetooth networks by coupling them with 802.11 (or Wi-Fi) networks for the large transfers. Be sure to fully understand your deployment and its location in your network. Bluetooth base stations should be secured in the same manner that you would secure a Wi-Fi station.
- Zigbee is a new wireless standard more formally known as 802.15.4. While there is a retail specification (still) in development right now, there is no plan to include payment processing as part of that specification. Ensure that if you are using this technology that you have separated this from any payment systems. The security of these implementations is largely unknown at this point in the retail space. You will see Zigbee primarily used in automation and utility management (namely electricity).
- GMRS as described above could be used as a local data service for companies looking to build an infrastructure outside of a phone company for a contained geographical area. This should be treated the same way Wi-Fi would be, and all communication should be encrypted independently of any link-layer encryption you might use.
- Worldwide Interoperability for Microwave Access (WiMAX) may also be a technology used in your city that could include data services for various devices that might process payments. Ensure that you have taken the appropriate steps to secure both your communications and your devices.

TOOLS AND BEST PRACTICES

Wireless technologies have permeated virtually every part of our technologically advanced society from the use of cellular technology and smart phones to enter credit cards and process payments to the casual Wi-Fi device lurking at a sleepy cafe. There are numerous tools you can use to both detect networks and defend against potential hackers.

Beginning with detecting, there are both commercial and open source solutions. Commercial solutions are readily available through your favorite search engine, and most of the major AP manufacturers have similar capabilities built into their devices. The solution that fits best for you may just come down to your specific requirements and budget.

As far as open source tools, three in particular lead the pack by far. NetStumbler (www.netstumbler.com) for Windows was one of the first tools with a graphical interface that was easy for any casual wireless junkie to use. It did have limitations (and in some cases still does) but is a decent tool for beginners.

NOTE

Although wireless scanning software has come a long way over the years, especially free ones like NetStumbler, users performing serious scanning activities should always use a combination of tools, not relying on any one single tool for all their results. As an example, some tools like NetStumbler can be noisy with active probes while tools like Airodump-ng are stealthy and do not make themselves known.

On the UNIX side, two tools top our list. Kismet (www.kismetwireless.net/) uses a curses interface, so users run it from a terminal window, but it visually displays its information (as well as creates detailed logs with GPS data if enabled) in a format that is easy to navigate and understand. In fact, we argue that it is just as easy as using NetStumbler on the windows side when using it via the Kali Live ISO. Newer implementations include more graphical options as well. The other tool to consider is aircrack-ng (www.aircrack-ng.org). It has a great traffic dumping utility but is reserved for more advanced users. This tool has excellent encryption cracking capabilities and was a staple in one author's toolkit during wireless assessments for customers.

From the hardware side, consider getting a high-power Wi-Fi card. For 802.11b and 802.11g targets, high-power cards with external antenna capabilities are readily available through various electronics outlets. You should also consider a good antenna. An omnidirectional antenna is probably the most useful for PCI DSS scanning, though you may have more fun with a "Pringles Can" or another yagi Wi-Fi antenna. 802.11a networks differ slightly in both their frequency and channel designations. Only certain channels are allowed to have a detachable antenna per the

FCC requirements. Outside the United States, check with your regulatory body for specific rules and standards. Finally, since the last writing, we've seen the development of 802.11n and 802.11ac. Both of these mechanisms are great for large data transfers, but don't believe that if you use one you are more safe than any of the other specifications.

WARNING

Keep in mind that illegally modifying equipment can land you in a heap of trouble with the various authorities that govern RF communications. Do yourself a favor and ensure that you do not break the rules!

There are plenty of tutorials for wireless scanning and penetration testing available through your favorite search engine or bookstore. If you are not a professional, your best bet is to leave this particular task to individuals who are. Whether you contract with a security consulting company or choose a wireless hardware vendor, be sure to select the most appropriate technology for your business and risk appetite. As with most things, you get what you pay for!

TOOLS

Here are several examples of good tutorials on wireless scanning and penetration testing:

Essential Wireless Hacking Tools: www.ethicalhacker.net/content/view/16/24/

Wireless Scanning by Andy, IT-Guy: <http://andyitguy.blogspot.com/2008/04/wireless-scanning.html>

Wi-Foo, The Secrets of Wireless Hacking: www.wi-foo.com/

WLAN Security Megaprimer course DVD—<http://www.securitytube.net/downloads>

Use these tools and methods at your own risk.

COMMON MISTAKES AND PITFALLS

Wireless networks should be treated the same as a wired network when it comes to security, with some added hiccups. Remember, wireless stretches the boundaries of your network past your physical walls to areas where you may not have a physical security presence. Per the requirements, one of the most basic prevention measures you can deploy is changing the default settings.

- Change the default passwords. Be sure you work with your manufacturer to find them all. Some APs come with several default accounts with varying levels of security permissions.
- Don't use shared keys. It's just not a good idea anymore and the systems to run individual keys work very well.
- Change the default SSID. The SSID differentiates one network from another. When coming up with an SSID for your AP, don't use the organization's name,

address, or any identifying characteristics that would either draw attention to it or assist an attacker in singling out your company's wireless. Disabling SSID broadcasting characteristics is a good idea as well, but remember that most modern wardriving tools can still extract an SSID from packets over the air even if your AP leaves that field blank while broadcasting its beacon frames. Thus, you should not rely on "SSID Hiding" as a method of security.

- Enable enterprise strength WPA2 or 802.11i for encryption. Preshared keys are not preferable because you have to change them frequently and have to touch every device to do so.

NOTE

Enterprise type authentication or authentication that uses a unique username and password or certificate per device is much preferred over a preshared key. Most network setups can handle this type of authentication with minimal cost of hardware or software. Keep in mind, embedded systems have come under fire recently because a lack of entropy can cause the building blocks of encryption keys to be weakened. Finally, any certificates tied to AD credentials for wireless authentication do not violate Requirement 3.4.1.

WHY IS WEP SO BAD?

WEP has been proven to be a very weak encryption technique to secure a wireless connection. The article, "Breaking 104-bit WEP in less than 60 seconds" (<http://eprint.iacr.org/2007/120.pdf>), discusses how easy it is to break WEP. In a nutshell, there is a 3-byte vector called an initialization vector (IV). The IV is prepended onto packets based on a preshared key that all clients who need to authenticate must know. For most WEP hacks, you will probably only need tools like Kismet and the aircrack-ng suite. These tools can be downloaded freely from the Internet. In addition, a basic tutorial for cracking more advanced encryption methods like WPA and WPA2 can be found at <http://j.mp/AFtgSE>.

CASE STUDY

Wireless compromises give security professionals plenty to write about when it comes to what not to do with respect to wireless. Let's walk through a couple of examples.

THE CASE OF THE UNTETHERED LAPTOP

Ashley's Archery Adventures aims to promote archery among enthusiasts and hunters alike. Ashley started her business last year and has seen steady growth as adults and kids alike take on the challenge of archery. Ashley's business is built on a small 50-acre plot of land just outside of several large suburbs. Luckily, she was close enough to those suburbs to get high-speed Internet access for her office and POS devices.

Ashley spends much of her time out in the field (literally) and has set up several small covered areas where her customers can enjoy water and packed lunches in between archery stations. Because Ashley found herself away from her desk quite often, she put a small Wi-Fi antenna on a modest 100-feet tower by the main office so that she could use her laptop. She took the appropriate precautions to secure her network and used a long preshared key that she changed every quarter.

One day while at the main office, she noticed some strange pop-up windows appearing on her computer and suffered intermittent network blackouts. She installed antivirus and set her automatic patch update to run weekly, so she thought that maybe it was one of her software programs automatically updating itself. If that was not the case, she thought maybe the weather caused her wireless network to go on the fritz. She made a mental note of it and went on with her day. That evening, she noticed that the problem seemed to go away and things were back to normal. Two months later, she learned that she had been compromised.

Ashley was a victim of a common attack against laptops with Wi-Fi cards called the Evil Twin. Ashley frequently visited her local coffee shop on the weekends and used their Wi-Fi connection from her laptop. When the owner of the coffee shop added free Wi-Fi for his customers, he dropped a basic wireless router with default settings on a separate broadband connection for his customers. He didn't want to mess with security settings for his customers and get involved with fixing esoteric problems with each patron's laptop. Default settings seemed to avoid those problems. He also didn't want one of his users to potentially use so much of the Wi-Fi network that his store network was at, or beyond, capacity. This solution worked well for him and his customers.

When Ashley's laptop was acting funny 2 weeks prior, an attacker was cycling through commonly used default SSIDs and got her laptop to associate with his attack machine. Because he provided a stronger signal than the one Ashley was using at the time, her laptop automatically associated with his machine, and he was able to launch an endpoint attack against it. Ashley's laptop had not yet downloaded and installed a patch that fixed a remote vulnerability in the operating system, allowing the attacker to exploit the vulnerability and gain control of her machine. From there, he installed a rootkit and was able to gain access to other machines on the network, including her POS devices. He was also able to grab the Wi-Fi key and casually observe and participate in the network at will.

One of the dangers of wireless networking is the devices that use it. One author has enjoyed watching overly confident security professionals boast about the security of their Wi-Fi networks, only to have a savvy consultant attack a laptop directly (instead of trying to break the network encryption) to gain access. Sometimes computers try to out think their users and do things they "believe" are in the best interest of their users. One of those things is to choose the strongest wireless signal to get the best possible Internet connection.

You can combat this by never "remembering" the networks that you connect to, requiring users to specifically choose each network in which they want to participate.

THE CASE OF THE EXPANSION PLAN

After learning her lesson, Ashley quickly cleaned up her breach and was able to refocus on her business. She kept her wireless network intact, but she added additional protection with a host-based firewall for her laptop and then removed all stored profiles except for her office Wi-Fi. Her device is the only authorized device on the network at this time, but part of her improvement plan for this year is to put small POS terminals in some of the covered areas. She plans to offer more products for sale like cold beverages and food or snack items. Each covered area would have a PC to keep track of inventory or allow employees to send quick notes via instant messages or e-mail.

Ashley has budgeted a small amount of money to purchase both the hardware for the expansion plan and to build a small back-office network to maintain these devices. She wanted to provide network services to the devices to back them up daily and put important files on a network file server for all machines to use. Being as these machines would be somewhat exposed to the elements, she knew that equipment failure was a much bigger possibility than if those devices were kept in a climate-controlled office environment.

She purchases and deploys a Microsoft Windows Server and sets up her new employees in Active Directory with usernames and passwords. Each machine must log into the domain, and her “Rent-an-IT-Guy” sets up some basic network shares and permissions for each user. After he finished setting everything up, he left a small easy-to-follow guide for Ashley should she need to make minor changes.

Now Ashley must decide how she wants these machines to connect to the wireless network. It’s impractical to go change Wi-Fi keys on these devices by hand every quarter, so she wants to find an automated solution or replace shared keys all together. What should she do?

Ashley has two choices. The first is to have her “Rent-an-IT-Guy” write a script that will change the Wi-Fi keys automatically on each PC in the field. This gives Ashley the advantage of keeping her existing setup that she is familiar with, but allowing the machines to easily change their keys while avoiding the headache of visiting each terminal and typing in the complex key by hand.

Ashley’s second choice is to upgrade her Wi-Fi router to one that will support an enterprise authentication scheme with 802.11i or WPA2. From there, she can add an agent into each field computer’s installation that requires users to authenticate with the network first with their existing username and password. This username and password could be part of their Active Directory credentials and would only require setting up a RADIUS server on the domain controller to enable this functionality.

Before security purists jump down the authors’ throats, yes, we do realize that a single username and password that accesses all resources may add additional risk of compromise. That said, in this instance, the risk is relatively low provided that each user receives training on how to create good passwords or pass phrases, and they are changed regularly. Alternatively, Ashley could deploy an inexpensive token-based solution to provide a second factor of authentication that would effectively remove this weakness.

THE CASE OF THE DOUBLE SECRET WIRELESS NETWORK

James's Junker Jubilee, a car rental facility that rents run down automobiles for a fraction of the cost of a traditional rental company, recently went through a PCI assessment. Upon arriving at their corporate headquarters, the assessors were placed into a conference room for the duration of the assessment. When it came time to ask about wireless technologies, Sally, a risk manager, proudly stated that wireless technologies were prohibited at James's and that employees found using these technologies were reprimanded with penalties up to termination.

The lead assessor casually looked over in the corner of the conference room where the audio/visual (A/V) equipment was stored and pointed out a blue device with two antennas on it. It oddly enough had a label on it that proclaimed "Property of IT, DO NOT REMOVE." When the lead assessor examined it closer, it was an AP from a well-known supplier and was plugged into the Ethernet port in the wall. The assessor had an older model in his house, so he was sure he was looking at an AP and not something related to the A/V features of the conference room. Sally looked at the device and said, "Well, that showed up last month and we just assumed it belonged to IT and didn't touch it. The CIO has a lot of political power at James's, and most employees have learned not to cross him."

It turns out, a hacker posing as a flower deliveryman gained access to the facility around Valentine's Day that year and placed the curiously labeled device in the conference room. He had been poking around the network ever since and had stolen both customer data and intellectual property. Because he had used several techniques to hide the device, the basic wireless sweeps that the company was performing did not pick up that device.

Had James's used a Wireless IDS or IPS to bolster his security instead of relying on his quarterly wireless analysis as part of Requirement 11.1, chances are he would have had a much better chance of catching the hacker in the act and shutting down the connection before the breach of data could occur.

The case of the detached POS

Karen's Kupcakes is a small bakery specializing in building delightful desserts using the cupcake as its foundation. Karen located her business in a small strip mall that is across a six-lane avenue from a large shopping mall. This year for Valentine's Day, she wants to focus on spreading the word about her business by opening a small kiosk in the parking lot outside the food court. She will sell Valentine's Day themed cupcakes with on-site customization to patrons entering and leaving the mall area. Since she will be doing the primary baking in her location, the only power she will need is for her register to process payments for the cupcakes.

She looks at a few options like shooting 802.11a Wi-Fi across the avenue, but ultimately settles on purchasing a remote payment processing terminal that will seamlessly integrate with her existing setup. She leases the equipment from a known provider, has them process the payments on her behalf and send over wire transfers at the end of each day. She has a security consultant take a look at the setup to ensure

it complies with PCI DSS as she doesn't want her customers to be subject to a breach on her behalf. It is powered through GSM networks, so she knows that she needs to have her traffic encrypted before it hits the cellular hardware.

Overall, her kiosk was a massive success, and well covered the investment she made in the portable payment processing hardware. Her efforts to process these payments securely paid off, and she plans to expand to multiple other kiosks in the near future.

SUMMARY

Wireless networking can be both safe and effective in extending your network's functionality for special events or as a normal course of business. From mobile users who have all-in-one devices that manage their inventory, schedules, and payments, to that cash machine at an outdoor arts festival, to a trendy bar with an outdoor patio and deck area in the spring and fall, wireless payments are here to stay.

As we have learned by reading this chapter, there are several things that companies must be aware of when they venture into potentially uncharted waters of mobile or wireless payments. This goes without saying, but be sure you understand both the technology you are implementing and have a trusted third-party review it for compliance and security (the latter probably being much more important in the grand scheme of things). Too many companies have ventured down this road with big ideas only to deploy an insecure technology and end up with a massive compromise bill.

Finally, pay close attention to the security features of your particular infrastructure components. Use all the capacity available (that makes the most sense for your network and setup), in your keys, and use the best encryption available.

Vulnerability management

9

INFORMATION IN THIS CHAPTER:

- PCI DSS requirements covered
- Vulnerability management in PCI
- Requirement 5 walk-through
- Requirement 6 walk-through
- Requirement 11 walk-through
- Internal vulnerability scanning
- Common PCI vulnerability management mistakes
- Case study

Before we discuss Payment Card Industry (PCI) requirements related to vulnerability management in-depth and find out what technical and nontechnical safeguards are prescribed there and how to address them, we need to address one underlying and confusing issue of defining some of the terms that the PCI Data Security Standard (DSS) documentation relies upon.

These are as follows:

- Vulnerability assessment;
- Penetration testing;
- Testing of controls, limitations, and restrictions;
- Preventing vulnerabilities via secure coding practices.

Defining vulnerability assessment is a little tricky, since the term has evolved over the years. The authors prefer to define it as a process of finding and assessing vulnerabilities on a set of systems or a network, which is a very broad definition. By the way, the term *vulnerability* is typically used to mean a software flaw or a weakness that makes the software susceptible to an attack or abuse. In the realm of information security, a vulnerability assessment is usually understood to be a vulnerability scan of the network with a scanner, implemented as installable software, a dedicated hardware appliance, or a scanning software-as-a-service (SaaS). Sometimes using the term *network vulnerability assessment* adds more clarity to this. The terms *network vulnerability scanning* or *network vulnerability testing* are usually understood to mean the same. In addition, the separate term *application vulnerability assessment* is typically understood to mean an assessment of application-level vulnerabilities in a particular application; most frequently, a Web-based application deployed publicly

on the Internet or internally on the intranet. A separate tool called an application vulnerability scanner (as opposed to the network vulnerability scanner mentioned above), is used to perform an application security assessment. By the way, concepts such as “port scan,” “protocol scan,” and “network service identification” belong to the domain of network vulnerability scanning, whereas concepts such as “site crawl,” “Hypertext Transfer Protocol (HTTP) requests,” “cross-site scripting,” and “client side vulnerabilities” belong to the domain of Web-application scanning (WAS). We will cover both types in this chapter as they are mandated by the latest version of PCI DSS, albeit in different requirements (6 and 11).

If you remember one thing from this chapter, please remember that a vulnerability assessment and a penetration test are completely and unquestionably different. Penetration testing is usually understood to mean an attempt to break into the network by a dedicated team, which can use the network and application scanning tools mentioned above and also other nontechnical means such as dumpster diving (i.e., looking for confidential information in the trash) or social engineering (i.e., attempting to subvert authorized information technology [IT] users to give out their access credential and other confidential information). Sometimes, penetration testers might rely on other techniques and methods such as custom-written attack tools. In fact, anybody who tries to sell you a penetration test, but only plans to run a vulnerability assessment tool against your systems is probably a scammer out to make a quick buck.

Testing of controls, mentioned in Requirement 11.1, does not have a simple definition. Sometimes referred to as a “site assessment” (but not to be confused with a Qualified Security Assessor [QSA] assessment), such testing implies either an in-depth assessment of security practices and controls by a team of outside experts or a diligent self-assessment by a company’s own staff. Such controls assessment will likely not include attempts to break into the network.

Preventing vulnerabilities, covered in Requirement 6, addresses vulnerability management by assuring that newly created software does not contain known flaws and problems. Requirements 5, 6, and 11 also mandate various protection technologies such as antivirus, Web firewalls, intrusion detection or prevention, and others.

The core of Requirement 11 discussed in this chapter covers all the above and some of the practices that help mitigate the impact of problems, such as the use of intrusion prevention tools. Such practices fall into broad domains of vulnerability management and threat management. Although there are common definitions of vulnerability management (covered below), threat management is typically defined ad hoc as “dealing with threats to information assets.”

PCI DSS REQUIREMENTS COVERED

Vulnerability management controls are present in PCI DSS Requirements 5, 6, and 11.

- PCI Requirement 5 “Protect all systems against malware and regularly update anti-virus software or programs” covers antimalware measures (albeit from a weaker signature-basis); these are tangentially related to what is commonly seen as vulnerability management, but it helps deal with the impact of vulnerabilities.

- PCI Requirement 6 “Develop and maintain secure systems and applications” covers a broad range of application security subjects, application vulnerability scanning, secure software development, and so on.
- PCI Requirement 11 “Regularly test security systems and processes” covers a broad range of security testing, including network vulnerability scanning by approved scanning vendors (ASVs), internal scanning, and other requirements. We will focus on Requirements 11.2 and 11.3 in this chapter.

VULNERABILITY MANAGEMENT IN PCI

Before we start our discussion of the role of vulnerability management for PCI compliance, we need to briefly discuss what is commonly covered under vulnerability management in the domain of information security. It appears that some industry pundits have proclaimed that vulnerability management is simple: just patch all those pesky software problems and you are done. Others struggle with it because the scope of platforms and applications to patch and other weaknesses to rectify is out of control in most large, diverse organizations. The problems move from intractable to down-right scary when you consider all the Web applications being developed in the world of HTML 5 and cloud computing, including all the in-house development projects, outsourced development efforts, partner development, and so on. And we have not even mentioned the mobile application—many of which handle payments directly.

Such applications may never get that much-needed patch from the vendor because you are simultaneously a user and a vendor; a code change by your own engineers might be the only way to solve the issue.

Thus, vulnerability management is not the same as just keeping your systems patched; it expands into software security and application security, secure development practices, and other adjacent domains. If you are busy every first Tuesday when Microsoft releases its batch of patches, but not doing anything to eliminate a broad range of application vulnerabilities during the other 29 days in a month, you are not managing your vulnerabilities efficiently, if at all. Vulnerability management *was* a mix of technical and nontechnical process even in the time when patching was most of what organizations needed to do to stay secure. Nowadays, it touches an even bigger part of your organization: not only network group, system group, desktop group, but also your development and development partners, and possibly even individual businesses deploying their own, possible “in the cloud” or mobile applications (it is not unheard of that such applications will handle or contain payment card data).

Clearly, vulnerability management is not only about technology and “patching the holes.” As everybody in the security industry knows, technology for discovering vulnerabilities is getting better every day. Moreover, the same technology is also used to detect configuration errors and nonvulnerability related security issues. For instance, a fully patched system is still highly vulnerable if it has a blank administrator or root password, even though no patch is missing. The other benefit derived from vulnerability management is the detection of “rogue” hosts, which are sometimes deployed by business units and are sitting outside of control of your IT department,

and thus, might not be deployed in adherence with PCI DSS requirements. One of the basic tenets of adhering to the PCI standard is to limit the scope of PCI by strictly segmenting and controlling the cardholder data environment (CDE). Proper implementation of internal and external vulnerability scanning can assist in maintaining a pristine CDE.

As a result, it's useful to define vulnerability management as managing the life-cycle of processes and technologies needed to discover and then reduce (or, ideally, remove) the vulnerabilities in software required for the business to operate, and thus, bring the risk to business information to the acceptable threshold.

Network vulnerability scanners (and is how vulnerability assessment tools are commonly called, even if some may use an agent on the machine and not an actual port scan) can detect vulnerabilities from the network side with good accuracy and from the host side with better accuracy. Host-side detection is typically accomplished via internal scans by using credentials to log into systems (the so-called “authenticated” or “trusted” scanning), so configuration files, registry entries, file versions, and so on can be read, thus increasing the accuracy of results. Such scanning is performed only from inside the network, not from the Internet.

NOTE

Sometimes vulnerability scanning tools are capable of running trusted or authenticated scanning where the tools will actually log into a system, just like a regular user would, and then perform the search for vulnerabilities. If you successfully run a trusted or authenticated scan from the Internet and discover configuration issues and vulnerabilities, you have a serious issue because no one should be able to directly log into hosts or network devices directly from the Internet side, whether to network device or servers. Believe it or not, this has happened in real production environments, subject to PCI DSS! Also, PCI ASV scanning procedures prohibit authentication scanning when performing ASV scan validation. Authenticated or trusted scans are extremely useful for PCI DSS compliance and security, but they should always be performed from inside the network perimeter.

However, many merchants that implement periodic vulnerability scanning at higher than a quarterly frequency have discovered that the volumes of data generated far exceed their expectations and abilities. A quick scan-then-fix approach turns into an endless wheel of pain, obviously more dramatic for PCI DSS external scanning because you have no choice in fixing vulnerability, which leads to validation failure (we will review the exact criteria for failure below). Many free and low-cost commercial-vulnerability scanners suffer from this more than their more advanced brethren; thus, exacerbating the problem for price-sensitive organizations such as smaller merchants. Using vulnerability scanners presents other challenges, including having network visibility of the critical systems, perceived or real impact on the network bandwidth, as well as system stability. Overall, vulnerability management involves more process than technology and your follow-up actions should be based on the overall risk and not simply on the volume of incoming scanner data.

STAGES OF VULNERABILITY MANAGEMENT PROCESS

Let's outline some critical stages of the vulnerability management process. Those include defining the policy, collecting the data, deciding what to remediate (i.e., fix for good) or mitigate (i.e., temporarily protect from exploitation) and then taking action.

Policy definition

Indeed, the vulnerability management process starts from the policy definition that covers your organization's assets, such as systems and applications and their users, as well as partners, customers, and whoever touches those resources. Such documents and the accompanying detailed security procedures define the scope of the vulnerability management effort and create a "known good" state of those IT resources. Policy creation should involve business and technology teams, as well as senior management who would be responsible for overall compliance. PCI DSS requirements directly affect such policy documents and mandate its creation (see Requirement 12 that states that one needs to "maintain a policy that addresses information security"). Marking the assets that are in scope for PCI compliance is also part of this step.

Data acquisition

The data acquisition process comes next. A network vulnerability scanner or an agent-based host scanner is a common choice; a passive vulnerability assessment tool that sniffs network traffic can also be used. Both excellent freeware and commercial solutions are available. In addition, established standards for vulnerability naming, such as CVE (<http://cve.mitre.org>) and vulnerability scoring, such as Common Vulnerability Scoring System ([CVSS], www.first.org/cvss) can help provide a consistent way to encode vulnerabilities, weaknesses, and organization-specific policy violations across the popular computing platforms. CVSS, specifically, is utilized for measuring vulnerability severity in PCI DSS. Moreover, an effort by US National Institute of Standards and Technology called Security Content Automation Protocol (SCAP) has combined the above standards into a joint standard bundle to enable more automation of vulnerability management. See <http://scap.nist.org> for more details on SCAP.

Scanning for compliance purposes is somewhat different from scanning purely for remediation. Namely, PCI DSS reports that QSA will ask for to validate your compliance should show the list of systems that were scanned for all PCI-relevant vulnerabilities, as well as an indication that all systems test clean. Scanning tools also provide support for Requirements 1 and 2, secure-system configurations, and many other requirements described below. This shows that while "scanning for remediation" only requires a list of vulnerable systems with their vulnerabilities, "scanning for compliance" also calls for having a list of systems found not to be vulnerable.

Prioritization

The next phase, prioritization, is a key phase in the entire process. It is highly likely that even with a well-defined specific scan policy (which is derived from PCI DSS requirements, of course) and a quality vulnerability scanner, the amount of data on various vulnerabilities from a large organization will be enormous. Even looking at

the in-scope systems might lead to such data deluge; it is not uncommon for an organization with a flat network to have thousands of systems in scope for PCI DSS. No organization will likely “fix” all the problems, especially if their remediation is not mandated by explicit rules. Some kind of prioritization will occur. Various estimates indicate that even applying a periodic batch of Windows patches (“black” Tuesday) often takes longer than the period between patch releases (longer than 1 month). Accordingly, there is a chance that the organization will not finish the previous patching round before the next one rushes in. To intelligently prioritize vulnerabilities for remediation, you need to take into account various factors about your own IT environment as well as the outside world. Ideally, such prioritization should not only be based on PCI DSS but also on organization’s view and approach to information risk (per the note in Requirement 6.1). Also, even when working within PCI DSS scope, it makes sense to fix vulnerability with higher risk to card data first, even if this is not mandated by PCI DSS standards.

Those include the following:

- Specific regulatory requirement: fix all medium- and high-severity vulnerabilities as indicated by the scanning vendor; fix all vulnerabilities that can lead to Structured Query Language (SQL) injection, cross-site scripting attacks, and so on.
- Vulnerability severity for the environment: fix all other vulnerabilities on publicly exposed and then on other in-scope systems.
- Related threat information and threat relevance: fix all vulnerabilities on the frequently attacked systems.
- Fix vulnerabilities that have exploit code publicly available (and exploitation ongoing), and/or those used by malware
- Business value and role information about the target system: address vulnerabilities on high-value critical servers.

To formalize such prioritization, one can use the CVSS (www.first.org/CVSS), which takes into account various vulnerability properties such as priority, exploitability, and impact, as well as multiple, local site-specific properties. The CVSS scheme offers a way to provide a uniform way of scoring vulnerabilities on a scale from 0 to 10. PCI DSS mandates the use of CVSS by ASVs; moreover, PCI validation scanning prescribes that all vulnerability with the score equal to or higher than 4.0 must be fixed to pass the scan. National Vulnerability Database (NVD) located at <http://nvd.nist.org> provides CVSS scores for many publicly disclosed vulnerabilities (Figure 9.1). CVSS, however, can be enhanced to also account for exploit code availability and reports of active exploitation—to fix these vulnerabilities first for maximum risk reduction.

PCI DSS 2.0 also introduced (and PCI DSS 3.0 retained) the concept of a threshold for internal vulnerability scanning. Validation procedure 11.2.1.b stats that a QSA must “Review the scan reports and verify that the scan process includes rescans until all ‘high-risk’ vulnerabilities as defined in PCI DSS Requirement 6.1 are resolved” for internal scanning.

The screenshot shows the NIST National Vulnerability Database homepage. The URL in the address bar is <http://nvd.nist.gov/vuln/detail.cgi?cve=CVE-2008-6876>. The page title is "National Vulnerability Database". The main content area displays the following information for CVE-2008-6876:

- Vulnerability Summary for CVE-2008-6876**
- Original release date: 07/24/2009
- Last revised: 07/24/2009
- Source: US-CERT/NIST
- Overview**: Cross-site scripting (XSS) vulnerability in login.php in EsPeritaires 1.0 allows remote attackers to inject arbitrary web script or HTML via the msg parameter. NOTE: the EsPeritaires 1.0 issue is covered in CVE-2008-2037.
- Impact**:
 - CVE Score for Version 2.0: CVSS v2 Base Score 4.6 (MEDIUM) [AV:N/AC:M/U:N/C:N/I:N/O:N] (legend)
 - Impact Subscore: 2.9
 - Exploitability Subscore: 8.6
 - CVSS Version 2 Metrics:
 - Access Vector: Network Exploitabile; Victim must voluntarily interact with attack mechanism
 - Access Complexity: Medium
 - Authentication: not required to exploit
 - Impact Type: Allows unauthorized modification
- References to Advisories, Solutions, and Tools**: By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this page. There may be other web sites that are more appropriate for your needs. NIST does not necessarily endorse the views expressed, or concur with the facts presented on these sites. Further, NIST does not endorse any commercial products that may be mentioned on these sites. Please address comments about this page to nvd@nist.gov.

FIGURE 9.1 National Vulnerability Database

Mitigation

The next phase of mitigation is important in many environments where immediate patching or reconfiguration is impossible, such as a critical server running unusual or custom-built applications. Despite the above, in some cases, when a worm is released or a novel attack is seen in similar environments, protecting such a system becomes unavoidable. In this case, you immediately need to do something to mitigate the vulnerability temporarily. This step might be performed by a host or network intrusion prevention system (IPS); sometimes even a firewall blocking a network port will do. The important question here is choosing the best mitigation strategy that won't create additional risk by blocking legitimate business transactions. In case of a Web-application vulnerability, a separate dedicated device such as a Web-application firewall needs to be deployed in addition to the traditional network security safeguards such as firewalls, filtering routers, IPSs, and so on.

In this context, using antivirus and intrusion prevention technologies might be seen as part of vulnerability mitigation because these technologies help protect companies from vulnerability exploitation (either by malware or human attackers).

Ideally, all vulnerabilities that impact card data need to be fixed, such as patched or remediated in some other way prescribed by the above prioritization procedure taking into account the steps we took to temporarily mitigate the vulnerability above. In a large environment, it is not simply the question of “let's go patch the server.” Often, a complicated workflow with multiple approval points and regression testing on different systems is required.

To make sure that vulnerability management becomes a process, an organization should monitor vulnerability management on an ongoing basis. This involves looking at the implemented technical and process controls aimed at decreasing risk.

Such monitoring goes beyond vulnerability management into other security management areas. It is also important to be able to report to senior management about your progress.

Vulnerability management is not a panacea even after all the “known” vulnerabilities are remediated. “Zero-day” attacks, which use vulnerabilities with no resolution publicly available, will still be able to cause damage. Such cases need to be addressed by using the principle of “defense in-depth” during the security infrastructure design.

Now, we will walk through all the requirements in PCI DSS guidance that are related to vulnerability management. We should again note that vulnerability management guidance is spread across Requirements 5, 6, and 11.

REQUIREMENT 5 WALK-THROUGH

While antivirus solutions have little to do with finding and fixing vulnerabilities, in PCI DSS they are covered under the broad umbrella definition of vulnerability management. One might be able to argue that antivirus solutions help when a vulnerability is present and is being exploited by malicious software such as a computer virus, worm, Trojan horse, or spyware. Thus, antivirus tools help mitigate the consequences of exploited vulnerabilities in some scenarios.

In PCI DSS 3.0, Requirement 5.1 mandates the organization to “Protect all systems against malware and regularly update anti-virus software or programs.” Indeed, many antivirus vendors moved to daily (and some to hourly) updates of their virus definitions. Needless to say, virus protection software is next to useless without an up-to-date malware definition set.

PCI creators wisely chose to avoid the trap of saying “antivirus must be on all systems,” but instead chose to state that you need to “deploy anti-virus software on all systems commonly affected by malicious software (particularly personal computers and servers).” This ends up causing a ton of confusion, and in many cases, companies fight deploying antivirus, even when the operating system manufacturer recommends it (e.g., Apple’s OS X or even Android for mobile devices). A good rule of thumb is to deploy it on all Microsoft Windows machines and any desktop machine with users regularly accessing untrusted networks (like the Internet) where an antimalware solution is available.

Now in version 3.0, PCI also explains how to check for what systems are in fact subject to malware (Requirement 5.1.2): “For systems considered to be not commonly affected by malicious software, perform periodic evaluations to identify and evaluate evolving malware threats in order to confirm whether such systems continue to not require anti-virus software.” For example, if many merchants end up using Android-based devices, this line of thinking will help them determine whether they need antimalware there.

Subsection 5.1.1 states that one needs to “Ensure that anti-virus programs are capable of detecting, removing, and protecting against all known types of malicious software.” They spell out all the detection, protection, and removal of various types

of malicious software, knowing full well that such protection is highly desirable, but not really achievable, given the current state of malware research. In fact, recent evidence suggests that an antivirus product's ability to protect you from all the malware is less certain everyday as more backdoors, Trojans, rootkits, and other forms of malware enter the scene.

Finally, Sections 5.2 and 5.3 drive the point home: “Ensure that all anti-virus mechanisms are maintained as follows: a) Are kept current, b) Perform periodic scans and c) Generate audit logs” and “Ensure that anti-virus mechanisms are actively running and cannot be disabled or altered by users, unless specifically authorized by management on a case-by-case basis for a limited time period.” A new (in PCI DSS 3.0) Requirement 5.4 “Ensure that security policies and operational procedures for protecting systems against malware are documented, in use, and known to all affected parties” adds the concept of operational procedures related to fighting malware.

This combines three different requirements, which are sometimes overlooked by organizations that deployed antivirus products. First, they need to be current—updated as frequently as their vendor is able to push updates. Daily, not weekly or monthly, is a standard now. Second, the running status of security tools needs to be monitored. Third, as mentioned in Chapter 10, logs are critical for PCI compliance. This section reminds PCI implementers that antivirus tools also need to generate logs, and such logs need to be reviewed in accordance with Requirement 10. Expect your Assessor to ask for logs from your antimalware solution to substantiate this requirement, including logs from the A/V console.

WHAT TO DO TO BE SECURE AND COMPLIANT?

1. Requirement 5 offers simple and obvious action items:
 - a. Deploy antivirus software on in-scope systems, wherever such software is available and wherever the system can suffer from malware. Free antivirus products can be downloaded from several vendors such as AVG (go to <http://free.avg.com> to get the software) or Avast (go to www.avast.com to get it).
 - b. Assess that systems that are not thought to be subject to malware really are not (in compliance with PCI DSS 3.0 requirement 5.1.2).
2. Configure the obtained antimalware software to update at least *daily*. Please forget the security advice from the 1990s when weekly updates were seen as sufficient. Daily is the minimum acceptable update frequency today—but be prepared that in many cases it will be way too late as malware will already be on your systems. This will deal with Requirement 5.1.1—but might not provide enough security for your organization.
3. Verify that your antivirus software is generating audit logs. This will take care of Requirement 5.2. Please refer to Chapter 10 to learn how to deal with all the logs, including antivirus logs.
4. Document operational procedures related to fighting malware (QSAs are required to review such documents!)

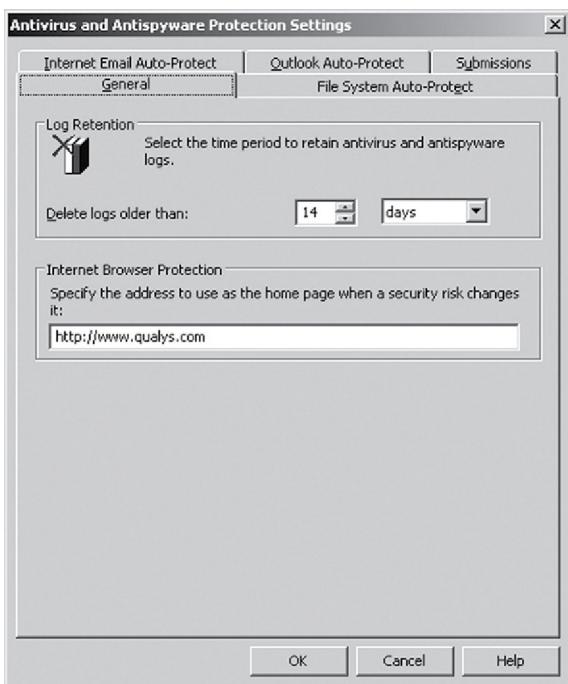


FIGURE 9.2 Antivirus Log Setting

NOTE

For example, Symantec AntiVirus and many other enterprise-grade antimalware tools will log all detections by default; there is no need to “enable logging.” To preserve, please make sure that the setting shown in [Figure 9.2](#) allows your centralized log collection system to get the logs before they are deleted.

REQUIREMENT 6 WALK-THROUGH

Another requirement of PCI covered under the vulnerability management umbrella is Requirement 6, which covers the need to “develop and maintain secure systems and applications.” Thus, it touches vulnerability management from another side: making sure that those pesky flaws and holes never appear in software in the first place. At the same time, this requirement covers the need to plan and execute a patch-management program to assure that, once discovered, the flaws are corrected via software vendor patches or other means. In addition, it deals with a requirement to scan applications, especially Web applications, for vulnerabilities.

Thus, one finds three types of requirements in Requirement 6: those that help you patch the holes in commercial applications, those that help you prevent holes in the in-house developed applications, and those that deal with verifying the security of Web application (Requirement 6.6).

Requirement 6.1 (“Establish a process to identify security vulnerabilities, using reputable outside sources for security vulnerability information, and assign a risk ranking” to them) starts the vulnerability management process by prescribing to find and rank the vulnerabilities. Note that this doesn’t mean “scanning for vulnerabilities” in your environment, but looking for newly discovered vulnerabilities via vulnerability alert services, some of which are free such as the one from Secunia (see www.secunia.com), whereas others are targeted at enterprises. Some services can be highly customized to only send alerts applicable to your environment, and also fixes for vulnerabilities that are not public. They are not free, but may surely be worth the money paid for them. You can also monitor public mailing lists for vulnerability information (BugTraq is a primary example: www.securityfocus.com/archive/1), which usually requires a significant time commitment. Twitter can be useful here as certain streams contain excellent real-time vulnerability data.

Furthermore, Requirement 6.2 states that an organization must “Ensure that all system components and software are protected from known vulnerabilities by installing applicable vendor-supplied security patches. Install critical security patches within one month of release” [2]. Then, it covers the prescribed way of dealing with vulnerabilities in custom, homegrown applications: careful application of secure coding techniques, and incorporating them into a standard software-development lifecycle. Specifically, the document says “for in-house developed applications, numerous vulnerabilities can be avoided by using standard system-development processes and secure coding techniques.” Finally, it addresses the need to test the security of publicly exposed Web applications by mandating that “for public-facing Web applications, address new threats, and vulnerabilities on an ongoing basis.”

Apart from requiring that organizations “ensure that all system components and software have the latest vendor-supplied security patches installed,” Requirement 6.2 attempts to settle the debates in security industry, which is between a need for prompt patching in case of an imminent threat and a need for careful patch testing. They take the simplistic approach of saying that one must install patches within 1 month. Such an approach, while obviously “PCI-compliant,” might sometimes be problematic: 1 month is way too long in case of a worm outbreak (all vulnerable systems will be firmly in the hands of the attackers), and on the contrary, too short in case of complicated mission-critical systems and overworked IT staff (such as for database and enterprise application patches). Therefore, a later clarification was added, which explicitly mentions a risk-based approach. Specifically, this can allow an organization to get an extension to the above 1-month deadline “security patches for critical or at-risk systems are installed within 30 days, and other lower-risk patches are installed within 2-3 months.”

NOTE

If you decide to use public mailing lists, you need to have a list of all operating systems and commercial software that is in-scope. You may want to set up a specific mailbox that multiple team members have access to, so new vulnerabilities are not “missed” when someone is out of the office. Checking these lists as part of your normal Security Operation Center (SOC) analyst duties can help ensure this activity regularly takes place. Even if your organization is small and does not have a real SOC with rows of security analysis dedicated to security monitoring or a virtual SOC, checking the lists and services frequently will help satisfy this requirement.

Other aspects of your vulnerability management program apply to securing the software developed in-house. Section 6.3 states that one needs to “Develop internal and external software applications (including Web-based administrative access to applications) securely.” The unfortunate truth, however, is that there is no single authoritative source for such security “best practices” and, at the same time, current software “industry best practices” rarely include “information security throughout the software-development life cycle.” Here are some recent examples of projects that aim at standardizing security programming best practices, which are freely available for download and contain detailed technical guidance:

- BSIMM “The Building Security In Maturity Model”; see www.bsi-mm.com/;
- OWASP “Secure Coding Principles”; see www.owasp.org/index.php/Secure_Coding_Principles;
- SANS and MITRE “CWE/SANS TOP 25 Most Dangerous Programming Errors”; see www.sans.org/top25errors/ or <http://cwe.mitre.org/top25/>;
- SAFECode “Fundamental Practices for Secure Software Development”; see www.safecode.org/.

Detailed coverage of secure programming topic goes far beyond the scope of this book.

Sections 6.3 and 6.4 go over software development and maintenance practices. Requirement 6.3 mandates that for PCI compliance, an organization must develop applications “In accordance with PCI DSS (for example, secure authentication and logging), based on industry standards and/or best practices and Incorporating information security throughout the software-development life cycle.” This guidance is obviously quite unclear and the burden of making the judgment call is on the QSAs, who are rarely experts in secure application development lifecycle as well as in general in the esoteric aspects of software security.

Let’s review some of the subrequirements of 6.4, which are clear, specific, and leave the overall theme of “following industry best practices” to your particular QSA.

The next one simply presents security common sense (Requirement 6.4.1): “Separate development/test environments from production environments, and enforce the separation with access controls.” This is very important because some recent attacks penetrated publicly available development and testing or staging sites. Take note of the enforcement provision added in PCI DSS 3.0!

A key requirement 6.4.3, which states “production data (live primary account numbers [PANs]) are not used for testing or development,” is one that is the most critical and also the most commonly violated, and with the most disastrous consequences. Many companies found their data was stolen because uninformed developers moved data from the more secure production environment to a much less-protected test environment, as well as on mobile devices (laptops), remote offices, and so on.

On the contrary, contaminating the production environment with test code, utilities, and accounts is also critical (and was known to lead to just as disastrous compromises of production data) and is covered in Section 6.4.4, which regulate the use of “test data and accounts” and prerelease “custom application accounts” “custom code.” Similarly, recent attackers have focused on looking for left over admin logins, test code, hard-coded passwords, and so on.

The next requirement is absolutely a key to application security (6.3.2): “Review of custom code prior to release to production or customers to identify any potential coding vulnerability.” Please also pay attention to the clarification to this requirement: “This requirement for code reviews applies to all custom code (both internal and public-facing and custom scripts) as part of the system development lifecycle. Do you have an administrative script that automates some regular task? It’s in scope! This mandates application security code review for in-scope and public applications. This scoping statement is of supreme importance: BOTH public (all public) and internal in-scope applications must be secured. (“This requirement for code reviews applies to all custom code (both internal and public-facing), as part of the system development life cycle.”)

Furthermore, Section 6.4 covers a critical area of IT governance: change control. Change control can be considered a vulnerability management measure because unpredicted, unauthorized changes often lead to opening vulnerabilities in both custom and off-the-shelf software and systems. It states that one must “follow change control procedures for all system and software configuration changes,” and even helps the organization define what the proper procedures must include (Requirement 6.4.5 and all its subrequirements).

The simple way to remember is: if you change something somewhere in your IT environment, document it. Whether it is a bound notebook (small company) or a change control system (large company) is secondary, leaving a record is primary.

To put this into context, most other IT governance frameworks, such as COBIT (www.isaca.org/cobit) or ITIL (www.itil.co.uk/), cover change control as one of the most significant areas that directly affect system security. Indeed, having documentation and sign-off for changes and an ability to “undo” things will help achieve both security and operation goals by reducing the risk, and striving toward operational excellence. Please refer to Chapter 14 to learn how to combine multiple compliance efforts.

Another critical area of PCI DSS covers Web-application security; it is contained in Sections 6.5 and 6.6 that go together when implementing compliance controls.

WEB-APPLICATION SECURITY AND WEB VULNERABILITIES

Section 6.5 has been expanded beyond Web applications, because it is the type of application that will more likely be developed in-house and, at the same time, more likely to exposed to the hostile Internet (a killer combo—likely less-skilled programmers with larger number of malicious attackers!). Companies that outsource to the lowest price are particularly vulnerable because they may not have the skills to validate the code coming back as secure.

The requirement now prescribes the need to “Address common coding vulnerabilities in software-development processes”. Still, fewer organizations choose to write their own Windows or Unix software from scratch compared to those creating or customizing Web-application frameworks.

In addition, it also calls to “review custom-application code to identify coding vulnerabilities.” It adds that companies must follow “industry best practices for vulnerability management … (e.g., the OWASP Guide, SANS CWE Top 25, CERT Secure Coding, etc.), the current best practices must be used for these requirements.”

Although a detailed review of secure coding goes much beyond the scope of this book, there are many other books devoted to the subject including detailed coverage of secure Web-application programming, Web-application security, and methods for discovering Web site vulnerabilities well beyond the scope of this book. See *Hacking Exposed Web Applications*, Third Edition and *HackNotes™ Web Security Portable Reference* for more details. OWASP has also launched a project to provide additional guidance on satisfying Web-application security requirements for PCI (see “OWASP PCI” online).

PCI DSS goes into great level of details here, covering common types of coding-related weaknesses in Web applications. Those are as follows:

- “6.5.1 Injection flaws, particularly SQL injection. Also consider OS Command Injection, LDAP and XPath injection flaws as well as other injection flaws.”
- “6.5.4 Insecure communications.”
- “6.5.6 All “High” vulnerabilities identified in the vulnerability identification process (as defined in PCI DSS Requirement 6.2).”
- “6.5.7 Cross-site scripting (XSS).”
- “6.5.9 Cross-site request forgery (CSRF).”

and others (see PCI DSS Requirement 6.5).

In addition to secure coding to prevent vulnerabilities, organizations might need to take care of the existing deployed applications by looking into Web-application firewalls (WAFs) and WAS, sometimes also called Dynamic Application Security Testing (DAST). An interesting part of Requirement 6.6 is that PCI DSS recommends either a vulnerability scan followed by a remediation (“Reviewing public-facing web applications via manual or automated application vulnerability security assessment tools”) or a WAF (“Installing a application firewall in front of public-facing web applications”), completely ignoring the principal of layered defense or defense-in-depth. In reality, deploying both is highly recommended for effective protection of

Web applications. In fact, PCI DSS makes a huge mistake here allowing the WAF to be deployed in monitoring mode, that is, with no actual protection: WAF can in fact be “configured to either block Web-based attacks, or generate an alert.” (so, if you want to be compliant, but blatantly not secure, you can follow that bit of advice!)

WAS

Before progressing with the discussion of WAS, we need to remind our readers that cross-site scripting and SQL injection Web site vulnerabilities account for a massive percentage of card data loss. These same types of Web vulnerabilities are also very frequently discovered during PCI DSS scans. For example, Qualys vulnerability research indicates that cross-site scripting is one of the most commonly discovered vulnerabilities seen during PCI scanning; it was also specifically called out by name in one PCI Council document [2] as a vulnerability that leads to PCI validation failure.

You need to ensure that whichever solution you use covers the current OWASP Top Ten list. This list may change over time and you need to ensure the WAS you are using keeps up with the changes. Some WAS products will need to add or modify detections to continue to meet this requirement. If you are using a more full-featured WAS, you may need to modify the scan options from time-to-time as the Top Ten list changes.

There are many commercial and even some free WAS solutions available. Common examples of free or open-source tools are as follows:

- Nessus, free vulnerability scanner, now has some detection of Web-application security issues, see www.nessus.org.
- WebScarab, direct from OWASP Project (see www.owasp.org/index.php/Category:OWASP_WebScarab_Project) is also a must for your assessment efforts. The new one, WebScrab NG, is being created as well (see www.owasp.org/index.php/OWASP_WebScarab_NG_Project).
- w3af, is a Web Application Attack and Audit Framework (see <http://w3af.sourceforge.net/>), which can be used as well.
- Wikto, even if a bit dated, is still useful (see www.sensepost.com/research/wikto/).
- Ratproxy is not a scanner, but a passive discovery and assessment tool (see <http://code.google.com/p/ratproxy/>).
- Another classic passive assessment tool is Paros proxy (<http://sourceforge.net/projects/paros/>).

Commercial tools’ vendors include Qualys, IBM, HP, Whitehat Security, and others. Apart from procuring the above tools and starting to use them on your public Web applications, it is worthwhile to learn a few things about their effective use. We will present those below and highlight their use for card data security.

First, if you are familiar with ASV network scanning (covered below in the section on Requirement 11), you need to know that Web-application security scanning is often more intrusive than network-level vulnerability scanning performed by an ASV for external scan validation. For example, testing for SQL injection or cross-site

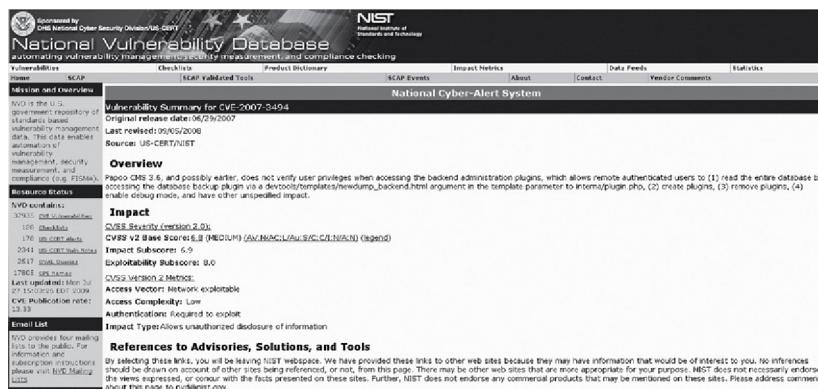


FIGURE 9.3 User Privilege Violation Vulnerability in NVD

scripting often requires actual attempts to perform an injection of SQL code into a database or a script into a Web site. Doing so may cause some databases/applications to hang or to have spurious entries to appear in your Web application.

Just as with network scanning, application scanners may need to perform authentication to get access to more of your application. Many flaws that allow a regular user to become an application administrator can only be discovered via a trusted scan. If the main page on your Web application has a login form, you will need to perform a trusted scan, that is, one that logs in. In addition to finding flaws where a user can become an admin, you also need to ensure that customers cannot intentionally, or inadvertently, traverse within the Web application to see other customers' data. This has happened many times in Web applications. See Figure 9.3, which shows an example of such vulnerability.

Also, depending on how your Web site handles authentication, you may need to log in manually first with the account you will use for scanning, grab the cookie by using a tool like Paros or WebScarab, and then load it into the scanner. Depending on time-outs, you may need to perform this activity just before the scan. In this case, do not plan on being able to schedule scans and have them run automatically.

In addition, WAS requires a more detailed knowledge of software vulnerabilities and attack methodologies to allow for the correct interpretation of results than network-based or “traditional” vulnerability scanning does. Remember to always do research in a laboratory environment, not connected to your corporate environment!

WARNING

It is perfectly reasonable to use an advanced Web-application security scanner to scan applications deployed in production environment, but only after you tried it more than a few times in the laboratory.

When Web farms and Web portals are in the mix, scoping can become somewhat cloudy. Sometimes, you may have a Web portal that will send all transactions involving the transmission or processing of credit card data to different systems. It is likely that the entire cluster will be in-scope for PCI in this case.

Finally, WAS (as an option in Requirement 6.6) is not a substitute for a Web-application penetration test (mandated in Requirement 11.3). Modern Web application scanners can do a lot of poking and probing, but they cannot completely perform tasks performed by a human who is attacking a Web application. For example, full discovery of cross-site request forgery (XSRF or CSRF) flaws is not possible using automated scanners today.

NOTE

Network vulnerability scanning (mandated in Requirement 11.2) and Web-application security testing (as an option in Requirement 6.6) have nothing to do with each other! Please don't confuse them! Network vulnerability scanning is mostly about looking for security issues in operating systems and off-the-shelf applications, such as Microsoft Office or Apache Web server, while Web-application security testing typically looks for security issues in custom and customized Web applications. Simply scanning your Web site with a network vulnerability scanner does not satisfy Requirement 6.6 at all.

Some vendors, like Qualys, do offer both—but they are priced separately (and differently) and should be used based on different operational practices.

Again, just a reminder, whether network or application, the act of scanning is not sufficient as it is because it will only tell you about the issues but will not make you secure; you need to either fix the issue in code or deploy a WAF to block possible exploitation of the issues.

That is what we are going to discuss next.

WARNING

While talking about network or application scanning, you rarely if ever scan to “just know what is out there.” The work does not end when the scan completes; it only begins. The risk reduction from vulnerability scanning comes when you actually remediate the vulnerability or at least mitigate the possible loss. Even if scanning for PCI DSS compliance, your goal is ultimately risk reduction, which only comes when the scan results come back clean.

Web-application firewalls

Let's briefly address the WAFs. Before the discussion of this technology, remember that a network firewall deployed in front of a Web site does not a Web firewall make. Web-application firewalls got their unfortunate name (that of a firewall) from the

hands of marketers. They are more like an intrusion detection system (IDS)/IPS specifically made for Web applications.

These fine folks didn't consider the fact that a network firewall serves to block or allow network traffic based on network protocol and port as well as source and destination, whereas a WAF has to analyze the application behavior before blocking or allowing the interaction of a browser with the Web-application framework.

A WAF, like a network IDS or IPS, needs to be tuned to be effective. To tune it, run reports that give total counts per violation type. Use these reports to tune the WAF. Often, a few messages that you determine to be acceptable traffic for your environment and applications will clean up 80% of the clutter in the alert window. This sounds like an obvious statement, but you would be amazed how many people try to tune WAF technologies in blocking mode while causing the application availability issues in your environment.

If you have a development or quality assurance (QA) environment, placing a WAF (the same type you use in production) in front of one or more of these environments (even in just read/passive mode) can assist you, to some extent, in discovering flaws in Web applications. This then allows for a more planned code fix. Sometimes, you may need to deploy to production with blocking rules until the code can be remediated. In addition, place a WAF in a manner that will block all the directions from where the application attacks might come from (yes, including the dreaded insider attacks).

Finally, unlike the early versions, WAFs are now actually usable and need to be used to protect the Web site from exploitation of the vulnerabilities you discover while scanning.

WHAT TO DO TO BE SECURE AND COMPLIANT?

Requirement 6 asks for more than a few simple things; you might need to invest time to learn about application security before you can bring your organization into compliance.

- Read up on software security (pointers to OWASP, SANS, NIST, MITRE, BSIMM are given above).
- In particular, read up on Web-application security and secure Web-application development
- If you develop software internally or use other custom code, start building your software security program. Such a program must focus on both secure programming to secure the code written within your organization and on code review to secure custom code written by other people for you. No, it is not easy, and likely will take some time.
- Invest in a Web-application security scanner; both free open-source and quality commercial offerings that cover most of OWASP Top 10 are available.
- Also, possibly invest in WAF to block the attacks against the issues discovered while scanning. Tune the firewall and deploy it in blocking model after such tuning.

REQUIREMENT 11 WALK-THROUGH

Let's walk through Requirement 11 to see what is being asked. First, the requirement name itself asks users to "Regularly test security systems and processes," which indicates that the focus of this requirement goes beyond just buffer overflows and format string vulnerabilities from the technical realm, but also includes process weaknesses vulnerabilities. A simple example of a process weakness is using default passwords or easily guessable passwords (such as the infamous "1234" password—or its more modern version "123456"). The above process weaknesses can be checked from the technical side, for example during the network scan by a scanner that can do authenticated policy and configuration audits, such as password strength checks. However, another policy weakness, requiring overly complicated passwords and frequent changes, which in almost all cases lead to users writing the password on the infamous yellow sticky notes, cannot be "scanned for" and will only be revealed during an annual penetration test. Thus, technical controls can be automated, whereas most policy and awareness controls cannot be.

The requirement text goes into a brief description of vulnerabilities in a somewhat illogical manner: "Vulnerabilities are being discovered continually by malicious individuals and researchers, and being introduced by new software." Admittedly, vulnerabilities are being introduced first by software vendors and then discovered by researchers (which are sometimes called "white hats") and attackers ("black hats").

The requirement then calls for frequent testing of software for vulnerabilities: "Systems, processes, and custom software should be tested frequently to ensure security is maintained over time and with any changes in software." An interesting thing to notice in this section is that they explicitly call for testing of systems (such as operating systems software or embedded operating systems), processes (such as the password-management process examples referenced above), and custom software, but don't mention the commercial off-the-shelf software applications. The reason for this is that it is included as part of the definition of "a system" because it is not only the operating system code, but vendor application code contains vulnerabilities. Today, most of the currently exploited vulnerabilities are found in applications and even in desktop applications such as MS Office, Adobe, Java, and at the same time, there is a relative decreased weakness in core Windows system services.

Wireless network testing (Requirement 11.1) states: "use a wireless analyzer at least quarterly to identify all wireless devices in use." Indeed, the retail environments today make heavy use of wireless networks in a few common cases where POS wireless network traffic was compromised by the attackers. Please refer to Chapter 7 for wireless guidance.

Furthermore, Section 11.2 requires you to "run internal and external network vulnerability scans at least quarterly and after any significant change in the network (such as new system component installations, changes in network topology, firewall rule modifications, product upgrades)."

This requirement has an interesting twist, however. Quarterly external vulnerability scans must be performed by a scan vendor qualified by the payment card industry.

Thus, just using any scanner won't do; you need to pick it from the list of ASVs, which we mentioned in Chapter 3. At the same time, the requirements for scans performed after changes are more relaxed: "Scans conducted after network changes may be performed by the company's internal staff." This is not surprising given that such changes occur much more frequently in most networks.

The next section covers the specifics of ASV scanning and the section after covers the internal scanning.

EXTERNAL VULNERABILITY SCANNING WITH ASV

We will look into the operational issues of using an ASV, cover some tips about picking one, and then discuss what to expect from an ASV.

What is an ASV?

As we mentioned in Chapter 3, "Why Is PCI Here?," PCI DSS validation also includes network vulnerability scanning by an ASV. To become an ASV, companies must undergo a process similar to QSA qualification. The difference is that in the case of QSAs, the individual assessors attend classroom training on an annual basis, whereas ASVs submit a scan conducted against a test network perimeter. An organization can choose to become both QSA and ASV, which allows the merchants and service providers to select a single vendor for PCI compliance validation.

ASVs are security companies that help you satisfy one of the two third-party validation requirements in PCI. ASVs go through a rigorous laboratory test process to confirm that their scanning technology is sufficient for PCI validation.

Also, it is worthwhile to mention that validation via an external ASV scan only applies to those merchants that are required to validate requirement 11. In particular, those who don't have to validate Requirement 11 are those that outsource payment processing, those who don't process or store any data on their premises and those with dial-up (non-Internet) terminals. This is important, so it bears repeating; if you have no system to scan because you don't process or store data in-house, you don't have to scan. If this is you, you probably don't have to do too much around PCI DSS. Of course, it goes without saying that deploying a vulnerability management system to reduce your information risk is appropriate even if PCI DSS didn't exist at all.

CONSIDERATIONS WHEN PICKING AN ASV

First, your acquiring bank might have picked an ASV [3] for you. In this case, you might or might not have to use its choice. Note, however, that such prepicked ASV might be neither the best nor the cheapest.

While looking at the whole list of ASVs and then picking the one that "sounds nice" is one way to pick, it is likely not the one that will ensure trouble-free PCI validation and increased card data security as well as reduced risk of data theft. At the time of this writing, the ASV list has grown tremendously, from small one to two persons consulting outlets to IBMs and Verizons of the world, located on all the continents (save Antarctica). How do you pick?

First, one strategy, that needs to be unearthed and explained right away, is as simple as it is harmful for your card data security and PCI DSS compliance status. Namely, organizations that blindly assume that “all ASVs are the same” based on the fact that all are certified by the PCI Council to satisfy PCI DSS scan validation requirements would sometimes just pick on price. This same assumption sometimes applies to QSAs, and as many security industry insiders have pointed out (including both authors), they all are not created equal!

As a result, passing the scan validation requirement and submitting the report that indicates “Pass” will definitely confirm your PCI validation (as long as your ASV remains in good standing with the Council). Sadly, it will not do nearly enough for your card-holder data security. Even if certified, ASVs coverage of vulnerabilities varies greatly; all of them do the mandatory minimum, but more than a few cut corners and stay at that minimum (which, by the way, they are perfectly allowed doing), while others help you uncover other holes and flaws that allow malicious hackers to get to that juicy card data.

Thus, your strategy might follow these steps.

First, realize that all ASVs are not created equal; at the very least, prices for their services will be different, which should give you a hint that the value they provide will also be different.

Second, realize that all ASVs roughly fall into two groups: those that do the minimum necessary according to the above guidance documents (focus on compliance) and those that intelligently interpret the standard and help you with your data security and not just with PCI DSS compliance (focus on security). Typically, the way to tell the two groups apart is to look at the price. In addition, nearly 60% of all currently registered ASVs use the scanning technology from Qualys (www.qualys.com), while many of the rest use Nessus (www.nessus.org) to perform PCI validation.

In addition, though the pricing models for ASV services vary they roughly fall into two groups: in one model, you can scan your systems many times (unlimited scanning) while the other focuses on providing you the mandatory quarterly scan (i.e., four a year). In the latter case, if your initial scan shows the vulnerabilities and need to fix and rescan to arrive at a passing scan, you will be paying extra. Overall, it is extremely unlikely that you can get away with only scanning your network from the outside four times a year.

Third, even though an ASV does not have to be used for internal scanning, it is more logical to pick the same scanning provider for external (must be done by an ASV) and internal (must be done by somebody skilled in using vulnerability management tools). Using the same technology provider will allow you to have the same familiar report format and the same presentation of vulnerability findings. Similarly, and perhaps more importantly, even though ASV scanning does not require authenticated or trusted scanning, picking an ASV that can run authenticated scans on your internal network is useful since such scanning can be used to automate the checking for the presence of other DSS controls, such as password length, account security settings, use of encryption, availability of antimalware defenses, and so on.

Table 9.1 shows a sample list of PCI DSS controls that may be performed using automated scanning tools that perform authenticated or trusted scanning.

Table 9.1 Automatic Validation of PCI DSS Controls

Requirement	PCI DSS 1.2 [1] Requirement	Technical Validation of PCI Requirements
1.4	Install personal firewall software on any mobile and/or employee-owned computers with direct connectivity to the Internet (for example, laptops used by employees), which are used to access the organization's network.	Automated tools are able to check for the presence of personal firewalls deployed on servers, desktops, and laptops remotely.
2.1	Always change vendor-supplied defaults before installing a system on the network, including but not limited to passwords, simple network management protocol (SNMP) community strings, and elimination of unnecessary accounts.	Automated tools can be used to verify that vendor defaults are not used by checking for default and system accounts on servers, desktops, and network devices.
2.1.1	For wireless environments connected to the CDE or transmitting cardholder data, change wireless vendor defaults, including but not limited to default wireless encryption keys, passwords, and SNMP community strings.	Automated tools can be used to verify that default settings and default passwords are not used across wireless devices connected to the wired network.
2.2	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.	Automated tools can validate the compliance of deployed systems to configuration standards, mandated by the PCI DSS.
2.2.2	Enable only necessary and secure services, protocols, daemons, and so on, as required for the function of the system.	Automated tools can help discover systems on the network as well as detect the network-exposed services that are running on systems, and thus significantly reduce the effort needed to bring the environment in compliance.
2.2.4	Remove all unnecessary functionality, such as scripts, drivers, features, subsystems, file systems, and unnecessary Web servers.	Automated tools can help discover some of the insecure and unnecessary functionality exposed to the network, and thus significantly reduce the effort needed to bring the environment in compliance.
2.3	Encrypt all non-console administrative access using strong cryptography. Use technologies such as SSH, VPN, or SSL/TLS for Web-based management and other non-console administrative access.	Automated tools can help validate that encrypted protocols are in use across the systems and that unencrypted communication is not enabled on servers and workstations (SSH, not Telnet; SSL, not unencrypted HTTP, etc.).

Table 9.1 Automatic Validation of PCI DSS Controls (*Cont.*)

Requirement	PCI DSS 1.2 [1] Requirement	Technical Validation of PCI Requirements
3.4	Render PAN unreadable anywhere it is stored (including on portable digital media, backup media, and in logs).	Automated tools can confirm that encryption is in use across the PCI in-scope systems by checking system configuration settings relevant to encryption.
3.5	Protect any keys used to secure cardholder data against disclosure and misuse.	Automated tools can be used to validate security settings relevant to protection of system encryption keys.
4.1	Use strong cryptography and security protocols (e.g., SSL/TLS, IPSEC, SSH, etc.) to safeguard sensitive cardholder data during transmission over open, public networks.	Automated tools can be used to validate the use of strong cryptographic protocols by checking relevant system configuration settings and detect instances of insecure cipher use across the in-scope systems.
4.1.1	Ensure wireless networks transmitting cardholder data or connected to the CDE, use industry best practices (e.g., IEEE 802.11) to implement strong encryption for authentication and transmission.	Automated tools can attempt to detect wireless access points from the network side and to validate the use of proper encryption across those access points.
5.1	Deploy antivirus software on all systems commonly affected by malicious software (particularly, personal computers and servers).	Automated tools can validate whether antivirus software is installed on in-scope systems.
5.2	Ensure that all antivirus mechanisms are current, actively running, and capable of generating audit logs.	Automated tools can be used to check for running status of antivirus tools.
6.1	Ensure that all system components and software are protected from known vulnerabilities by having the latest vendor-supplied security patches installed. Install critical security patches within 1 month of release.	Automated tools can be used to detect missing OS, application patches, and security updates.
6.2	Establish a process to identify and assign a risk ranking to newly discovered security vulnerabilities.	Automated tools are constantly updated with new vulnerability information and can be used in tracking newly discovered vulnerabilities.

(Continued)

Table 9.1 Automatic Validation of PCI DSS Controls (*Cont.*)

Requirement	PCI DSS 1.2 [1] Requirement	Technical Validation of PCI Requirements
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.	Automated tools can be used to assess Web-application security in support of PCI Requirement 6.6.
7.1	Limit access to system components and cardholder data to only those individuals whose job requires such access.	Automated tools can analyze database user right and permissions, looking for broad and insecure permissions.
8.1	Assign all users a unique ID before allowing them to access system components or cardholder data.	In partial support of this requirement, automated tools are used to look for active default, generic accounts (root, system, etc.), which indicate that account sharing takes place.
8.2	In addition to assigning a unique ID, use at least one of the following methods to authenticate all users: Password or passphrase.	Automated tools can be used to look for user accounts with improper authentication settings, such as accounts with no passwords or with blank passwords.
8.4	Render all passwords unreadable during transmission and storage on all system components using strong cryptography.	Automated tools can be used to detect system configuration settings, permitting unencrypted and inadequately encrypted passwords across systems.
8.5	Ensure proper user authentication and password management for non-consumer users and administrators on all system components.	Automated tools can be used to validate an extensive set of user account security settings and password security parameters across systems in support of this PCI requirement.
11.1	Test for the presence of wireless access points and detect unauthorized wireless access points on a quarterly basis.	Automated tools can attempt to detect wireless access points from the network side, thus to help the detection of rogue access points.
11.2	Run internal and external network vulnerability scans at least quarterly and after any significant change in the network (such as new system component installations, changes in network topology, firewall rule modifications, and product upgrades).	Automated tools can be used to scan for vulnerabilities both from inside and from outside the network.

Fourth, look for how ASV workflow matches your experience and expectation. Are there many manual tasks required to perform a vulnerability scan and create a report or is everything automated? Fully automated ASV services where launching a scan and presenting a compliance report to your acquirer can be done from the same interface are available. Still, if you need help with fixing the issues before you can rescan and validate your compliance, hiring an ASV that offers help with remediation is advisable. It goes without saying that picking an ASV that requires you to purchase any hardware or software is not advisable; all external scan requirements can be satisfied by scanning from the Internet.

Finally, even though this strategy focuses on picking an ASV, you and your organization have a role to play as well, namely, in fixing the vulnerabilities that the scan discovered to arrive at a compliant status—clean scan with no failures. We discuss the criteria that ASVs use for pass/fail below.

HOW ASV SCANNING WORKS

ASVs use standard vulnerability scanning technology to detect vulnerabilities that are deemed by the PCI Council to be relevant for PCI DSS compliance. This information will help you understand what exactly you are dealing with when you retain the scanning services of an ASV. It will also help you learn how to pass or fail the PCI scan criteria and how to prepare your environment for an ASV scan.

Specifically, the ASV procedures mandate that ASV covers the following in its scan (the list below is heavily abridged, please refer to “Technical and Operational Requirements for Approved Scanning Vendors (ASVs)” [2] for more details):

- Identify issues in “operating systems, Web servers, Web application servers, common Web scripts, database servers, mail servers, firewalls, routers, wireless access points, common (such as DNS, FTP, SNMP, etc.) services, custom Web applications.”
- “ASV must use the CVSS base score for the severity level” (please see the main site for CVSS at www.first.org/cvss for more information).
- After the above resources are scanned, the following criteria are used to pass/fail the PCI validation (also [2]):
 - “Generally, to be considered compliant, a component must not contain any vulnerability that has been assigned a CVSS base score equal to or higher than 4.0.” Any curious reader can look up a CVSS score for many publicly disclosed vulnerabilities by going to NVD at <http://nvd.nist.gov>.
 - “If a CVSS base score is not available for a given vulnerability identified in the component, then the compliance criteria to be used by the ASV depend on the identified vulnerability leading to a data compromise.” This criterion makes sure that ASV security personnel can use their own internal scoring methodology when CVSS scores cannot be produced.
 - There are additional exceptions to the above rules. Some vulnerability types are included in pass/fail criteria, no matter what their scores are, while others are excluded. Here are the inclusions:
 - “A component must be considered noncompliant if the installed SSL version is limited to Version 2.0, or older.”

- “The presence of application vulnerabilities on a component that may lead to SQL injection attacks and cross-site scripting flaws must result in a noncompliant status” [2].
- The exclusions are as follows:
- “Vulnerabilities or misconfigurations that may lead to DoS should not be taken into consideration” [2].

The above criteria highlight the fact that PCI DSS external scanning is not everything you need to do for security. After all, PCI DSS focuses on card data loss, not on the availability of your key IT resources for your organization and not on their resistance to malicious hackers.

Each ASV will interpret the requirements a little differently. However, quality ASV identifies many different types of vulnerabilities in addition to PCI DSS.

When the scan completes, the report is generated, which can then be used to substantiate your PCI validation via vulnerability scanning.

To summarize, ASV quality scanning will detect all possible external vulnerabilities and highlight those that are reasons for PCI DSS validation failure. The same process needs to be repeated for quarterly scans—usually toward the end of the quarter but not during the last day because remediation activities needs to happen before a final rescan takes place. In fact, let’s talk about operationalizing the ASV scanning.

OPERATIONALIZING ASV SCANNING

To recap PCI DSS Requirement 11.2 calls for quarterly scanning. In addition, every scan may lead to remediation activities, and those aren’t limited to patching. Moreover, validation procedures mention that a QSA will ask for four passing reports during an assessment.

The above calls for an operation process for dealing with this requirement. Let’s build this process together now.

First, it is a very good idea to scan monthly or even weekly if possible. Why would you be doing it to satisfy a quarterly scanning requirement? Well, consider the following scenario: on the last day of the quarter, you perform an external vulnerability scan and you discover a critical vulnerability. The discovered vulnerability is present on 20% of your systems, which totals to 200 systems. Now, you have exactly 1 day to fix the vulnerability on all systems and perform a passing vulnerability scan, which will be retained for your records. Is this realistic? The scenario can happen and, in fact, has happened in many companies that postpone their quarterly vulnerability scan until the very last day and did not perform any ongoing vulnerability scanning. Considering the fact that many acquiring institutions are becoming more stringent with PCI validation requirements, many will not grant you an exception. Beyond the first day over the next month, the scenario will certainly incur unnecessary pain and suffering on your company and your IT staff. What is the way to avoid it? Performing external scans every month or even every week. It is also a good idea to perform an external scan after you apply a patch to external systems.

NOTE

Most companies run their external scans monthly, even though those are called “quarterly scans.” That way, issues can be resolved in time to have a clean quarterly report since there are no surprises. There are known cases where organizations have been burned by waiting until the last month of a quarter to run an external scan. This can cause a serious amount of last-minute, emergency code and system configuration changes, and an overall sense of panic, which is not conducive to good security management.

After you run the scan, carefully review the results of the reports. Are those passing or failing reports? If the report indicates that you do not pass the PCI validation requirement, please note which systems and vulnerabilities do not pass the criteria. Next, distribute the report to those in your IT organization who is responsible for the systems that fail the test. Offer them some guidance on how to fix the vulnerabilities and bring those systems back to PCI compliance. These intermediate reports will absolutely not be shared with your acquiring institutions.

When you receive the indication that those vulnerabilities have been successfully fixed, rescan to obtain a clean report. Repeat the process every month or week.

Finally, scan a final round before the end of the quarter and preserve the reports for the assessor. Thus, your shields will be up at all times. If you are only checking them four times a year, you’re suffering from two problems. First, you are most likely not PCI compliant throughout most of the year. Second, you burden yourself with a massive emergency effort right at the end of the quarter when other people at your organization expect IT systems to operate at its peak. Don’t be the one telling finance that they cannot run that quarterly report!

WHAT DO YOU EXPECT FROM AN ASV?

Discussing the expectations while dealing with an ASV and working toward PCI DSS scan validation is a valuable exercise. The critical considerations are described below.

First, ASV can scan you and present the data (report) to you. It is your job to then bring the environment in compliance. After that, ASV can again be used to validate your compliant status and produce a clean report. Remember, ASV scanning does not make you compliant, *you do*, by making sure that no PCI-fail vulnerabilities are present in your network.

Second, you don’t have to hire expensive consultants just to run an ASV scan for you every quarter. Some ASVs will automatically perform the scan with the correct settings and parameters, without you learning the esoteric nature of a particular vulnerability scanner. In fact, you can sometimes even pay for it online and get the scan right away—yes, you guessed right—using a credit card.

Third, you should expect that a quality ASV will discover more vulnerabilities than is required for PCI DSS compliance. You’d need to make your own judgment

call on whether to fix them. One common case where you might want to address the issue is vulnerabilities that allow hackers to crash your systems (denial of service [DoS] vulnerabilities). Such flaws are out of scope for PCI because they cannot directly cause the theft of card data (this depends on this cause of the DoS); however, by not fixing them, you are allowing the attackers to disrupt your online business operation.

Finally, let us offer some common tips on ASV scanning.

First comes the question: what system must you scan for PCI DSS compliance? The answer to this splits into two parts, for external and internal scanning.

Specifically, for external systems or those visible from the Internet, the guidance from the PCI Council is clear: “all Internet-facing Internet Protocol (IP) addresses and/or ranges, including all network components and devices that are involved in e-commerce transactions or retail transactions that use IP to transmit data over the Internet” (source: “Technical and Operational Requirements for Approved Scanning Vendors (ASVs)” [2] by PCI Council). The obvious answer can be “none” if your business has no connection to the Internet.

For internal systems, the answer covers all systems that are considered in-scope for PCI, which is either those involved with card processing or directly connected to them. The answer can also be “none” if you have no systems inside your perimeter, which are in-scope for PCI DSS.

Second, the question about pass/fail criteria for internal scans often arises as well. While the external ASV scans have clear criteria discussed above, internal scans from within your network have no set pass/fail criteria. The decision is thus based on your idea of risk; there is nothing wrong in using the same criteria as above, of course, if you think that it matches your view of risks to card data in your environment.

Another common question is how to pass the PCI DSS scan validation? Just as above, the answer is very clear for external scans: you satisfy the above criteria. If you don’t, you need to fix the vulnerabilities that are causing you to fail and the rescan. Then, you pass and get a “passing report,” that you can submit to your acquiring bank.

For internal scans, the pass/fail criteria are not expressly written in the PCI Council documentation. Expect your assessor to ask for clean internal and external scans as part of Requirement 11.2. Typically, QSAs will define “clean” internal scans as those not having high-severity vulnerability across the in-scope systems. For a very common scale of vulnerability ranking from 1 to 5 (with 5 being the most severe), vulnerability of severities 3–5 are usually not acceptable. If CVSS scoring is used, 4.0 becomes the cutoff point; vulnerabilities with severities above 4.0 are not accepted unless compensating controls are present and are accepted by the QSA. It was reported that in some situations, a QSA would accept a workable plan that aims to remove these vulnerabilities from the internal environment.

Also, people often ask whether they become “PCI compliant” if they get a passing scan for their external systems. The answer is certainly a “no.” You only satisfied one of the PCI DSS requirements; namely, PCI DSS validation via an external ASV scan. This is not the end. Likely, this is the beginning.

INTERNAL VULNERABILITY SCANNING

As we mentioned in the section, “Vulnerability Management in PCI,” internal vulnerability scanning must be performed every quarter and after every material system change, and no high-severity vulnerability must be present in the final scan preserved for presentation to your QSA. Internal scanning is governed by the PCI Council document called Security Scanning Procedures [4]. Requirement 11.2.1 (since PCI DSS 2.0) states that a merchant must “Perform quarterly internal vulnerability scans.” A QSA must validate and “review the scan reports and verify that the scan process includes rescans until passing results are obtained, or all “High” vulnerabilities as defined in PCI DSS Requirement 6.2 are resolved.” Requirement 11.2.3 also mentioned internal scanning “Perform internal and external scans, and rescans as needed, after any significant change.”

First, using the same template your ASV uses for external scanning is a really good idea, but you can use more reliable trusted or authenticated scanning, which will reveal key application security issues on your in-scope systems that regular, unauthenticated scanning may sometimes miss.

Remediation may take the form of hardening, patching, architecture changes, technology/tool implementations, or a combination thereof. Remediation efforts after internal scanning are prioritized based on risk and can be managed better than external ASV scans. Something to keep in mind for PCI environments is that the remediation of critical vulnerabilities on all in-scope systems is mandatory. This makes all critical and high-severity vulnerabilities found on in-scope PCI systems a high priority. Follow the same process we covered in the “operationalizing ASV scanning” section and work toward removing the high-severity vulnerabilities from the environment before presenting the clean report to the QSA.

Reports that show the finding and remediation of vulnerabilities for in-scope systems over time become artifacts that are needed to satisfy assessment requirements. You should consider that having a place to keep archives of all internal and external scan reports (summary, detailed, and remediation) for a 12-month period is a good idea. Your ASV may offer to keep them for you and also as an added service. However, it is ultimately your responsibility.

This is a continuous process. As with other PCI compliance efforts, it is important to realize that PCI compliance is an effort that takes place 24/7, 365 days a year.

For Internal scanning, you can create different reports for technicians who will fix issues found and summary reports for management. However, overdoing it is bad as well: handing a 10,000-page report to a technician will typically not result in remediation taking place. We are not even talking about a possibility of showing such a report to senior management. Working with the team responsible for remediation to ensure the reports give them actionable data, without overwhelming them, is very much worth the time spent.

Servers that are in-scope are usually scanned off-hours. Be sure your scan windows do not occur during maintenance windows or the target hosts may be off-line for maintenance. If you have workstations in-scope, scans may need to be run during

business hours. For systems that must be scanned during business hours, you may need to make the scans run at a lower intensity.

Until you have thoroughly defined processes (documentation again—many efforts in PCI DSS require both “doing” and “recording”) for all scanning, remediation, and reporting functions tied to your PCI needs, you do not truly “own” the tool you are using.

Finally, issues will undoubtedly occur as you begin your scanning efforts. Here, having a well-defined root cause analysis helps a lot. The sidebar covers how to handle such issues.

Let’s also address the issues of a system change. It is your responsibility to perform a scan after these events have taken place (“Perform internal and external scans, and rescans as needed, after any significant change. Scans must be performed by qualified personnel.”)

Finally, remember that internal scanning is as mandatory for in-scope systems as the ASV scanning is mandatory for external systems.

TOOLS

Here is a sample PCI DSS scan issue tracking process in four steps:

Step 1: Gather inputs from the issue.

Gather Host/Application information: application name, version, patch level, port usage information, and so on.

Was the application disrupted, a system service, or the entire operating system?

What had to be done to recover from the outage? Service restart or host reboot?

Step 2: Verify that the issue was caused by the scan.

Check system logs and try to match the time of the incident to the time of the scan.

Step 3: Place a support call with the application vendor or development team.

Verify that all patches have been applied to the application for “denial of service” and “buffer overflow” problems.

Step 4: If the issue is not resolved by the application vendor, engage support from your scanning vendor.

*Thanks to Derek Milroy for providing the sample process.

PENETRATION TESTING

Requirement 11.3 covers penetration testing requirements. It says: “Implement a methodology for penetration testing” and it is one of the largest changes in PCI DSS 3.0 since it now includes a requirement for penetration testing methodologies. The logic here is again similar: periodic (annual) and after major changes. It appears that in this case, the changes that trigger a penetration test should be of a much larger scale because penetration test services aren’t exactly cheap.

The new 11.3 specifies that a pentest must include the following:

- “Is based on industry-accepted penetration testing approaches (for example, NIST SP800-115)
- Includes coverage for the entire CDE perimeter and critical systems

- Includes testing from both inside and outside the network
- Includes testing to validate any segmentation and scope-reduction controls
- Defines application-layer penetration tests to include, at a minimum, the vulnerabilities listed in Requirement 6.5
- Defines network-layer penetration tests to include components that support network functions as well as operating systems
- Includes review and consideration of threats and vulnerabilities experienced in the last 12 months
- Specifies retention of penetration testing results and remediation activities results.”

By the way, multiple books have been written on the art and science of the penetration test. There is no chance to cover it in our book. However, it makes sense to remind people that a penetration test will always involve a skilled, human attacker, not an automated tool.

Every penetration test begins with one concept—communication. A penetration test should be viewed by a security team as a hostile act—provided they are not asleep at the wheel. After all, the point is to break through active and passive defenses erected around an information system. Communication is important because somebody is about to break your security. During the time of the penetration test, alarm bells should ring, processes would be put into motion, and, if communication has not occurred, and appropriate permissions to perform these tests have not been obtained, law enforcement authorities may be contacted to investigate. Now wouldn’t that be an embarrassment if your PCI-driven penetration test, planned for months, had not been approved by your chief information officer (CIO)?

Moreover, PCI DSS dives deeper into penetration testing details. These penetration tests must include the following:

- 11.3.1: “Perform external penetration testing at least annually and after any significant infrastructure or application upgrade or modification”
- 11.3.2: “Perform internal penetration testing at least annually and after any significant infrastructure or application upgrade or modification”

A test should be repeated until clean (!): “Exploitable vulnerabilities found during penetration testing are corrected and testing is repeated to verify the corrections.” (also new in PCI DSS 3.0)

In addition, they need to verify segmentation (new requirement 11.3.4) that states that “verify that the segmentation methods are operational and effective, and isolate all out-of-scope systems from in-scope systems.” (this checks for very real factors have affected organizations during breaches)

Admittedly, most skilled penetration testing teams will perform such nontechnical testing as well, but not mentioning it explicitly in PCI official documents seems like a minor oversight.

COMMON PCI VULNERABILITY MANAGEMENT MISTAKES

It is worthwhile to point out a few common mistakes that organizations make while working toward satisfying the vulnerability management requirements.

We hinted at the first mistake when we described the password example. It is in focusing only on the technical assessment means (which are indeed easier and more automatic) and omitting the process-based mistakes and issues. In particular, for PCI DSS, it applies to testing only the technology controls but not checking for policy controls such as security awareness, presence of plans and procedures, and so on. Thus, people often focus on the technical vulnerabilities and forget all the human vulnerabilities, such as susceptibility of many enterprise IT users to social engineering, and other lapses of corporate controls. The way to avoid this mistake is to keep in mind that even though you use a scanning vendor, your credit card data might still be pilfered, and addressing the “softer” part of security is just as critical.

Another commonly “lost and forgotten” thing is application-level vulnerabilities, which is not only about open ports and buffer overflows in network-exposed server code. It is also about all the Web applications—from a now-common cross-site scripting and SQL injection flaws to cross-site request forgery to more esoteric flaws in Flash code and other browser-side languages.

Similarly, client-side applications including all the recent Internet Explorer versions (and, frequently, Firefox versions as well), MS Office, and Adobe weaknesses lead to many a government agency falling victim to malicious hackers. What is in common with those “newer” vulnerabilities? Scanning for them is not as easy to automate as finding open Telnet ports and overflows in Internet Information Services (IIS), which was the staple of vulnerability scanning in the late 1990s and early 2000s. PCI requirements refer to such weaknesses but, still, more attention seems to be paid to the network-level stuff exposed to the Internet. The way to avoid this mistake is to keep in mind that a lot of hacking happens on the application layer and to use internal authenticated scanning to look for such issues inside your in-scope network. Such scanning does not have to be performed by the ASV, but if you follow our guidance above, you hopefully picked an ASV that offers internal and authenticated scanning and not just external, mandatory ASV scanning.

Recent Qualys research into Laws of Vulnerabilities [5] shows that attention is not paid to client-side issues. If you limit the scope of analysis to core OS vulnerabilities, the half-life drops to 15 days (which means that people patch those quickly!). On the contrary, if you limit it to Adobe and MS Office flaws, the half-life sharply rises to 60 days (which means people just don’t care—and the current dramatic compromise rampage will continue). The data that supports that situation is shown in [Figures 9.4 and 9.5](#).

Even when application-layer vulnerabilities are not forgotten, and patching and other remediation are happening on an aggressive schedule (nowadays, patching all servers within a “single day” time frame is considered aggressive, that is, what is done by “security leaders”), there is something else to be missed: vulnerability in the applications that were written in-house. Indeed, no vulnerability scanner vendor will have knowledge of your custom-written systems, and even if your penetration-testing

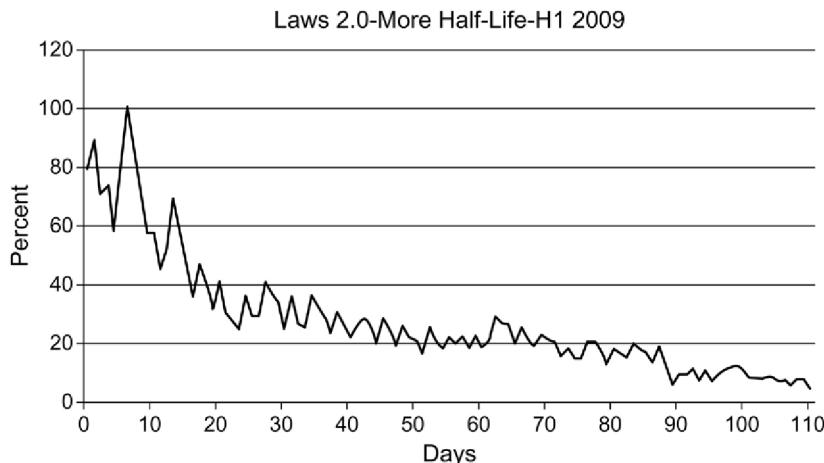


FIGURE 9.4 Half-Life of Core Operating System Vulnerability. Source: Qualys Laws of Vulnerabilities Research

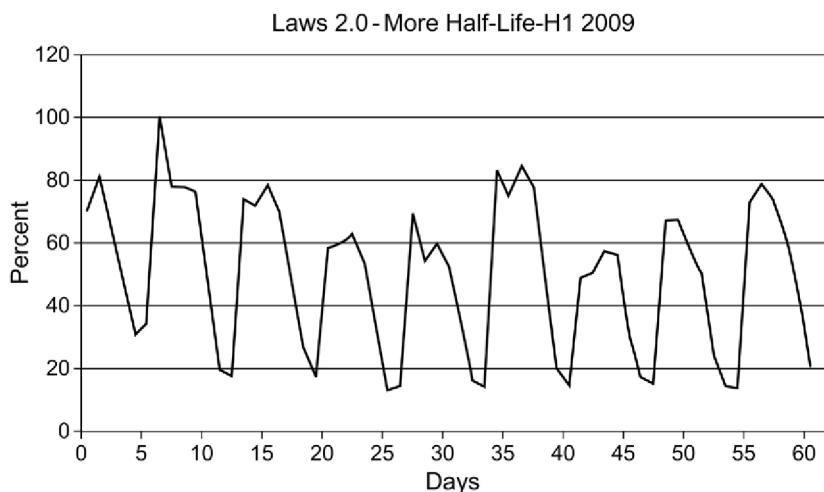


FIGURE 9.5 Half-Life of Client Application Vulnerability. Source: Qualys Laws of Vulnerabilities Research

consultant or an internal “red team” will be able to discover some of them during an annual penetration test, a lot of application code can be written in a year (and thus a lot more vulnerability introduced). The way to avoid this mistake is to train your software engineering staff to use secure programming practices to minimize the occurrence of such flaws, as we discussed in a previous section on Requirement 6 (the detailed coverage of it goes well beyond the scope of this book). While having a good

application tester on staff is unlikely, assessing the security of the homegrown application needs to be undertaken more frequently than once a year. Obviously, initial focus on Web-based and Internet-exposed applications is a must.

The last mistake we mention is misjudging the list of in-scope systems or “scoping errors.” Indeed, modern, large-scale, payment processing systems are complicated and have many dependencies. Avoiding this mistake is not easy: the only way to find all the systems that might need to be scanned and protected is to have your internal staff (who know the systems best) work with an external PCI consultant or QSA (who knows the regulation best) to find out what should be in scope for your particular environment. Primarily, avoid these mistakes by knowing and being able to describe all the business processes that touch the card data. This will take care of the known, authorized locations of card data (which is very important!) and can give you more ideas on scope reduction and reducing card data storage and processing. In addition, even though data discovery technologies are not mandated by PCI, it is advisable to use them to discover other locations of card data, which are not authorized and are not known to be a part of legitimate business process. The latter can be either eliminated or documented and added to PCI DSS scope: these are the only two choices, and “ignored” is not one of them.

Keeping these mistakes in mind has the chance of making your PCI compliance experience a lot less painful.

CASE STUDY

The case studies below illustrate how vulnerability management for PCI is implemented in a few real-world organizations.

PCI AT A RETAIL CHAIN

This case study covers how PCI Requirement 11 was dealt with at a large retail chain in the US Midwest. Nancy’s Natural Foods did not perform any periodic network vulnerability scanning and didn’t use the services of a penetration-testing firm, which put them in a clear violation of PCI DSS rules. Their IT security staff sometimes used the freeware tools to scan a specific system for open ports or sometimes for vulnerabilities, but all such efforts were ad hoc and not tied to any program.

Upon the approach of PCI DSS compliance deadline, the company had to start the scanning using the PCI-ASV every quarter. They chose to deploy a service-based vulnerability scanning from a major vendor. The choice of vendor was determined after a brief proof-of-concept study.

Initially, they suffered from having no information or no knowledge of their vulnerability posture to having too much since they decided to scan all the Internet-facing systems. They reduced the scope to what they considered to be “in-scope” systems, such as those processing payments (few of those systems are ever visible from the internet, however) and those connected to such systems.

Later, their scanning vendor introduced a method to scan the internal systems, which was immediately used by the retailer. However, it turned out that finding the internal systems that are in-scope is even more complicated since many systems have legitimate reasons to connect to those that process credit card transactions. For example, even their internal patch management system was deemed to be in-scope since it frequently connected to the transaction processing servers.

As a result, their route to PCI vulnerability management nirvana took a few months following a phased approach. Implementation followed the following steps:

1. All Internet-facing systems that can be scanned;
2. A smaller set of Internet-facing systems that were deemed to be “in-scope;”
3. A set of internal systems that either process payments or connect to those that do;
4. From there, the company will probably move to scanning select important systems that are not connected to payment processing, but are still critical in its business.

Even though the organization chose not to implement the intrusion detection earlier, their QSA strongly suggested that they look at some options in this area. The company chose to upgrade their firewalls to Unified Threat Management (UTM) devices that combined the capabilities of a firewall and a network IPS. An external consultant suggested their initial intrusion prevention rule set, which the company deployed.

Overall, the project ended up with a successful, if longish, implementation of PCI Requirement 11 using a scanning service as well as UTM devices in place of their firewalls. The organization did pass the PCI assessment, even though they were told to also look at deploying a file integrity monitoring software, which is offered by a few commercial vendors.

PCI AT AN E-COMMERCE SITE

This case study is based on a major e-commerce implementation of a commercial scanning service, a penetration testing by a security consultancy, and a host IPS and file integrity monitoring on critical servers.

Upon encountering PCI compliance requirements, Andie’s Audio has assessed their current security efforts, which include the use of host IPS on their demilitarized zone (DMZ) servers as well as periodic vulnerability scanning. They realized that they needed to additionally satisfy the penetration-testing requirements and file integrity-checking requirements to be truly compliant. Their IT staff performed an extensive research of file integrity monitoring vendors, and chose one with the most advanced centralized management system (to ease the management of all the integrity-checking results). They also contracted a small IT security consultancy to perform the penetration testing for them.

In addition, the team used its previously acquired log-management solution to aggregate the host IPS and file integrity checking, to create a single-data presentation and reporting interface for their PCI assessors. Overall, this project was a successful illustration of a mature security program that needed to only “fill the gaps” to be PCI compliant.

Table 9.2 Vulnerability Management Activities in PCI DSS

Vulnerability-Related Activity Prescribed by PCI DSS	Requirement
Secure coding guidance in regular and Web applications	6
Secure software deployment	6
Code review for vulnerabilities	6
Vulnerability scanning	11
Patching and remediation	6
Technologies that protect from vulnerability exploitation	5, 6, and 11
Site assessment and penetration testing	11

SUMMARY

To conclude, the PCI DSS document covers a lot of activities related to software vulnerabilities. Let us summarize what areas are covered since such requirements are spread over multiple requirements, even belonging to multiple sections. **Table 9.2** covers the vulnerability management activities that we covered in this chapter.

As a result, PCI allows for a fairly comprehensive, if a bit jumbled, look at the entire vulnerability landscape, from coding to remediation and mitigation. Thus, you need to make sure that you look for all vulnerability-related guidance while planning your PCI-driven vulnerability management program. While focusing on vulnerability management, don't reduce it to patch management—do not forget custom applications written in-house or by partners. You need to have an ongoing program to deal with discovered vulnerabilities. Wherever you can, automate the remediation of discovered vulnerabilities and focus on what you cannot. Finally, make sure that you rescan to ensure your reports come back clean.

REFERENCES

- [1] Prioritized Approach for PCI DSS 3.0. <https://www.pcisecuritystandards.org/documents/Prioritized_Approach_for_PCI_DSS_v3_.pdf>.
- [2] Validation Requirements for Approved Scanning Vendors (ASV). PCI Council. <https://www.pcisecuritystandards.org/documents/asv_qualification_requirements_v2.1.pdf>
- [3] ASVProgramGuide,v2.0.PCICouncil.<https://www.pcisecuritystandards.org/documents/ASV_Program_Guide_v2.pdf>.
- [4] Security Scanning Procedures. Version 1.1. PCI Council; 2006.
- [5] Qualys Laws of Vulnerabilities Research. <<http://laws.qualys.com>>; 2009 [accessed 12.07.09].

Logging events and monitoring the cardholder data environment

10

INFORMATION IN THIS CHAPTER:

- PCI requirements covered
- Why logging and monitoring in PCI DSS?
- Logging and monitoring in depth
- PCI relevance of logs
- Logging in PCI requirement 10
- Monitoring Data and log security issues
- Logging and monitoring in PCI—all other requirements
- PCI DSS logging policies and procedures
- Tools for logging in PCI
- Other monitoring tools
- Intrusion detection and prevention
- Integrity monitoring
- Case study

When most people think about information security, the idea of blocking, deflecting, denying, or otherwise stopping a malicious hacker attack comes to mind. Secure network architecture, secure server operating systems, data encryption, and other security technologies are deployed to shield your assets from that evil influence that can steal your information, commit fraud, or disrupt the operation of systems and networks.

Indeed, the visions of tall castle walls, deep moats, or more modern armor and battleships pervade most people's view of warfare as well as information security. However, there is more to warfare (and more to security!) than armor and shields. We are talking about the other keystone of ancient as well as modern (and, likely, future!) warfare: intelligence. Those archers who glance from the top of the castle walls and modern spy satellites that glance down to Earth are no less mandatory to winning (or "not losing," as we have it in the field of information security) the war than fortifications and armored divisions.

In fact, security professionals often organize what they do in security into the following:

- Prevention
- Detection
- Response

“Prevention” is what covers all the blocking, deflecting, denying, or stopping attacks. Notice that it includes the actual blocking of a live attack (such as running a network intrusion prevention system [IPS]) as well as making sure that such an attack cannot take place (such as deploying a patch management system). However, what happens if such prevention measures actually *fail* to prevent or block an attack? Wouldn’t it be nice to know that when it happens?

This is exactly where “detection” comes in. All the logging and monitoring technologies, whether relevant for PCI DSS or not, are things that allow you to *know*. Specifically, know that you are attacked, know that a prevention measure gave way, and know that an attacker has penetrated the network and is about to make it out with the loot. They also allow you to simply know what is going on!

As any self-respecting security guidance, PCI DSS guidance mandates not just prevention but detection in the form of logging, alerting, and monitoring. Let’s discuss all these in detail.

PCI REQUIREMENTS COVERED

Contrary to popular belief, logging and monitoring are not constrained to Requirement 10, but in fact permeate all 12 of the PCI DSS requirements. Still, the key areas where logging and monitoring are mandated in PCI DSS are Requirement 10 and sections of Requirement 11.

WHY LOGGING AND MONITORING IN PCI DSS?

As we mentioned previously, for those who are used to thinking of security as prevention or blocking, the benefits of monitoring might need to be spelled out explicitly, before we go into describing what security monitoring measures are prescribed by PCI, we will do exactly that.

So, why monitor?

First comes situational awareness. It simply means knowing what is going on in your network and on your systems and applications. Examples here are “Who is doing what on that server?”, “Who is accessing my network?,” or “What is that application doing with card data?” In addition, system logging helps you know not only what is going on but also what was going on; a vital component needed for investigations and later incident response.

Next comes new threat discovery, which is simply knowing the bad stuff that is happening. This illustrates one of the major use cases to collect and review logs as well as to deploy intrusion detection systems (IDSs).

Third, logging helps you to get more value out of the network and security infrastructure, deployed for blocking and prevention. For example, using firewall logs for intrusion detection—often justified during assessments but not as commonly used—is an example of that.

What is even more interesting is that logging and monitoring controls (notice that these are considered security “controls” even though they don’t really “control” anything) allow one to measure security and compliance by building metrics and trends. This means that one can use such data for a range of applications, from a simple “Top users by bandwidth” report obtained from firewall logs, all the way to sophisticated tracking of gmail.com card data movements and use.

Last, but not least, if the worst does happen, then you would need to have as much data as possible during your incident response process. You might not use it all, but having reliable logs, assessment trails, and network capture data from all affected systems is indispensable for a hectic postincident environment.

Requirements 10 and 11 are easily capable of inflating PCI compliance costs to the point of consuming the small margins of card transactions. No one wants to lose money to be PCI compliant. Therefore, the ability to meet the requirements above all must make business sense. Nowhere else in PCI compliance does the middle ground of design philosophy come into play more than in the discipline of monitoring, but this is also where minimizing the risk can hurt most.

Finally, assuming that you’ve designed your PCI environment to have appropriate physical and logical boundaries through the use of segregated networks and dedicated application space as we described in Chapter 4 you should be able to identify the boundaries of your monitoring scope. If you haven’t done this part, go back to Requirement 1 and start over!

LOGGING AND MONITORING IN DEPTH

Even today, many organizations, whether under PCI DSS or not, still think of security as blocking and denying and leave monitoring and logging aside, despite their importance.

On the contrary, as computer and Internet technology continues to spread and computers start playing an even more important role in our lives, the records that they produce, such as logs and other traces, start to play a bigger role. From firewalls and routers to databases and enterprise applications, to wireless access points and Voice over Internet Protocol (VoIP) gateways, logs are being spewed forth at an ever-increasing pace. Both security and other IT components not only increase in numbers but also often come with more logging enabled out of the box. An example of this trend includes the Linux operating system as well as Web servers, both commercial and open-source, that now ship with increased levels of logging out of the box. In addition, such additional monitoring methods as full network packet capture, database audit and protection (DAP) tools, and special-purpose application monitoring are becoming common as well. And all this data begs for constant attention!

WARNING

A log management problem is to a large extent a data management problem. Thus, if you deal with logs (and deal with them you must—and not only due to PCI DSS mandates), you'll deal with plenty of data. On top of this, much of this data is often unstructured and requires normalization to get common messages into a common format and sent to a centralized place for analysis.

Still, with logs, it is much better to err on the side of keeping more—in case you'd need to look at it later. This is not the same as saying that you have to keep every log message, but retaining more log data can save you in some situations (e.g., in a legal matter).

This has led some people to proclaim that log analysis is inherently a “big data” problem. Indeed, log data does fit some of the common definitions of big data (such as based on volume, velocity, and variety), but clearly the problem of log overload predates the emergence of the field of big data and big data tools such as Hadoop. In essence, log data was big, before there was big data!

But this is easier said than done. Immense volumes of monitoring data are being generated on payment card processing networks and customer-facing Web resources. In turn, it results in a need to manage, store, and search all this data. Moreover, such review of data needs to happen both reactively—after a suspected incident—and proactively—in search of potential risks and future problems. For example, a typical large retailer generates hundreds of thousands of log messages per day amounting to many terabytes or even petabytes per year. An online merchant can generate millions of various log messages every day. One of America's largest retailers has more than a petabyte (!) of log data on their systems at any given time. Unlike other companies, retailers do not have the option of not managing their logs due to PCI DSS.

NOTE

Even though we often refer to “retailers” as a company subject to PCI DSS, PCI is not only about retailers. Remember, anyone who “stores, processes, or transmits” member-branded credit or debit card numbers must comply with PCI DSS. This applies to hospitals, service providers, restaurants, hotels, and the like.

To start our discussion of PCI logging and monitoring requirements, [Table 10.1](#) shows a sample list of technologies that produce logs of relevance to PCI. Though this list is not comprehensive, it is likely that the readers find at least one system that they have in their cardholder data environment and for which logs are not being collected, much less looked at, and that is not being monitored at all.

Despite the multitude of log sources and types, people typically start from network and firewall logs and then progress upward on the protocol stack as well as sideways toward other nonnetwork applications. For example, just about any firewall or network administrator will look at a simple summary of connections that his or her Cisco ASA or Checkpoint firewall is logging. Many firewalls log in standard syslog format, and such logs are easy to collect and review.

Table 10.1 Log-Producing Technologies, Monitored Using Their Logs

Type	Example Logs
Operating Systems	Linux, Solaris syslog, Windows Event Log
Databases	Oracle, SQL Server assessment trails
Network infrastructure	Cisco routers and switches syslog
Remote access	Virtual private network logs
Network security	Cisco PIX firewalls syslog
Intrusion detection and preventions	Snort network intrusion detection system syslog and packet capture
Enterprise applications	SAP, PeopleSoft logs
Web servers	Apache logs, Internet Information Server logs
Proxy servers	BlueCoat, Squid logs
E-mail servers	Sendmail syslog, various Exchange logs
DNS servers	Bind DNS logs, MS DNS
Antivirus and antispyware	Symantec AV event logs, TrendMicro AV logs
Physical access control	IDenticard, CoreStreet
Wireless networking	Cisco Aironet AP logs

For example, here is a Juniper firewall log message in syslog format:

```
NOC-FWa: NetScreen device_id=NOC-FWa system-notification-
00257(traffic): start_time="2007-05-01 19:17:37" duration=60
policy_id=9 service=snmp proto=17 src zone=noc-services dst zone=-
access-ethernet action=Permit sent=547 rcvd=432 src=10.0.12.10
dst=10.2.16.10 src_port=1184 dst_port=161 src-xlated ip=10.0.12.10
port=1184
```

And, here is one from Cisco ASA firewall device:

```
%ASA-6-106100: access-list outside_access_in denied icmp -
outside/10.88.81.77(0) -> inside/192.10.10.246(11) hit-cnt 1 (first
hit)
```

Finally, one from a Linux IPTables firewall:

```
Nov 12 08:49:50 fw kernel: [3005768.228266] IPT-global R 25 -- ACCEPT
IN=eth3 OUT=eth0 SRC=10.19.10.251 DST=207.232.83.70 LEN=76 TOS=0x00
PREC=0x00 TTL=63 ID=32906 DF PROTO=UDP SPT=34530 DPT=123 LEN=56
```

Reviewing network intrusion detection system (NIDS) or network IPS logs, although “interesting” in case of an incident, is often a frustrating task since NIDS would sometimes produce “false alarms” and dutifully log volumes of them. Still, NIDS log analysis, at least the postmortem kind for investigative purposes, often happens right after firewall logs are looked at. The NIDS logs themselves might be next, checking for signature updates, logins, and changes to the appliance.

Even though system administrators always knew to look at logs in case of problems, large-scale server operating system log analysis (both Windows and Unix/Linux variants) didn’t materialize until more recently. Collecting logs from Windows servers, for example, was hindered by the lack of agentless log collection tools as well as Windows support for log centralization (included in Microsoft server OS since Windows 2008). On the contrary, Unix server log analysis was severely undercut by a total lack of unified format for log content in syslog records.

Web server logs were long analyzed by marketing departments to check on their online campaign successes. Most Web server administrators would also not ignore those logs. However, because Web servers don’t have native log forwarding capabilities (most log to files stored on the server itself), consistent centralized Web log analysis for both security and other IT purposes is still ramping up.

For example, the open-source Apache Web server has several types of logs. The most typical among them are *access_log* that contains all page requests made to the server (with their response codes) and *error_log* that contains various errors and problems. Other Apache logs relate to Secure Sockets Layer (SSL) (*ssl_error_log*) as well as optional granular assessment logs that can be configured using tools such as ModSecurity (which produces an additional highly detailed *audit_log*).

Similarly, e-mail tracking through e-mail server logs languishes in a somewhat similar manner: people only turn to e-mail logs when something goes wrong (e-mail failures) or horribly wrong (external party subpoenas your logs). Lack of native centralization and, to some extent, complicated log formats slowed down the e-mail log analysis initiatives.

Even more than e-mail, database logging wasn’t on the radar of most IT folks until last year. In fact, IT folks were perfectly happy with the fact that even though Relational Database Management Systems had extensive logging and data access assessment capabilities, most of them were never turned on—many times citing performance issues. Oracle, Microsoft Structured Query Language (SQL) Server, IBM DB2, and MySQL all provide excellent logging, if you know how to enable it, configure it for your specific needs, and analyze and leverage the resulting onslaught of data. In the context of PCI DSS, Database Activity Monitoring is often performed not using logs but instead using a separate software tools. Emerging big data tools such as Hadoop also process log data from various components of the framework.

What’s next? Web applications and large enterprise application frameworks largely lived in a world of their own, but now people are starting to realize that their log data provides unique insight into insider attacks, insider data theft, and other trusted access abuse. Many retailers are ramping up their application log management efforts. Additionally, desktop operating system log analysis from large numbers of deployed desktops will also follow.

PCI RELEVANCE OF LOGS

Before we begin with covering additional details on logging and monitoring in PCI, one question needs to be addressed. It often happens that PCI Qualified Security Assessors (QSAs) or security consultants are approached by the merchants: what exactly must they log and monitor for PCI DSS compliance? The honest answer to the above question is that there is no list of what exactly you must be logging due to PCI on each system or, pretty much, any other recent compliance mandate. That is true despite the fact that PCI rules are more specific than most other recent regulations, affecting information security. However, the above does not mean that you can log nothing.

The only thing that can be explained is what you *should* be logging. There is no easy “MUST-log-this” list; it is pretty much up to individual assessor, consultant, vendor, engineer, and so forth to interpret—not simply “read,” but interpret!—PCI DSS guidance in your own environment. In addition, when planning what to log and monitor, it makes sense to start from compliance requirement as opposed to end with what PCI DSS suggests. After all, organization can derive value from using it, even without regulatory or industry compliance.

So, which logs are relevant to your PCI project? In some circumstances, the answer is “all of them,” but it is more likely that logs from systems that handle credit card information, as well as systems they connect to, will be in-scope. Please refer to the data flow diagram that was described in Chapter 4 to determine which systems actually PCI DSS requirements apply to all members, merchants, and service providers that store, process, or transmit cardholder data. Additionally, these requirements apply to all “system components,” which are defined as “any network component, server, or application included in, or connected to, the cardholder data environment.” Network components include, but are not limited to, firewalls, switches, routers, wireless access points, network appliances, and other security appliances. Servers include, but are not limited to, Web, database, authentication, Domain Name System (DNS), e-mail, proxy, and Network Time Protocol (NTP). Applications include all off-the-shelf and custom-built applications, including internally facing and externally facing Web applications.

By the way, it is important to remind you that approaching logging and monitoring with the *sole* purpose of becoming PCI compliant is not only wasteful but can actually undermine the intent of PCI DSS compliance. Starting from its intent—cardholder data security—is the way to go, which we advocate.

If you’ve been at this for a while, you may remember an interpretation clarification that happened many years ago on the actual capture of actions, where must it happen? If the database is logging access, do I need to also log that access in the application? You will need to choose the best option for your particular system, but you only need to capture the action of accessing the data **once**. Access could be captured in the DB (good for batch), application (good for interactive), or system (good for service accounts).

LOGGING IN PCI REQUIREMENT 10

Let’s quickly go through Requirement 10, which directly addresses logging. We will go through it line by line and then go into details, examples, and implementation guidance.

The requirement itself is called “Track and monitor all access to network resources and cardholder data” and is organized under the “Regularly monitor and test networks” heading. The theme thus deals with both periodic (test) and ongoing (monitor) aspects of maintaining your security; we are focusing on logging and monitoring and have addressed periodic testing in Chapter 10. More specifically, it requires a network operator to track and monitor all access to network resources and cardholder data. Thus, both network resources that handle the data and the data itself are subject to those protections.

Furthermore, the requirement states that logging is critical, primarily when “something does go wrong” and one needs to “determine the cause of a compromise” or other problem. Indeed, logs are of immense importance for incident response. However, using logs for routine user tracking and system analysis cannot be underestimated. Next, the requirement is organized in several sections on process, events that need to be logged, suggested level of details, time synchronization, assessment log security, required log review, and log retention policy.

Specifically, Requirement 10.1 covers “establish[ing] a process for linking all access to system components (especially access done with administrative privileges such as root) to each individual user.” This is a very interesting requirement indeed; it doesn’t just mandate for logs to be there or for a logging process to be set, but instead it mentions that logs must be tied to individual persons (not computers or “devices” where they are produced). It is this requirement that often creates problems for PCI implementers because many think of logs as “records of people actions,” while in reality they will only have the “records of computer actions.” Mapping the latter to actual flesh-and-blood users often presents an additional challenge. By the way, PCI DSS Requirement 8.1 mandates that an organization “assigns all users a unique ID before allowing them to access system components or cardholder data, which” helps to make the logs more useful here.

NOTE

Question: Do I have to manually read every single log record daily to satisfy PCI Requirement 10?

Answer: No, automated log analysis and review is acceptable and, in fact, recommended in PCI DSS.

Next, Section 10.2 defines a minimum list of system events to be logged (or, to allow “the events to be reconstructed”). Such requirements are motivated by the need to assess and monitor user actions as well as other events that can affect credit card data (such as system failures).

Following is the list from the requirements (events that must be logged) from PCI DSS:

- 10.2.1: All individual user accesses to cardholder data,
- 10.2.2: All actions taken by any individual with root or administrative privileges,

- 10.2.3: Access to all audit trails,
- 10.2.4: Invalid logical access attempts,
- 10.2.5: Use of and changes to identification and authentication mechanisms—including but not limited to creation of new accounts and elevation of privileges—and all changes, additions, or deletions to accounts with root or administrative privileges
- 10.2.6: Initialization, stopping, or pausing of the audit logs,
- 10.2.7: Creation and deletion of system-level objects.

These requirements cover data access, privileged user actions, log access and initialization, failed and invalid access attempts, authentication and authorization decisions, and system object changes. These lists have their roots in IT governance “best practices,” which prescribe monitoring access, authentication, authorization change management, system availability, and suspicious activity. Thus, other regulations, such as the Sarbanes–Oxley Act, Healthcare Information Portability and Accountability Act (HIPAA) (by means of related NIST guidance) and IT governance frameworks such as COBIT or ISO27001, contain or imply similar lists of events that need to be logged.

NOTE

We hope that in the future, such a list of system events will be determined by the overall log standards that go beyond PCI. There are promising log standard projects such as Common Event Expression (CEE) from MITRE [1] that have a chance to produce such a universally accepted (or at least, industry-accepted) list of events in the medium to long-term future.

Table 10.2 is a practical example of the list on the previous page.

Table 10.2 Logging Requirement and How to Address Them

Requirement Number	Requirement	Example Type of a Log Message
10.2.1	All individual user accesses to cardholder data	Successful logins to processing server (Unix, Windows)
10.2.2	All actions taken by any individual with root or administrative privileges	Sudo root actions on a processing server
10.2.3	Access to all audit trails	Execution of Windows event viewer
10.2.4	Invalid logical access attempts	Failed logins (Unix, Windows)
10.2.5	Use of identification and authentication mechanisms	All successful, failed, and invalid login attempts
10.2.6	Initialization of the audit logs	Windows audit log cleaned alert
10.2.7	Creation and deletion of system-level objects	Unix user added, Windows security policy updated, database created

Moreover, PCI DSS Requirement 10 goes into an even deeper level of detail and covers specific data fields or values that need to be logged for each event. They provide a healthy minimum requirement, which is commonly exceeded by logging mechanisms in various IT platforms.

Such fields are as follows:

- 10.3.1: User identification,
- 10.3.2: Type of event,
- 10.3.3: Date and time,
- 10.3.4: Success or failure indication,
- 10.3.5: Origination of event,
- 10.3.6: Identity or name of affected data, system component, or resource.

As shown, this minimum list contains all the basic attributes needed for incident analysis and for answering the questions: when, who, where, what, and where from. For example, if you are trying to discover who modified a credit card database to copy all the transactions with all the details into a hidden file (a typical insider privilege abuse), you would need-to-know all the above. [Table 10.3](#) summarizes the above fields in this case.

Requirement 10.4 addresses a commonly overlooked but critical requirement: a need to have accurate and consistent time in all of the logs. It seems fairly straightforward that time and security event monitoring would go hand in hand as well.

Table 10.3 PCI Event Details

PCI Requirement	Purpose
10.3.1: User identification	Which user account is associated with the event being logged? This might not necessarily mean “which person,” only which username
10.3.2: Type of event	Was it a system configuration change? File addition? Database configuration change? Explains what exactly happened
10.3.3: Date and time	When did it happen? This information helps in tying the event to an actual person
10.3.4: Success or failure indication	Did he or she try to do something else that failed before his or her success in changing the configuration?
10.3.5: Origination of event	Where did he or she connect from? Was it a local access or network access? This also helps in tying the log event to a person. Note that this can also refer to the process or application that originated the event
10.3.6: Identity or name of affected data, system component, or resource	What is the name of the database, system object, and so forth which was affected? Which server did it happen on? This provides important additional information about the event

System time is frequently found to be arbitrary in a home or small office network. It's whatever time your server was set at, or if you designed your network for some level of reliance, your systems are configured to obtain time synchronization from a reliable source, like the NTP servers.

A need to "Using time-synchronization technology, synchronize all critical system clocks and times and ensure that the following is implemented for acquiring, distributing, and storing time" (clarified in PCI DSS 3.0) can make or break your security incident response or lead to countless hours spent figuring out the actual times of events by correlating multiple sources of information together. In some cases, uncertainty about the log timestamps might even lead a court case to be dismissed because uncertainty about timestamps might lead to uncertainty in other claims as well. For example, from "so you are saying you are not sure when exactly it happened?" an expert attorney might jump to "so maybe you are not even sure what happened?" Fortunately, this requirement is relatively straightforward to address by configuring an NTP environment and then configuring all servers to synchronize time with it. The primary NTP servers can synchronize time with *time.nist.gov* or other official time sources, also called "Stratum 1" sources. For the most part, you can use any number of official time sources, including going down to Stratum 2 or 3 type devices. The goal is to have one time for your organization that everyone can synchronize with for services. Otherwise, you will end up with outages in directory systems, some authentication systems, and performing event discovery or analytics will become a nightmare.

Security of the logs themselves is of paramount importance for reasons similar to the above concerns about the log time synchronization. Requirement 10.5 states that one needs to "secure audit trails so they cannot be altered" and then clarifies various risks that need to be addressed.

Because it is a key issue, we will look at the security of monitoring data and logs in the next section.

MONITORING DATA AND LOG FOR SECURITY ISSUES

Although PCI is more about being compliant and protecting the data than about being hacked (well, not directly at least!), logs certainly help to answer this question. However, if logs themselves are compromised by the attackers and the log server is broken into, they lose all value for either security or compliance purposes. Thus, having assured log confidentiality, integrity, and availability (CIA) is a requirement for PCI as well as a best practice for other log uses.

First, one needs to address all the CIA of logs. Section 10.5.1 of PCI DSS covers the confidentiality: "Limit viewing of audit trails to those with a job-related need." This means that only those who need to see the logs to accomplish their jobs should be able to. What is so sensitive about logs? One of the obvious answers is that authentication-related logs will always contain usernames. Although not truly secret, username information provides 50% of the information needed for password guessing (password

being the other 50%). Why give possible attackers (whether internal or external) this information? Moreover, because of users mistyping their credentials, it is not uncommon for passwords themselves to show up in logs. Poorly written Web applications might result in a password being logged together with the Web Uniform Resource Locator in Web server logs. Similarly, a Unix server log might contain a user password if the user accidentally presses “Enter” one extra time while logging in.

Next comes “integrity.” As per Section 10.5.2 of PCI DSS, one needs to “protect audit trail files from unauthorized modifications.” This one is blatantly obvious; because if logs can be modified by unauthorized parties (or by anybody, in fact), they stop being an objective assessment trail of system and user activities.

However, one needs to preserve the logs not only from malicious users but also from system failures and consequences of system configuration errors. This touches upon both the “availability” and “integrity” of log data. Specifically, Section 10.5.3 of PCI DSS covers that one needs to “promptly back-up audit trail files to a centralized log server or media that is difficult to alter.” Indeed, centralizing logs to a server or a set of servers that can be used for log analysis is essential for both log protection and increasing log usefulness. Backing up logs to DVDs (or tapes, for that matter) is another action you might have to perform as a result of this requirement. You should always keep in mind that logs on tape are not easily accessible and not searchable in case of an incident.

Many pieces of network infrastructure such as routers and switches are designed to log to an external server and only preserve a minimum (or none) of logs on the device itself. Thus, for those systems, centralizing logs is most critical.

To further decrease the risk of log alteration as well as to enable proof that such alteration didn’t take place, Requirement 10.5.5 calls for the “use file integrity monitoring and change detection software on logs to ensure that existing log data cannot be changed without generating alerts.” At the same time, adding new log data to a log file should not generate an alert because log files tend to grow and not shrink on their own (unless logs are rotated or archived to external storage). File integrity monitoring systems use cryptographic hashing algorithms to compare files to a known good copy. The issue with logs is that log files tend to grow due to new record addition, thus undermining the utility of integrity checking. To resolve this difficulty, note that integrity monitoring can only assure the integrity of logs that are not being actively written to by the logging components. However, there are solutions that can verify the integrity of growing logs.

The next requirement is one of the most important as well as one of the most overlooked. Many PCI implementers simply forget that PCI Requirement 10 does not just call for “having logs” but for “having the logs and looking at them.” Specifically, Section 10.6 states that the PCI organization must, as per PCI DSS, “Review logs and security events for all system components to identify anomalies or suspicious activity.”

Now in PCI DSS 3.0, 10.6.1 clarifies that some logs need to be reviewed daily while others can be analyzed less frequently, based on risk assessment. Specifically, “All security events, Logs of all system components that store, process, or transmit CHD and/or SAD, or that could impact the security of CHD and/or SAD, Logs of

all critical system components and Logs of all servers and system components that perform security functions” must be reviewed daily.

Thus, the requirement covers the scope of log sources that need to be “reviewed daily” and not just configured to log and have logs preserved or centralized. Given that a Fortune 1000 IT environment might produce gigabytes of logs per day, it is humanly impossible to read all of the logs. That is why a note is added to this PCI DSS requirement that states “Log harvesting, parsing, and alerting tools may be used to achieve compliance with Requirement 10.6.” Indeed, log management tools are the only practical way to satisfy this requirement.

The Requirement 10.7 deals with another hugely important logging question – log retention. It says to “retain audit trail history for at least one year, with a minimum of three months online availability.” Unlike countless other requirements, this deals with the complicated log retention question directly. Thus, if you are not able to go back one year and look at the logs, you are in violation.

Finally, PCI DSS 3.0 focus on operational practices added requirement 10.8: “Ensure that security policies and operational procedures for monitoring all access to network resources and cardholder data are documented, in use, and known to all affected parties” This simply stresses the point that the organization must do things as part of their normal operation, as well as document was being done. QSAs are encouraged to verify that procedures are documented and in fact implemented by skilled personnel.

So, let us summarize what we have learned so far on logging in PCI:

- PCI Requirement 10 calls for logging specific events with a predefined level of details from all in-scope systems.
- PCI calls for tying the actual users to all logged actions.
- All clocks and time on the in-scope systems should be synchronized.
- The CIA of all collected logs should be protected.
- Logs should be regularly reviewed; specific logs should be reviewed at least daily.
- All in-scope logs should be retained for at least 1 year.
- Organizations must have documented and actually implemented procedures and processes for log monitoring.

Now, we are ready to dig deeper to discover that logs and monitoring “live” not only within Requirement 10 but in all other PCI requirements.

LOGGING AND MONITORING IN PCI—ALL OTHER REQUIREMENTS

Although many people think that logs in PCI are represented only by Requirement 10, the reality is more complicated: logs are in fact present, undercover, in all other sections. We will now reveal where they hide. [Table 10.4](#) highlights some of the places where logging requirements are implied or mentioned. The overall theme here is that logging and log management assists with validation and verification of many other requirements.

Table 10.4 Logging and Monitoring across PCI DSS Requirements

	Domain	Requirement	Logging Relevance and Recommendations
1	Build and maintain a secure network	Install and maintain a firewall configuration to protect cardholder data	Enable firewall logging, review logs for access violations, use of risky protocols, device configuration changes, accesses to critical network segments
2	Build and maintain a secure network	Do not use vendor-supplied defaults for system passwords and other security parameters	Review logs to look for insecure services, additional services starting on servers, as well as password changes upon server deployment
3	Protect cardholder data	Protect stored cardholder data	Review the logs related to key management to verify that the requirements (such as key changes) are being followed
4	Protect cardholder data	Encrypt transmission of cardholder data across open, public networks	Look at firewall, virtual private network logs to verify that only secure network communication is used
5	Maintain a vulnerability management program	Use and regularly update antivirus software	Verify that antivirus software is updated by looking at antivirus logs; also look for detection and mitigation failures that might indicate that malware is present on the network
6	Maintain a vulnerability management program	Develop and maintain secure systems and applications	Make sure that custom applications written or customized for your environment also provide logging. Watch logs of system update and software distribution servers to make sure that patches are being deployed when needed on all relevant servers
7	Implement strong access control measures	Restrict access to cardholder data by business need-to-know	Verify that such access is indeed limited by reviewing the access logs
8	Implement strong access control measures	Assign a unique ID to each person with computer access	Perform log correlation to detect ID sharing in violation of this requirement; review logs indicating changes to users' privileges; verify password changes based on authentication systems logs, and so forth. Look for administrator and root accounts that can sometimes be shared (and are rarely removed from systems)

Table 10.4 Logging and Monitoring across PCI DSS Requirements (Cont.)

	Domain	Requirement	Logging Relevance and Recommendations
9	Implement strong access control measures	Restrict physical access to cardholder data	Collect, analyze, and review physical access control system logs
10	Regularly monitor and test networks	Track and monitor all access to network resources and cardholder data	Covered above; this is the main logging and monitoring requirement
11	Regularly monitor and test networks	Regularly test security systems and processes	Monitor intrusion detection/prevention systems and file integrity checking
12	Maintain an information security policy	Maintain a policy that addresses information security	Make sure that logging and monitoring are represented in your security policy as well as operational standards, procedures, and management reports

Now, let's dive deeper into the role of logs to further explain that logs are not only about the Requirement 10. Just about every claim that is made to satisfy the requirements, such as data encryption or antivirus updates, can make effective use of log files to actually substantiate it.

For example, Requirement 1, “Install and maintain a firewall configuration to protect cardholder data,” mentions that organizations must have “a formal process for approving and testing all external network connections and changes to the firewall configuration.” However, after such a process is established, you need to validate that firewall configuration changes do happen with authorization and in accordance with documented change management procedures. That is where logging becomes extremely handy, because it shows you what actually happened and not just what was supposed to happen.

Specifically, seeing a message such as this Cisco ASA appliance record should indicate that someone is likely trying to modify the appliance configuration.

```
%ASA-5-502103: User priv level changed: Uname: jsmith From:  
privilege_level1 To: privilege_level2
```

Other log-related areas within Requirement 1 include Section 1.1.6.

“Documentation and business justification for use of all services, protocols, and ports allowed, including documentation of security features implemented for those protocols considered to be insecure” where logs should be used to watch for all events triggered due to such communication.

The entire Requirement 1.3 contains guidance to firewall configuration, with specific statements about inbound and outbound connectivity. One must use firewall

logs to verify this; even a review of configuration would not be sufficient, because only logs show “how it really happened” and not just “how it was configured.”

Similarly, Requirement 2 talks about password management “best practices” as well as general security hardening, such as not running unneeded services. Logs can show when such previously disabled services are being started, either by misinformed system administrators or by attackers.

For example, if Apache Web server is disabled on an e-mail server system, a message such as the following should trigger an alert because the service should not be starting or restarting.

```
[Sun Jul 18 04:02:09 2004] [notice] Apache/1.3.19 (Unix) (Red-Hat/Linux) mod_ssl/2.8.1 OpenSSL/0.9.6 DAV/1.0.2 PHP/4.0.4pl1 mod_perl/1.24_01 configured-- resuming normal operations
```

Furthermore, Requirement 3, which deals with data encryption, has unambiguous links to logging. Specifically, key generation, distribution, and revocation are logged by most encryption systems, and such logs are critical for satisfying this requirement. Requirement 4, which also deals with encryption, has logging implications for similar reasons.

Requirement 5 refers to antivirus defenses. Of course, to satisfy Section 5.2, which requires that you “Ensure that all antivirus mechanisms are current, actively running, and capable of generating audit logs,” one needs to see such mentioned logs.

For example, Symantec AntiVirus might produce the following log record that occurs when the antivirus software experiences problems and cannot continue scanning, thus putting you in violation of PCI DSS rules.

```
Product: Symantec AntiVirus--Error 1706. AntiVirus cannot continue.
```

So, even the requirement to “use and regularly update antivirus software” will likely generate requests for log data during the assessment, because the information is present in antivirus assessment logs. It is also well-known that failed antivirus updates, also reflected in logs, expose the company to malware risks because antivirus without the latest signature updates only creates a false sense of security and undermines the compliance effort.

Requirement 6 is in the same league: it calls for the organizations to “Develop and maintain secure systems and applications,” which is unthinkable without a strong logging functions, logs useful for security analysis and incident response as well as application security monitoring.

Requirement 7, which states that one needs to “Restrict access to cardholder data by business need-to-know,” requires logs to validate who actually had access to said data. If the users who should be prevented from seeing the data appear in the log files as accessing the data usefully, remediation is needed.

Assigning an unique ID to each user accessing the system fits with other security good practices. In PCI, it is not just good practice; it is a requirement (Requirement 8 “Assign a unique ID to each person with computer access”).

Obviously, one needs to “Control addition, deletion, and modification of user IDs, credentials, and other identifier objects” (Section 8.1.2 of PCI DSS 3.0). Most systems log such activities.

For example, the message below indicates a new user being added to a PIX/ASA firewall.

```
%ASA-5-502101: New user added to local dbase: Uname: anilt Priv: 1  
Encpass: 56Rt8U
```

In addition, Section 8.2.4, “Change user passwords/passphrases at least every 90 days” can also be verified by reviewing the logs files from the server to assure that all the accounts have their password changed at least every 90 days.

Requirement 9 presents a new realm of security—physical access control. Even Section 9.4 that covers maintaining a visitor log (likely in the form of a physical log book) is connected to log management if such a visitor log is electronic. There are separate data retention requirements for such logs: “A visitor log is used to maintain a physical audit trail of visitor activity to the facility as well as computer rooms and data centers where cardholder data is stored or transmitted. Document the visitor’s name, the firm represented, and the onsite personnel authorizing physical access on the log. Retain this log for a minimum of three months, unless otherwise restricted by law” (Requirement 9.4.4 of PCI DSS 3.0)

Requirement 11 addresses the need to scan the in-scope systems for vulnerabilities. However, it also calls for the use of IDS or IPS in Section 11.4: “Use intrusion-detection and/or intrusion-prevention techniques to detect and/or prevent intrusions into the network. Monitor all traffic at the perimeter of the cardholder data environment as well as at critical points in the cardholder data environment, and alert personnel to suspected compromises.” Intrusion detection is only useful if monitored!

Requirement 12 covers the issues on a higher level—security policy as well as security standards and daily operational procedures (e.g., a procedure for daily log review mandates by Requirement 10 should be reflected here). However, it also has logging implications because assessment logging should be a part of every security policy. In addition, incident response requirements are also tied to logging: “Establish, document, and distribute security incident response and escalation procedures to ensure timely and effective handling of all situations” is unthinkable to satisfy without effective collection and timely review of log data.

Thus, event logging and security monitoring in PCI DSS program goes much beyond Requirement 10.

PCI DSS LOGGING POLICIES AND PROCEDURES

At this stage, we went through all of the PCI guidelines and uncovered where logging and monitoring are referenced. We now have a mammoth task ahead—how to address all those requirements?

In light of the above discussion, a PCI-derived logging policy must at least contain the following:

- Adequate logging, that covers both logged event types and details;
- Log aggregation and retention (1 year);
- Log protection;
- Log review.

Let's start focusing in-depth on log review as the most complex of the requirements. PCI testing and validation procedures for log review mandate that a QSA should "obtain and examine security policies and procedures to verify that they include procedures to review security logs at least daily and that follow-up to exceptions is required." QSA must also assure "Through observation and interviews, verify that regular log reviews are performed for all system components."

Thus the organization should at least address the following:

1. Log review practices, procedures and tasks;
2. Exception investigation and analysis.

The procedures can be implemented using automated log management tools as well as manually when tools are not available.

The overall connection between the three types of PCI-mandates procedure is as follows ([Figure 10.1](#)).

In other words, "Periodic Log Review Practices" are performed every day (or less frequently, if daily review is impossible) and any discovered exceptions or are escalated to "Exception Investigation and Analysis."

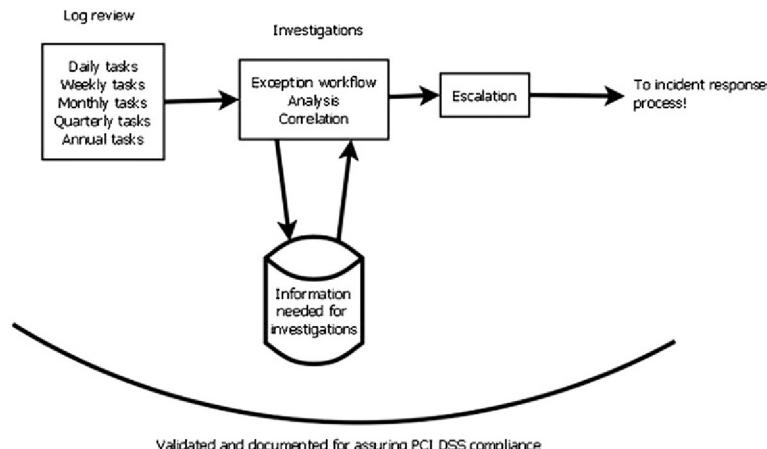


FIGURE 10.1 PCI Log Flow

The basic principle of PCI DSS periodic log review (further referred to as “daily log review” even if it might not be performed daily for all the applications) is to accomplish the following:

- Assure that card holder data has not been compromised by the attackers;
- Detect possible risks to cardholder data, as early as possible;
- Satisfy the explicit PCI DSS requirement for log review;

Even given that fact that PCI DSS is the motivation for daily log review, other goals are accomplished by performing daily log review:

- Assure that systems that process cardholder data are operating securely and efficiently;
- Reconcile all possible anomalies observed in logs with other systems activities (such as application code changes or patch deployments).

In light of the above goals, the daily log review is built around the concept of “baselining” or learning and documenting normal set of messages appearing in logs. Baseling is then followed by the process of finding “exceptions” from the normal routine and investigating them to assure that no breach of cardholder data has occurred or imminent.

The process can be visualized as follows (Figure 10.2).

Before PCI daily log review is put into practice, it is critical to become familiar with normal activities logged on each of the applications.

Explicit event types might not always be available for some log types. For example, some Java application logs and some Unix logs don’t have explicit log or event types recorded in logs. What is needed is to create an implicit event type. The procedure for this case is as follows:

1. Review the log message,
2. Identify which part of the log message identifies what it is about,

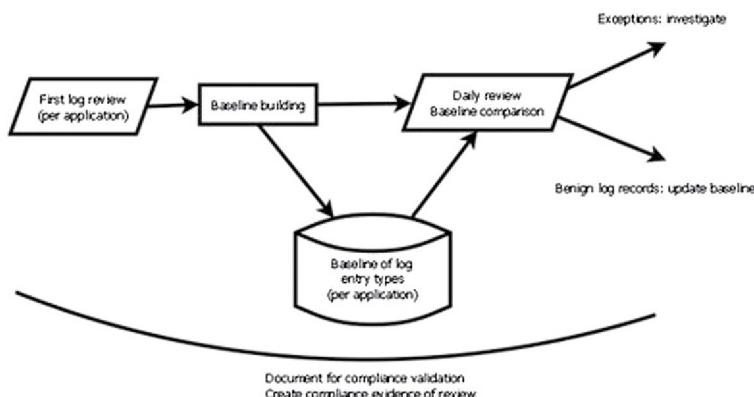


FIGURE 10.2 Finding Exceptions

3. Determine whether this part of the message is unique,
4. Create an event ID from this part of the message.

Even though log management tools perform the process automatically, it makes sense to go through an example of doing it manually in case manual log review procedure is utilized.

For example:

1. Review the log message

The log message is:

[Mon Jan 26 22:55:37 2011] [notice] Digest: generating secret for digest authentication.

2. Identify which part of the log message identifies what it is about

It is very likely that the key part of the message is “generating secret for digest authentication” or even “generating secret.”

3. Determine whether this part of the message is unique

A review of other messages in the log indicates that no other messages contain the same phrase and thus this phrase can be used to classify a message as a particular type.

4. Create an event ID from this part of the message

We can create a message ID or message type as “generating_secret.” Now we can update our baseline that this type of message was observed today.

BUILDING AN INITIAL BASELINE MANUALLY

To build a baseline without using a log management tool has to be done when logs are not compatible with an available tool or the available tool has poor understanding of log data. To do it, perform the following:

1. Make sure that relevant logs from a PCI application are saved in one location,
2. Select a time period for an initial baseline: “90 days” or “all time” if logs have been collected for less than 90 days; check the timestamp on the earliest logs to determine that,
3. Review log entries starting from the oldest to the newest, attempting to identify their types,
4. Manually create a summary of all observed types; if realistic, collect the counts of time each message was seen (not likely in case of high log data volume),
5. Assuming that no breaches of card data have been discovered in that time period, we can accept the above report as a baseline for “routine operation”
6. An additional step should be performed while creating a baseline: even though we assume that no compromise of card data has taken place, there is a chance that some of the log messages recorded over the 90-day period triggered some kind of action or remediation. Such messages are referred to as “known bad” and should be marked as such.

The logs could be very large text and .csv files. If you are doing your analysis notepad ++ and csved are two tools that can help. They also have good search features and counting functions within them.

GUIDANCE FOR IDENTIFYING “KNOWN BAD” MESSAGES

The following are some rough guidelines for marking some messages as “known bad” during the process of creating the baseline. If generated, these messages will be looked at first during the daily review process.

1. Login and other “access granted” log messages occurring at unusual hour,
2. Credential and access modifications log messages occurring outside of a change window,
3. Any log messages produced by the expired user accounts,
4. Reboot/restart messages outside of maintenance window (if defined),
5. Backup/export of data outside of backup windows (if defined),
6. Log data deletion,
7. Logging termination on system or application,
8. Any change to logging configuration on the system or application,
9. Any log message that has triggered any action in the past: system configuration, investigation, and so on.
10. Other logs clearly associated with security policy violations.

As we can see, this list is also very useful for creating “what to monitor in near-real-time?” policy and not just for logging. Over time, this list should be expanded based on the knowledge of local application logs and past investigations.

After we built the initial baselines we can start the daily log review.

Main workflow: daily log review

This is the very central piece of the log review—comparing the logs produced over the last day (in case of a daily review) with an accumulated baseline.

Daily workflow follows this model ([Figure 10.3](#)).

This diagram summarizes the actions of the log analyst who performs the daily log review.

EXCEPTION INVESTIGATION AND ANALYSIS

A message not fitting the profile of a normal is flagged “an exception.” It is important to note that an exception is not the same as a security incident, but it might be an early indication that one is taking place.

At this stage we have an individual log message that is outside of routine/normal operation. How do we figure out whether it is significant, determine impact on security and PCI compliance status?

The following high-level investigative process (“Initial Investigation”) is used on each “exception” entry (more details are added further in the document) ([Figure 10.4](#)).

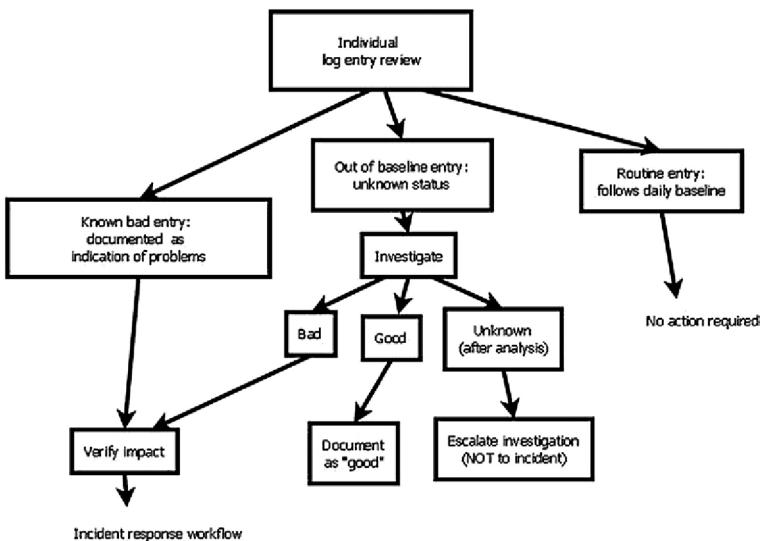


FIGURE 10.3 PCI DSS Daily Workflows

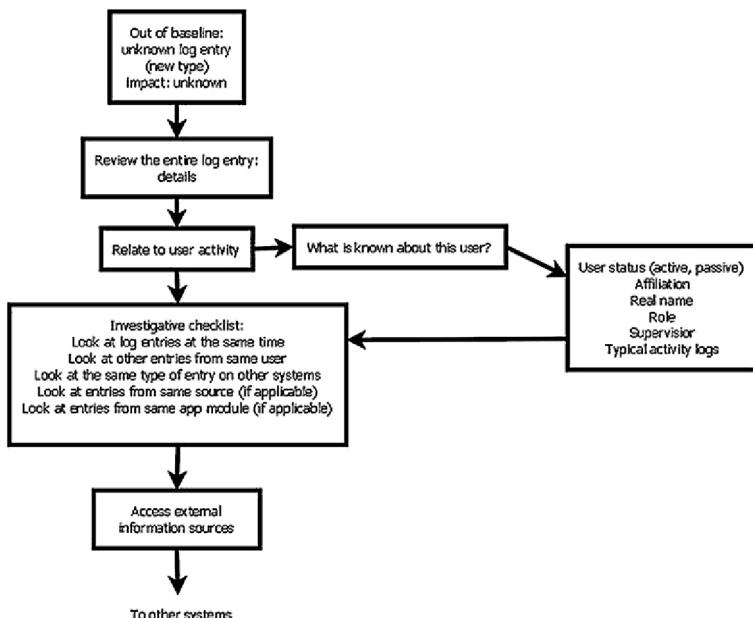


FIGURE 10.4 High Level Investigative Process

Specifically, the above process makes use of a log investigative checklist, which is explained below in more details.

1. **Look at log entries at the same time:** this technique involves looking at an increasing range of time periods around the log message that is being investigated. Most log management products can allow you to review logs or to search for all logs within a specific time frame. For example:
 - a. First, look at other log messages triggered 1 min before and 1 min after the “suspicious” log
 - b. Second, look at other log messages triggered 10 min before and 10 min after the “suspicious” log
 - c. Third, look at other log messages triggered 1 h before and 1 h after the “suspicious” log
2. **Look at other entries from same user:** this technique includes looking for other log entries produced by the activities of the same user. It often happens that a particular logged event of a user activity can only be interpreted in the context of other activities of the same user. Most log management products can allow you to “drill down into” or search for a specific user within a specific time frame.
3. **Look at the same type of entry on other systems:** this method covers looking for other log messages of the same type, but on different systems in order to determine its impact. Learning when the same message was products on other system may hold clues to understanding the impact of this log message.
4. **Look at entries from same source (if applicable):** this method involves reviewing all other log messages from the network source address (where relevant).
5. **Look at entries from same app module (if applicable):** this method involves reviewing all other log messages from the same application module or components. While other messages in the same time frame (see item 1. above) may be significant, reviewing all recent logs from the same components typically helps to reveal what is going on.

VALIDATION OF LOG REVIEW

Final and critical part of compliance-motivated log review is making sure that there is sufficient evidence of the process, its real-world implementation and diligence in following the process. The good news here is that the same data can be used for management reporting about the logging and log review processes.

Let's determine what documentation should be produced as proof of log review.

First, the common misconception is that having the logs actually provides that. That is not really true: “having logs” and “having logs reviewed” are completely different and sometime years of maturing the security and compliance program separates one and the other.

Just to remind you, we have several major pieces that we need to prove for PCI DSS compliance validation. Here is the master-list of all compliance proof we will

assemble. Unlike other sections, here we will cover proof of logging and not just proof of log review since the latter is so dependent on the former:

- Presence and adequacy of logging,
- Presence of log review processes and its implementation,
- Exception handling process and its implementation.

PCI COMPLIANCE EVIDENCE PACKAGE

Overall, it is useful to create a “PCI Compliance Evidence Package” to show it to the QSA that will establish our three keys of PCI DSS logging requirements:

- Presence and adequacy of logging,
- Presence of log review processes and its implementation,
- Exception handling process and its implementation.

While it is possible to prepare the evidence package before the assessment, it is much easier to maintain it on the ongoing basis and build scripts or other automated processes to refresh evidentiary data used during your assessment. For example, keep printed or electronic copies of the following:

1. Logging policy that covers all of the PCI DSS in-scope systems,
2. Logging and log review procedures (this document),
3. List of log sources—all systems and their components (applications) from the in-scope environment,
4. Sampling of configuration files that indicate that logging is configured according to the policy (e.g., /etc/syslog.conf for Unix, screenshots of audit policy for Windows, etc.),
5. Sampling of logs from in-scope systems that indicate that logs are being generated according to the policy and satisfy PCI DSS logging requirements,
6. Exported or printed report from a log management tools that shows that log reviews are taking place,
7. Up-to-date logbook defined above.

This will allow you to establish a compliant status and prove ongoing compliance.

Finally, let's summarize all the operational tasks the organization should be executing in connection with log review.

PERIODIC OPERATIONAL TASK SUMMARY

The following contains a summary of operational tasks related to logging and log review. Some of the tasks are described in detail in the document above; others are auxiliary tasks needed for successful implementation of PCI DSS log review program.

DAILY TASKS

Table 10.5 contains daily tasks, responsible role that performs them as well as what record or evidence is created of their execution:

Table 10.5 PCI DSS Daily Tasks

Task	Responsible Role	Evidence
Review all the types of logs produced over the last day as described in the daily log review procedures	Security administrator, security analyst, (if authorized) application administrator	Record of reports being run on a log management tool
(As needed) investigate the anomalous log entries as described in the investigative procedures	Security administrator, security analyst, (if authorized) application administrator	Recorded logbook entries for investigated events
(As needed) take actions as needed to mitigate, remediate or reconcile the results of the investigations	Security administrator, security analyst, (if authorized) application administrator, other parties	Recorded logbook entries for investigated events and taken actions
Verify that logging is taking place across all in-scope applications	Application administrator	Create a spreadsheet to record such activities for future assessment
(As needed) enabled logging if disabled or stopped	Application administrator	Create a spreadsheet to record such activities for future assessment

Now we are ready to discuss the tools you might use for logging while meeting PCI DSS compliance.

TOOLS FOR LOGGING IN PCI

First, if we are talking about a single server and a single piece of network gear such as a router there might be no need for automation and tools. One can easily configure logging and then look at the logs (measuring a few pages of text a day or more), as well as save a copy of said logs to make sure that one can go back a year and review the old logs if needed. However, this approach fails miserably when the number of systems grows from 1 to, say, 10. In reality, a large e-commerce site or a whole chain of stores might easily have thousands of in-scope systems, starting from mainframes with customer databases down to servers to complex network architectures (including classic LANs, WANs, wireless networks, and remote access systems with hundreds of remote users) to point-of-sale (POS) systems and all the way down to wireless card scanners. A handheld wireless card scanner is “a credit card processing system” and thus is in-scope for PCI compliance. In fact, credit card information has indeed been stolen this way.

Once it has been accepted that manual review of logs is not feasible or effective, the next attempt to satisfy the requirements usually comes in the form of scripts written by system administrators, to filter, review, and centralize the logs as well as

makeshift remote logging collectors (usually limited to those log sources that support syslog, which is easy to forward to another system).

For example, one might configure all in-scope Unix servers to log a single “log server” and then write a Perl script that will scan the logs for specific strings (such as “fail*,” “attack,” “denied,” “modify,” and so forth) and then send an e-mail to the administrator upon finding one of those strings. Another script might be used to summarize the log records into a simple “report” that highlights, for example, “Top Users” or “Top IP Addresses.” A few creative tools were written to implement various advanced methods of log analysis, such as rule-based or stateful correlation.

Such “solutions” work well and do not require any initial investment. Other advantages of such “homegrown” approaches are as follows:

- You are likely to get exactly what you want because you design and build the tool for your environment.
- You can develop tools that have capabilities not offered by any commercial tool vendor.
- The choice of platform, development tools, analysis methods, and everything else is yours alone.
- You can later customize the solution to suit your future needs.
- There is no upfront cost for buying software and even hardware (if you are reusing some old unused servers, which is frequently the case for such log analysis projects); however, you may run into cost allocations around storage.
- Many system administrators say that “it is a fun thing to do.”

What makes it even easier is the availability of open source and freeware tools to address some of the pieces of log management for PCI. For example, [Table 10.6](#) summarizes a few popular tools that can be (and in fact, have been) used in PCI log management projects.

On the contrary, many organizations turn to commercial vendors when looking for solutions to PCI logging and monitoring challenges. On the logging side, commercial log management solutions can aggregate all data from the in-scope entities, whether applications, servers, or network gear. Such solutions enable satisfying the log data collection, monitoring, analysis, data protection, and data retention.

NOTE

Why do we keep saying “retention” where some people would have used to term “storage?” “Retention” usually implies making sure that data is stored, but on a determined schedule and lifecycle.

Vendors also help with system configuration guidance to enable optimum logging (sometimes for a fee as “professional services”). Advantages of such an approach are obvious: on day 1, you get a supported solution as well as a degree of certainty that the vendor will maintain and improve the technology as well as have a roadmap for addressing other log-related organization needs beyond PCI compliance.

Table 10.6 Logging Tools Useful for PCI DSS

Origin	Tool	License	Purpose	Satisfied PCI Requirement
BalaBit IT	syslog-ng	Open source	General purpose syslog replacement, reliable, and secure log transfer	Multiple sections of Requirement 10 and others; enabling infrastructure
Project LASSO	Project LASSO	Open source	Remote Windows event collection	Windows logging centralization; enables analysis of Windows logs covered by Requirement 10
Various	Stunnel, OpenSSH, FreeS/WAN	Open source	Secure data (including log) transfer	Log protection sections in Requirement 10
Various	MySQL, PostgreSQL	Open source	Data (including log) storage	Log retention section of Requirement 10
Various	Swatch, logwatch, logscopy	Open source	Small scripts for log filtering, alerting, and simple monitoring automation	Automated log review in Requirement 10
Risto Vaarandi	SEC	Open source	Log correlation and rule-based analysis	Automated log review in Requirement 10 on a more advanced level
OSSEC team	OSSEC	Open source	Log analysis	Automated log review in Requirement 10 on a more advanced level
Various	SecurityOnion	Open source	IDS, NSM, and Log Management	Requirement 10, 11
OSSIM team	OSSIM	Open source	Log analysis and correlation across logs and other information sources	Automated security monitoring across various systems

Fortunately, there are simple things you can do to avoid the pitfall of unmet requirements when acquiring a log management solution.

- Review PCI logging guidance such as this book (as well as the standard itself) to clarify the standard's requirements.
- Consider how a log management solution would work in your environment.
- Define the need by talking to all the stakeholders in your PCI project and have the above information in mind.

Look through various commercial log management and SIEM tools such as LogLogic (www.loglogic.com), HP ArcSight (www.arcgisight.com), Splunk (www.splunk.com), Q1Labs—now part of IBM (www.q1labs.com), McAfee's

NitroSecurity, RSA's Security Analytics or others to "case the joint" and to see what is out there. Here are some useful ideas on how to best talk to those vendors:

- Can your tool collect and aggregate 100% of all log data from all in-scope log sources on the network?
- Are your logs transported and stored securely to satisfy the CIA of log data?
- Are there packaged reports that suit the needs of your PCI projects stakeholders such as IT, assessors, maybe even Finance or Human Resources? Can you create the additional needed reports to organize collected log data quickly?
- Can you set alerts on anything in the logs to satisfy the monitoring requirements?
- Top three capacity concerns,
 - Events Per Second,
 - Storage,
 - Licensing,
- How do you size the environment?
- How does your investigation tool perform under 90, 180, and 360+ days of data (load factor)?

NOTE

Alerts are only useful if there is a process and personnel in place to intake, analyze, and respond to alerts on a timely basis. In other words, if nobody is looking for e-mail alerts and is prepared to respond, they are next to useless.

- Does the tool make it easy to look at log data on a daily basis?
- Can the tools help you prove that you are by maintaining an assessment trail of log review activities? (Indeed, it is common for the assessors to ask for a log that shows that you review other logs and not for the original logs from information systems! Yes, log analyst activities need to be logged as well—if this is news to you then welcome to the world of compliance!)
- Can you perform fast, targeted searches across all in-scope systems components for specific data when asked? Remember, PCI is not about dumping logs on tape or disk, but about using them for cardholder data security.
- Can you readily prove, based on logs, that security (such as antivirus and intrusion prevention), change management (such as user account management), and access control policies mandated by the PCI requirements are in use and up-to-date?

When such tools are deployed, a few useful reports and alerts are typically created. On a more detailed level, here are some sample PCI-related reports and alerts for log review and monitoring.

Alerts used for real-time monitoring of in-scope servers are as follows:

- New account created,
- New privileges added to a user account,
- Firewall rules change,

- Multiple failed logins,
- Critical system restarted,
- Antivirus protection failed to load,
- Malware detected,
- Privilege elevation,
- Log collection failed from an in-scope system,
- Additions to administrative groups,
- Group policy changes (if in an Active Directory environment).

Some of the recommended reports used for daily review of stored data are as follows:

- Traffic other than that allowed by PCI,
- Software update activities,
- User account changes on servers (e.g., additions, deletions, and modifications),
- Login activity on in-scope servers,
- User group membership changes,
- Password changes on in-scope servers and network devices,
- All administrator/root activities on in-scope servers,
- Log review activities on a log management solution.

NOTE

A good list of reports can be found in SANS “The 6 Categories of Critical Log Information” that can be found at <http://www.sans.edu/research/security-laboratory/article/sixtoplogcategories>, and another great resource is Randy Franklin Smith’s Web site at <http://www.ultimatewindowssecurity.com/securitylog/encyclopedia/Default.aspx>.

OTHER MONITORING TOOLS

It is critical to remember that the scope of security monitoring in PCI DSS is not limited to logs because system logs alone do not cover all the monitoring needs. Additional monitoring requirements are covered by the following technologies (as well as process requirements that accompany them):

- Intrusion detection or intrusion prevention; mandated by PCI Requirement 11.4 “Use intrusion-detection and/or intrusion-prevention techniques to detect and/or prevent intrusions into the network. Monitor all traffic at the perimeter of the cardholder data environment as well as at critical points in the cardholder data environment, and alert personnel to suspected compromises.”
- File integrity monitoring; mandated by PCI Requirement 11.5 “Deploy a change-detection mechanism (for example, file-integrity monitoring tools) to alert personnel to unauthorized modification of critical system files, configuration files, or content files; and configure the software to perform critical file comparisons at least weekly.” (updated for PCI DSS 3.0)

We will cover the critical issues related to these technologies in the section below.

INTRUSION DETECTION AND PREVENTION

NIDS and IPSs are becoming a standard information security safeguard. Together with firewalls and vulnerability scanners, intrusion detection is one of the pillars of modern computer security. When referring to IDSs and IPSs today, security professional typically refers to NIDS and network IPS. As we covered in Chapter 4, the former sniffs the network, looking for traces of attacks, whereas the latter sits “inline” and passes (or blocks) network traffic.

NIDS monitors the entire subnet for network attacks against machines connected to it, using a database of attack signatures or a set of algorithms to detect anomalies in network traffic. See [Figure 10.5](#) for a typical NIDS deployment scenario.

On the contrary, network IPS sits at a network choke point and protects such a network of systems from inbound attacks or outbound exfiltration. To simplify the difference, IDS alerts whereas IPS blocks. See [Figure 10.6](#) for a typical network IPS deployment.

The core technology of picking “badness” from the network traffic with a subsequent alert (IDS) or blocking (IPS) is essentially similar. Even when intrusion

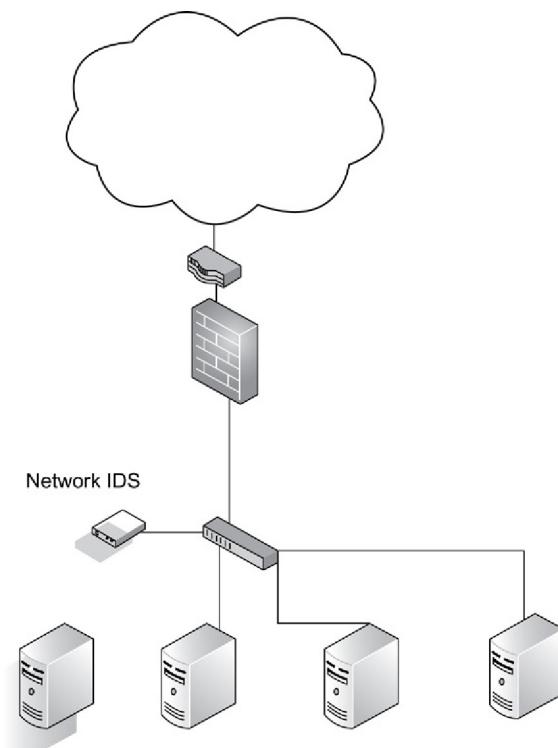


FIGURE 10.5 Network Intrusion Detection Deployment

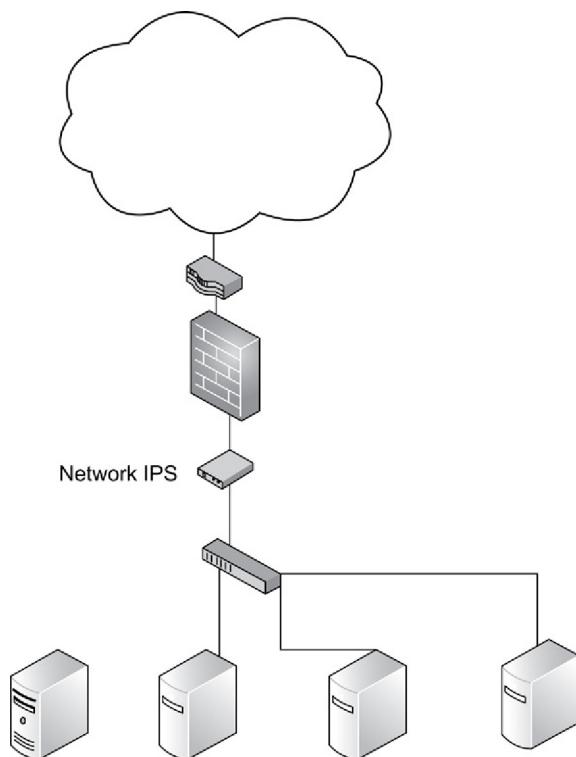


FIGURE 10.6 Network Intrusion Prevention Deployment

prevention functionality is integrated with other functions to be deployed as so-called Unified Threat Management (UTM), the idea remains the same: network traffic passes through the device with malicious traffic stopped, suspicious traffic logged, and benign passed through.

Also important is the fact that most of today's IDS and IPS rely upon the knowledge of attacks and thus require ongoing updates of signatures, rules, attack traces to look for, and so forth. This is exactly why PCI DSS mandates that IDS and IPS are not only deployed but also frequently updated and managed in accordance with manufacturer's recommendations.

In the context of PCI, IDS and IPS technologies are mentioned in the context of monitoring. Even though IDS can only alert and log attacks while IPS adds blocking functionality, both can and must be used to notify the security personnel about malicious and suspicious activities on the cardholder data networks. Below we present a few useful tips for deploying IDS and IPS for PCI DSS compliance and card data security. IPS is not required, but can be used in the place of an IDS.

Despite the domination of commercial vendors, the free open-source IDS/IPS Snort, now developed and maintained by its corporate parent, Sourcefire, is likely

the most popular IDS/IPS by the number of deployments worldwide. Given its price (free) and reliable rule updates from Sourcefire (www.sourcefire.com), it makes a logical first choice for smaller organizations. It also shouldn't be taken off the shortlist even for larger organizations seeking to implement intrusion detection or prevention.

Although a detailed review of IDS and IPS technologies and practices goes well beyond the scope of this book, we would like to present a few key practices for making your PCI-driven deployment successful.

First, four key facts about IDS and IPS, which are also highlighted in the PCI standard:

1. IDS or IPS technology must be deployed as per PCI DSS. If another device includes IDS or IPS functionality (such as UTM mentioned above), it will likely qualify as well.
2. IDS and IPS need to “see” the network traffic in cardholder; for IDS, it needs to be able to sniff it and for IPS, to pass it through. An IDS box sitting in the closet is not PCI compliance (and definitely not security!).
3. IDS and IPS must be actively monitored by actual people, devoted (full-time, part-time, or outsources) to doing just that. PCI DSS states that systems must be set to “alert personnel to suspected compromises.”
4. IDS and IPS rely on updates from the vendor; such updates must be deployed, or the devices will lose most of its value. PCI does highlight it by stating to “Keep all intrusion detection and prevention engines up-to-date.”

The above four facts define how IDS and IPS are used for the cardholder requirement. Despite the above knowledge, IDS technologies are not the easiest to deploy, especially in light of the number 3 consideration above. PCI DSS-driven IDS deployments suffer from a few of the common mistakes covered below.

First, using an IDS or an IPS to protect the cardholder environment and to satisfy PCI DSS requirement is impossible without giving it an ability to see all the network traffic. In other words, deploying an NIDS without sufficient network environment planning is a big mistake that reduces, if not destroys, the value of such tools. Network IPS, for example, should be deployed on the network choke point such as right inside the firewall leading to cardholder network, on the appropriate internal network segment, or in the De-Militarized Zone (DMZ). For the shared Ethernet-based networks, IDS will see all the network traffic within the Ethernet collision domain or subnet and also destined to and from the subnet but no more. For the switched networks, there are several IDS deployment scenarios that use special switch capabilities such as port mirroring or spanning. When one or more IDS devices are deployed, it is your responsibility to confirm that they can “cover” the entire “in-scope” network.

Port mirroring and spanning should be avoided whenever possible. Switch vendors documentation will tell you that they do not scale for this type of monitoring and you may not see all data when the network is busy.

Network taps are the preferred method of deployment for IDS. Even if you decide to put IPS in-line, you should consider using taps to get the appliances in line. Even

though all commercial IPS systems can be configured to fail open or “turn into a wire” if you do not use taps you may have to schedule a network outage whenever you have to swap out an appliance for maintenance or upgrades. In some environments, this can lead to failing to meet SLAs.

Second, even if an IDS is deployed appropriately, but nobody is looking at the alerts it generates, the deployment will end in failure and will not lead to PCI compliance. It’s well known that IDS is a “detection” technology, and it never promised to be a “shoot-and-forget” means of thwarting attacks. Although in some cases, the organization might get away with dropping the firewall in place and configuring the policy, such a deployment scenario never works for intrusion detection. If IDS alerts are reviewed only after a successful compromise, the system turns into an overpriced incident response helper tool, clearly not what the technology designers had in mind. Even with IPS, a lot of suspicious indicators are not reliable enough to be blocked automatically, thus monitoring is just as critical as with IDS.

PCI DSS Requirement 12.5.2 does state that an organization needs to “Monitor and analyze security alerts and information, and distribute to appropriate personnel.” Still, despite this, many organizations deploy IDS and develop a no response policy. As a result, their network IPS is deployed, it “sees” all the traffic, and there is somebody reviewing the alert stream. But what is the response for each potential alert? Panic, maybe? Does the person viewing the alerts know the best course of action needed for each event? What alerts are typically “false positives”—alerts being triggered on benign activity—and “false alarms”—alerts being triggered on attacks that cannot harm the target systems—in the protected environment? Unless these questions are answered, it is likely that no intelligent action is being taken based on IDS alerts—a big mistake by itself, even without PCI DSS. Some of the recent breaches of card data were directly attributed to ignored alerts from various security detection technologies.

The fourth and final mistake is simply not accepting the inherent limitations of network intrusion protection technology. Although anomaly-based IDSs might detect an unknown attack, most signature-based IDS will miss a new exploit if there is no rule written for it. IDS and IPS must frequently receive vendor signature updates, as mandates by the PCI DSS. Even if updates are applied on a schedule, exploits that are unknown to the IDS vendor will probably not be caught by the signature-based system. Attackers may also try to blind or evade the NIDS by using many tools available for download. There is a constant battle between the technology developers and those who want to escape detection. IPS/IDS are becoming more sophisticated and able to see through the old evasion methods, but new approaches are constantly being used by attackers like trusted server to trusted server or island hopping. Those deploying the NIDS technology should be aware of its limitations and practice “defense-in-depth” by deploying multiple and diverse security solutions.

Thus, IDS/IPS is a key monitoring technology for PCI DSS and data protection; however, when deploying it, many pitfalls need to be considered if it were to be useful for PCI compliance and security.

INTEGRITY MONITORING

In the prehistoric days of security industry, before all the current compliance frenzy and way before PCI DSS, the idea of monitoring key system files for changes was invented. As early system administrators engaged in an unequal battle with hackers who compromised almost every server connected to the Internet, the idea of a quick and easy way for verifying that system files was not modified and thus not subverted by hackers seemed like a god-send.

Indeed, just run a quick “scan” of a filesystem, compute cryptographic checksums, save them in a safe place (a floppy comes to mind—we are talking 1990s here, after all), and then have a “known good” record of the system files. In case of problems, run another checksum computation and compare the results; if differences are found, then something or someone has subverted the files.

That is exactly the idea of integrity checking and monitoring. The tools such as the pioneering *tripwire* (now developed by its corporate parent, Tripwire, Inc) have matured significantly and offer near real-time checks, an ability to revert to the previous version of the changed files, as well as a detailed analysis of observed changes across a wide range of platforms for servers, desktops, and even network devices.

NOTE

Question: What is the difference between host intrusion detection and integrity monitoring?

Answer: Many different types of applications are labeled as host-based intrusion detection. In general, the distinction is that with IDS, the end goal is the detection of malicious activity in a host environment, whereas an integrity monitoring system aims to provide visibility into all kinds of change. Detecting malicious change or activity is a big part of an integrity monitoring system but that is not the entire motivation behind its deployment.

As we mentioned, PCI DSS mandates the use of such tools via Requirement 11.5. Namely, “Deploy a change-detection mechanism (for example, file-integrity monitoring tools) to alert personnel to unauthorized modification of critical system files, configuration files, or content files; and configure the software to perform critical file comparisons at least weekly.”

This means that the tools must be deployed on in-scope systems, key files need to be monitored, and comparisons are to be run at least weekly. Indeed, knowing that the server has been compromised by attackers a week after the incident is a lot better than what some recent credit card losses indicate. For example, TJX card theft was discovered *years* after the actual intrusion took place, causing the company some massive embarrassment.

Some of the challenges with such tools include creating a list of key files to checksum. Typically, relying on the integrity checking tool vendor default (or even

“PCI DSS-focused”) policy is a good idea. Here is an example list of such files for a typical Linux server:

- Configuration files in/etc. directory.
- All system executables (/bin,/usr/bin, /usr/sbin, and other possible binary directories), ESPECIALLY those setuid to root.
- Key payment application executable, configuration, and data files.
- Log files /var/log (these require a special append-only mode in your integrity monitoring).

Also, just as with IDS and IPS, a response policy in case of breach of integrity is essential.

COMMON MISTAKES AND PITFALLS

By now the reader should be convinced that it is impossible to comply with PCI requirements without log data management processes and technologies in place. Complete log data is needed to prove that security, change management, access control, and other required processes and policies are in use, up-to-date, and are being adhered to. In addition, when managed well, log data can protect companies when compliance-related legal issues arise (e.g., when processes and procedures are in question or when an e-discovery process is initiated as part of an ongoing investigation). Not only does log data enable compliance but also allows companies to prove that they are implementing and continuously monitoring the processes outlined by the requirements.

- Logging in the PCI DSS is not confined to Requirement 10. As we discussed above, all the requirements imply having a solid log management policy, program, and tools.
- Logging in PCI is not only about log collection retention; Requirement 10.6 directly states that you need to review, not just accumulate logs.
- Log review in PCI does not mean that you have to read the logs yourself; using automated log management tools is not only allowed but suggested.
- A careful review of what is in-scope must be performed. Otherwise, one creates a potentially huge issue for the organization in terms of having what is thought of as a solid PCI program, but then suffering a data breach and getting fined due to missing a wireless POS system or some other commonly overlooked but clearly in-scope systems. (Well, at least it becomes clear after your organization is fined.)
- Your logging tools purchased and deployed for PCI compliance are almost certainly useful for other things ranging from other compliance mandates (see the above example of PCI and Sarbanes–Oxley) as well as operational, security, investigative, incident response, and other uses.

CASE STUDY

Next, we present two of the case studies that illustrate what is covered in this chapter.

THE CASE OF THE RISKY RISK-BASED APPROACH

This case study covers deployment of a log management solution to satisfy PCI requirements at a large retail chain in the Midwest. Bert's Bazaar, an off-price retailer servicing the Midwest and southern US states, decided to deploy a commercial log management solution when their PCI assessor strongly suggested that they need to look into it. Given that Bert has a unique combination of a large set of in-scope systems (some running esoteric operating systems and custom applications) and an extreme shortage of skilled IT personnel, they chose to buy a commercial solution without seriously considering an in-house development. So, they progressed from not doing anything with their logs directly to running an advanced log management system.

The project took a few months following a phased approach. Bert's IT staff decided to implement it from the outside based on their risk assessment. They started from their DMZ firewalls and then progressed with feeding the following logs into a log management system, while simultaneously defining alerts and running reports from vendor's PCI compliance package.

Their project proceeded as follows:

1. All Internet DMZ firewalls,
2. Select internal firewalls that control access to payment processing systems,
3. DMZ front-end processing servers—operating system only,
4. Other payment processing servers—operating system only,
5. Databases that are involved in payment processing,
6. Actual payment processing applications from all involved servers.

A few things need to be said about the above approach. One common piece of technology conspicuously missing from the list is network intrusion detection. The reason is that the organization chose not to implement it due to resource shortage (even though modern NIDS have improved, they still require people to provide care and feeding). The sequence is based on both their risk assessment and the complexity of log collection. The former led them to focus on the outside threat first, whereas the latter delayed some of the log collection efforts: it is much easier to forward Cisco PIX firewall logs to an analysis server, but a database logging configuration, collection, and analysis present a significant challenge (due to multiple factors from affecting performance to grabbing logs from a database itself in a secure and reliable manner to deciding exactly what you want to capture).

Overall, the project is a successful implementation of PCI logging requirements by using a commercial logging solution. The organization did pass their PCI assessment and was commended on their comprehensive approach to logging. In addition, the security team built a case that their PCI logging implementation actually

addresses a few other compliance mandates (such as US Sarbanes–Oxley Act) because PCI DSS goes into higher-level details while covering essentially the same areas of IT governance. At the same time, a log management tool also bolstered its operational capabilities and overall IT efficiency.

THE CASE OF TWEAKING TO COMPLY

This case study is based on a major e-commerce implementation of an off-the-shelf log management technology in combination with in-house developed tools to address a unique need alongside PCI compliance. Upon encountering PCI compliance requirements, Wade's Webbies developed its own set of scripts to go through a Web server and payment application's server logs to identify hacking and fraud attempts. Such scripts were very useful but proved to be onerous to operate, update, maintain, and troubleshoot. Additionally, a few key IT staffers who helped develop the solution left to join a consulting company.

Thus, IT management decided to pick a commercial log management application, which will have to work together or integrate with their previous scripts (that still delivered unique value to the organization); they would use a vendor's collection and analysis log management infrastructure but retain an ability to look for fraud using their own methods.

Their log management project proceeded as follows:

1. Web server logs from DMZ Web servers,
2. Operating system logs from the same Web servers,
3. Custom payment processing application logs,
4. Network logs from the DMZ (firewall, router).

The project approach was driven by the preexisting log analysis solution. Although the vendor solution was being deployed, they were adapting their scripts to run based on the log management vendor's API. Such API provided access to preanalyzed as well as raw log data and allowed the organization to retain a large part of the effort spent on developing the scripts. At the same time, such API capability allows the users of the tools to take advantage of the vendor's advanced log management technology as well as regular updates and customer support.

Overall, this project was a successful illustration of a combined approach of using a homegrown and commercial solution and thus achieving combined benefits.

SUMMARY

In conclusion, the authors would like to stress a few points that were covered as well as leave readers with a few thoughts about how PCI logging fits into the bigger picture of IT governance and risk management. Despite the fact that this book is about PCI DSS, we do not recommend that you embark on a log management project or on a security monitoring project solely for PCI DSS. Taking control of your logs is

useful for a huge number of IT and information security goals; thus doing logs “just for compliance” is akin to using an expensive and powerful tool as a hammer. Overall, a common scenario observed by the authors is “buy for compliance [such as PCI], use for security” or, even “use for all the needed purposes.” We recommend that the organizations that are subject to PCI DSS use the motivating power of PCI DSS to acquire the needed tools and then proceed to using them for solving all the problems in the whole IT security realm.

REFERENCE

- [1] Common Event Expression (CEE). <<http://cee.mitre.org>>; 2009 [accessed 13.07.09].

PCI DSS and cloud computing

11

INFORMATION IN THIS CHAPTER:

- Quick cloud intro
- Where PCI DSS and cloud interact
- PCI DSS cloud scenarios
- What guidance is available
- What are the risks
- What should be done

Checking the media headlines will lead many to believe that the new paradigm of cloud computing is taking the world by storm. And, true, organizations have been taking advantage of cloud computing models for many tasks—from making their e-commerce offerings better to testing application and storing data. Payment Card Industry Data Security Standard (PCI DSS) and payment security issues have found their areas of intersection with cloud computing.

CLOUD BASICS

Business is moving faster than ever, and competitors appear from almost every angle. Technology is a major driver in your competition's ability to gain the advantage in your industry. Many corporate leaders leap to deploy technology in a way to gain an advantage, but government and industry regulations prevent the adoption of certain kinds of technology including—in some cases—cloud computing. Slow-moving regulation can cause companies to scrap ideas altogether. In the business world, those that stand still will lose out on growth and profits.

PCI DSS is not a new standard, but its reach continues to grow as the industry struggles to self-regulate. The very first version of the standard was released on December 15, 2004, and even though it has been revised multiple times you will not find the word cloud anywhere in the document—even now in 2014, 10 years after that date. Merchants that want to leverage the power of the cloud are sometimes stuck between a slow-moving standard and an inexperienced Qualified Security Assessor (QSA)—who may not even know how to spell “IaaS”—to help them evaluate a solution for compliance. As much as it pains IT workers, compliance initiatives have the power to trump good ideas.

Do you do the right thing and abide by the letter of slow-moving compliance initiatives, or do you deploy technologies that allow you to meet the control requirements while propelling your business forward? Many instinctively want to choose the latter, but how can we accomplish this? Technology, such as cloud computing, has polarized the security and compliance community due to its relatively open nature, but the benefits are immense—and growing bigger as the power of the cloud to transform industries has grown. Can we find a place for bleeding-edge technologies in environments governed by compliance?

WHAT IS THE CLOUD?

While cloud computing is mentioned in the media all the time, few people think about what is the actual definition—and what makes computing “cloudy.” NIST Definition of Cloud Computing states that “Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction.”

Furthermore, they define these essential cloud characteristics such as on-demand self-service, broad network access, resource pooling, rapid elasticity, and others.

Cloud computing is also divided into three types:

- Cloud Software as a Service (SaaS) where you use a provider’s applications over a network
- Cloud Platform as a Service (PaaS) where an organization deploys customer-created applications to a cloud
- Cloud Infrastructure as a Service (IaaS) where an organization rents processing, storage, network capacity, and other fundamental computing resources

People in the industry will use the terms public, private, and hybrid cloud to describe how they will deploy any combination of two or more clouds models.

Cloud computing usually happens at massive scale (think Amazon or Google) and makes heavy use of virtualization and low-cost software.

Example: IaaS Amazon Cloud includes such components as Elastic Compute Cloud (EC2) where one can run your own or Amazon’s OS “instances,” Simple Storage Service (S3) for storage, and many other services.

For example, Amazon includes this FAQ entry at their site:

“Q: What does this mean to me as a PCI merchant or service provider?

A: Our PCI Service Provider status means that customers who use our services to store, process or transmit cardholder data can rely on our PCI compliance validation for the technology infrastructure as they manage their own compliance and certification, including PCI audits and responses to incidents. Our service provider compliance covers all requirements as defined by PCI DSS for physical infrastructure service providers. Moving the entire cardholder environment to AWS can simplify your own PCI compliance by relying on our validated service provider status”

Cloud badness

Many organizations have compiled lists of various risks, threats, and dangers of cloud computing

Here is one list of cloud threats by cloud security alliance (CSA)

- Abuse & Nefarious Use of Cloud Computing
- Insecure Interfaces & APIs
- Malicious Insiders
- Shared Technology Issues
- Data Loss or Leakage
- Account or Service Hijacking
- Unknown Risk Profile

These issues illustrate that cloud computing will include some of the risks of traditional IT environments as well as many new risks that do not exist in legacy IT.

CLOUD CHANGES EVERYTHING! BUT DOES IT?

As technology changes, we need to adjust how we act in the presence of these advancements, to avoid increasing our risk dramatically without the associated benefits. Yet, as humans, we force new problems into the frameworks of old solutions to try and make sense out of them. When we look to solve problems, we tend to ask ourselves where we have solved similar problems previously. Information security and compliance is a notorious example of applying old methods to secure new things as security teams are constantly keeping up with the bad guys, and compliance tries to keep up with defense techniques designed by information security. By the way, cloud computing has already offered advantages to the attacking side, such as for cracking passwords and storing stolen data.

Current generation protection mechanisms (firewalls and SSL, anybody?) are generally poorly suited for protecting next-generation technology. As new technology provides expanded features, those features must be considered for what they do, but not in the context of how things have been done before. Using a port-level blocker to protect interactions at the application layer between the components of a SaaS application would not be as effective as something that is content and context aware.

Think about how we have changed our shopping habits over the last 20 years. At the emergence of the commercialized Internet, consumers had two primary methods to buy products and services—either in person or via mail or telephone order. As businesses figured out how to generate revenue from this new channel, they had to do it in a way that didn't completely cannibalize their current business. As consumers, we've moved from casually shopping online, to full online experiences, to omnichannel retail shopping and now to mobile shopping as well.

Today, most of the customers entering your store have less than \$20 cash, and 27% of them with smartphones will buy items directly on the device, as reported by recent surveys. More of the customers are combining their physical and digital shopping experiences by using quick response (QR) codes or competing price check

apps (3). Merchants will need the optimal allocation of IT assets to meet customers on their terms—while keeping the data secure and of course their IT environment PCI DSS compliant. New concerns not addressed by PCI DSS, such as denial of service attacks for ransom, have hit some merchants hard as well.

Retail is constantly experimenting with new pathways to take products to market. They use data analytics as a way to maximize sales and dynamically reconfigure bundles for customers in the store as well as on the Web. Cloud computing plays a major role in that by making advanced analytic algorithms available to wider populations of merchants. The goal for merchants is to change their business to capture the new reality of shopping to benefit their customers—and to do it fast. Given that online retailers have already invested heavily in technology as their primary pathway to market, extending to the cloud is a natural way to provide additional services to customers in an omni-channel retailing model.

CLOUD CHALLENGES AND YOU

One of the biggest changes in how we handle IT is the way that we procure technology. Instead of physically labeled rows of cabinets, which show each machine, function, name, and IP address, data centers are virtual and the hardware type is almost irrelevant. As long as it supports the virtual fabric of the data center, it's good to run!

Both business and IT needs flexible systems. This allows managers to better plan for their overall IT spend and maximize the utilization of assets deployed in normal operations. If deployments are approached with this type of flexibility in mind, jobs can be scheduled and scaled up and down as demand dictates. Every business has a peak season, and it's financially irresponsible to build fixed infrastructure solely while planning for your peaks. A better way is to plan for averages outside of your peaks, and then spin up cloud resources to handle high demand; a practice sometimes called “cloud bursting” since the capacity bursts up by using the public cloud resources. Online retailers shouldn't plan for sustained demand in the range of the traffic they get on Cyber Monday. They need to contract those cycles out!

One of the bigger challenges companies must address when considering a move to a cloud model is the concept of shared security, as in which party is responsible for which parts of security? Indeed, this presents THE central challenge for the whole “PCI and the cloud” debacle.

Depending on your industry, there are ways around this. As an example with Cyber Monday, shopping around does not always equal purchasing. Instead of building a massive private cloud to manage everything, some companies leverage public cloud resources for browsing the catalog and then seamlessly transfer the process to private cloud resources for purchasing.

Technologists need to manage up to the finance people in today's world. Irrespective of reporting structure, the Chief Financial Officer (CFO) controls the budget allocated to IT. So in order for those of us in IT to get what we need, we have to think like a CFO and cater to his desires for running the business. Over the last 5 years, we have seen a tremendous shift from large capital outlays (CapEx) to operational charges (OpEx) for

IT expenditures. The reason for this is that OpEx can more closely tie the spending on IT with the revenue associated with operations (think back to your Managerial Cost Accounting classes). With a more accurately predicted margin on product, CFOs can smooth out the lumpy CapEx charges in the financial statements. Instead of large capital expenditures that end up going obsolete rather quickly, Chief Information Officers (CIOs) can help CFOs run a tighter balance sheet with less depreciation charges by moving these charges to OpEx through the use of elastic compute resources such as cloud.

In fact, the PCI Council offers some thoughts on cloud challenges as well. Specifically, they mention that:

- “The distributed architectures of cloud environments add layers of technology and complexity to the environment.
- Public cloud environments are designed to be public-facing, to allow access into the environment from anywhere on the Internet.
- The infrastructure is by nature dynamic, and boundaries between tenant environments can be fluid.
- The hosted entity has limited or no visibility into the underlying infrastructure and related security controls.
- The hosted entity has limited or no oversight or control over cardholder data storage.
- The hosted entity has no knowledge of —who they are sharing resources with, or the potential risks their hosted neighbors may be introducing to the host system, data stores, or other resources shared across a multi-tenant environment.”

Source: PCI DSS Supplemental Guidance on Virtualization.

Leading researchers suggest that non-IT resources want CIOs to be more like a company executive, and not just the top IT guy (4). This means that the solutions brought to other executives’ areas should have the business in mind. Clearly, the ability to scale up and down to allow IT to move with the business is valued. Virtual and cloud infrastructure scales horizontally, not vertically. Meaning, instead of focusing on one massive process to run everything, the infrastructure scales out to a massively parallel group of smaller instances that do only their share of the work. While this is great for operations, it’s not so great for audit.

Virtual assets in the cloud are ephemeral, or only lasting for a short period of time, before they are re-consumed by the infrastructure as idle resource. When it comes to security, this is a major problem. Just because an asset is no longer needed to perform work doesn’t mean that its activity and logs can be discarded and forgotten. Just sit across from an auditor and tell them you can’t produce the logs for your infrastructure because the instances no longer exist. Then tell them that you can’t even produce the records of which instances were created, when they existed, and the jobs they performed because many of them only lasted for hours or minutes. You might witness a tantrum that would put a 4-year-old to shame!

Auditors and assessors need records of work performed and persistent logs from those jobs. PCI DSS 3.0 specifically requires the maintenance of an inventory of

in-scope systems. As a cloud user, you must know which resources are in scope at any given time, and keep the logs for those resources as well. Your assessor may not ask for it during your annual checkup, but you must maintain it for ongoing compliance. Ten years ago when these systems were physically racked, stacked, and labeled, this was pretty easy to do. Now when you can't even physically put your finger on the job as it is running, you need tools to help you accomplish this task. In order to keep your auditors happy, you need to demonstrate command of your IT infrastructure. With cloud, there must be some level of automation in order to be able to keep up with the virtual assets that process cardholder data (or are otherwise in scope). This could easily extend to instances that were at one point in-scope, but have now become dormant or just don't exist anymore.

Along with the challenges of inventorying ephemeral cloud assets, logging is another big one that can be troublesome in the cloud. As an experiment, try to see if you can get your cloud provider to share their underlying logging with you. My guess is that your request will be denied, and a lack of access to logs is a huge barrier to implementing compliance-governed applications in a nonowned cloud service environment. Any cloud-based infrastructure supporting production applications must generate running logs to satisfy both the security and compliance individuals responsible for the data contained within.

PCI CLOUD EXAMPLES

Early implementations of cloud infrastructure were analogous to the types of stories you might find in western flicks starring John Wayne. It was all about the technology. Let's demonstrate its abilities and practicality, but not in a way that allows us to do things in a safe and secure way. Some of the earliest implementations were found in the development and test environments to simplify technology deployments. Then we started to see noncritical services ending up in virtual or cloud infrastructures due to the sheer speed of deployment and resource maximization challenges. Now we see more and more critical jobs taking advantage of the cloud paradigm, and it's time for the long arm of the digital law to step in and de-risk the environment.

We've all heard a skeptic cluck their tongue when someone describes their desire to put something in the cloud. They proclaim "Without eyes on the hardware, you lose control of the device," and "if I can't see it, how can I secure it?" Physical control of a piece of hardware does not a secure platform make. In fact, it could even make the platform less secure.

Software-defined security is a new automation tool that professionals can leverage to defend their infrastructure. Temporary or ephemeral computing resources are hard for an attacker to control and leverage. A constantly changing network means that attackers must re-learn your setup with every iteration. When applying game theory to information security, we learn that the more moves the defender makes against an attacker, the harder it is for the attacker to make progress. Therefore, he gives up and moves elsewhere (5). Dynamically reconfiguring physical assets is impossible, but

building this resistance into a cloud-based infrastructure is not only possible, but scriptable. Simply, any electronic asset that relies on physical security is more vulnerable to compromise than an electronic asset that doesn't need physical security to keep it safe.

The moral of this story is that it's safe to take the plunge. There has never been a better time to figure out how to leverage cloud for your critical workloads subject to compliance initiatives. As the saying goes, we know that we have arrived in a world of safe public cloud computing when NASDAQ offers products that leverages one.

SO, CAN I USE CLOUD RESOURCES IN PCI DSS ENVIRONMENTS?

This is a burning question that CIOs, CFOs, and QSAs continue to wrestle with as the technology becomes more prolific. When we look for guidance from the PCI Council, they are surprisingly quiet. You won't find any help in the standard itself as the word "cloud" does not exist in PCI DSS 3.0.

At the very least, Requirement 12.8 "If cardholder data is shared with service providers, maintain and implement policies and procedures to manage service providers..." and Requirement A.1: "Shared hosting providers must protect the cardholder data environment" apply to cloud computing.

There is supplementary guidance, but it is not specific enough and doesn't integrate well with the standard. Supplemental guidance contains a few useful pointers:

- "The allocation of responsibility between client and provider for managing security controls does not exempt a client from the responsibility of ensuring that their cardholder data is properly secured according to applicable PCI DSS requirements."
- "clear responsibilities for operation, management and reporting need to be defined for each [PCI DSS] requirement."
- "The allocation of responsibility for managing security controls does not exempt a client from the responsibility of ensuring that their cardholder data is properly secured."
- "Where the Cloud Service Provider (CSP) maintains responsibility for PCI DSS controls, the client is still responsible for monitoring the CSP's ongoing compliance for all applicable requirements."
- "Segmentation on a cloud-computing infrastructure must provide an equivalent level of isolation as that achievable through physical network separation."
- "Any shared infrastructure used to house an in-scope client environment would be in scope for that client's PCI DSS assessment."
- "If adequate segmentation is not in place or cannot be verified, the entire cloud environment would be in-scope for any one client's assessment."
- "The more security controls the CSP is responsible for, the greater the scope of the CDE will potentially be, thereby increasing the complexity involved in defining and maintaining CDE boundaries."
- "Use of a PCI DSS compliant CSP does not result in PCI DSS compliance for the clients."

- “The CSP should ensure that any service offered as being “PCI compliant” is accompanied by a clear and unambiguous explanation, supported by appropriate evidence, of which aspects of the service have been validated as compliant and which have not.”
- “Due diligence is not simply reading the provider’s marketing material or relying on a provider’s claims of “PCI compliance” or secure operations.”

Compliance is notoriously behind the advancement of technology, so how can you leverage new technology in a way that doesn’t question your compliance? The answer lies inside the standard itself, but it’s not as cut and dry as we would like.

PCI DSS 3.0 made some significant additions that attempt to make compliance and security part of Business As Usual (BAU) activities (6). While controversial, managers should understand the implications of following the guidance provided here. There are not any specific requirements around assessing against the guidance, therefore it is simply guidance. The Council’s intent is to find ways to motivate companies to incorporate PCI DSS into their daily activities by calling it out specifically in the standard.

A company’s security posture relies on a vigilant approach to defense that starts at the top, permeates through the organization, and ultimately to the front line. Rather than focus on PCI compliance, companies should implement security process into their BAU activities. Companies need tools to help them accomplish this daunting task, especially in the world of cloud. By focusing on security with your IT resources, PCI Compliance will come naturally.

Each business activity has IT support resources that support its operations. In order to keep up with the speed of innovation in your industry, you are most likely looking to tools and process to help your company get features to market quickly. DevOps, agile development, and rapid iteration have to balance capacity planning, defense, and audit controls. Attempting to tackle this with manual process is virtually impossible. Security in the cloud is different, and you should be looking to do some of the following at a minimum:

- Build security controls into the workloads themselves, irrespective of the platform it is running on.
- Add automation into logging and event monitoring, bringing the details into a central location for review.
- Consider which processes you drop into cloud resources carefully, and ensure that the underlying platform’s security matches the risk associated with moving work onto that platform.

MORE CLOUD FOR BETTER SECURITY AND COMPLIANCE?

If you meet with any of the payment brands, perhaps during their office hours at the annual community meetings, they will tell you that PCI DSS was never designed to put companies out of business from the costs of security controls. Its purpose

is to reduce the risks associated with processing payment card data. IT innovation is pushing the limits on how we operate the digital side of our business, and PCI DSS requires that your security controls work in harmony with IT innovation. New IT infrastructure requires new security controls.

Whether you are in retail, manufacturing, transportation, or hospitality, your business will enjoy prosperous peaks and weather challenging valleys. The size of your business is irrelevant—planning for peaks can be challenging in the world of IT operations. The cloud isn’t the scary place that it used to be, and as an important stakeholder in your business (employees are key stakeholders) it’s your responsibility to figure out how to maximize asset utilization in your operations. Be a hero to your CFO and show the business a way to operate your IT environment efficiently.

Take advantage of the cost savings of cloud, and the ability to more closely tie the expenses of running your business with the revenue generated through operations (move to OpEx). Be the IT magician who can scale operations such that no matter the traffic load, your infrastructure responds with a snap. Do this with all of your auditing requirements met while keeping the data safe beyond just a compliance requirement. Leverage the cloud as a security tool to build a more resilient infrastructure that keeps attackers from gaining ground by constantly changing the game. And finally, free up cash to use in other parts of your business, not in massive IT capital expenditures for assets that become obsolete quickly.

MAINTAINING AND ASSESSING PCI DSS IN THE CLOUD

You must test your cloud infrastructure much like you would your physical infrastructure, but the nature of software-defined security allows for much more real-time testing. PCI DSS has a number of controls that you must test periodically, but why wait for a compliance requirement to tell you to test something? That’s like turning your firewall on for the day of the quarter that you test it, and then turning it off for the rest of the quarter assuming your network will be just fine. Attackers don’t conform to a schedule, so neither should you.

In order to maintain a healthy peace of mind toward PCI DSS, consider solutions with continuous monitoring and alerting to tell you when things are eking out of compliance. Just like you build automation into other parts of your new IT, automate your compliance checking such that the systems do the work for you. After all, if new virtual machine images are deployed automatically, how can compliance still be manual?

Don’t rely on a periodic check, make sure that your system has everything it needs to continuously monitor your environment for security and compliance violations.

When it comes to your assessment by either QSA or ISA, you may need to spend some time training the QSA how the controls are met with your cloud security tools. Just like security professionals struggle to keep up with the changes in technology,

QSAs and ISAs are often presented with nuances in environments that don't fit into a conventional infrastructure (7). Remember, cloud is not a term used in PCI DSS v3.0; therefore; you should expect that QSAs do not have all the knowledge required to assess a complex cloud environment. Spend time walking your QSA through the process by which you designed your controls for the cloud. Ensure you tie back requirements to specific controls or features, and allow the QSA to test their effectiveness (as a new subrequirement of 11.3).

CLOUD AND PCI DSS IN DEPTH

Now is the time to discuss various cloud computing models and their interaction with PCI DSS

ENTER THE MATRIX

The central point of any PCI compliance and the cloud exercise is the shared control matrix.

PCI SSC Virtualization Guidance (June 2011) contains such a high-level example of it:

Example of how scope and responsibility may differ by type of cloud service:*

	Type of Cloud Service		
Area of Responsibility	IAAS	PASS	SAAS
Data			
Software, user applications			
Operating systems, databases			
Virtual infrastructure (hypervisor, virtual appliances, VMs, virtual networks etc)			
Computer and network hardware (processor, memory, storage, cabling, etc.)			
Data center (physical facility)			

* **Note:** This is an example only. Cloud service offerings should be individually reviewed to determine how responsibilities between the cloud provider and cloud customer are assigned.

Key cloud items:

“CSP should clearly identify which PCI DSS requirements, system components, and services are covered by the cloud provider’s PCI DSS compliance program.”

SUMMARY

To conclude, the best and easiest way to become compliant in the cloud is to avoid exposure of unencrypted payment data to public cloud environments. It is useful to read the sentence a few times! Just like deleting payment data wherever possible is the easiest way to become compliant in traditional IT environments, avoiding toxic data in the cloud is the easiest way to achieve PCI compliance in any public cloud computing model. In essence, “Kill the scope” works in the cloud as well.

Also, it is better to have the payment processor handle more and the merchant/CSP handle less of the PCI burden. The CSP may do it, but MERCHANT is responsible and need to validate it—PCI Council cloud computing guidance is adamant about this.

Finally, a matrix of shared responsibility must be created by any merchant that considers utilizing public cloud providers. Make a matrix of shared responsibility (and “keep it with you at all times”!) Remember: the MERCHANT is on the hook, even if the CSP does it (as per PCI DSS).

The organization must involve legal departments in SLA and other discussions about regulated data in the cloud. “Trust but verify” principle MUST be applied to your CSP.

Scoping in the cloud presents new challenges—an organization should scan for YOUR sensitive data being put in the cloud by business partners—in THEIR clouds.

Finally, one can use Use PCI + cloud security thinking for other sensitive data: SSN, PHI, financials, and others.

FURTHER READING

- [1] PCI DSS Virtualization Guidance. https://www.pcisecuritystandards.org/documents/Virtualization_InfoSupp_v2.pdf
- [2] PCI DSS Cloud Computing Guidance. https://www.pcisecuritystandards.org/pdfs/PCI_DSS_v2_Cloud_Guidelines.pdf
- [3] Presentation PCI in the Cloud by Anton Chuvakin/CSA. https://cloudsecurityalliance.org/wp-content/uploads/2012/07/CSA_PCI_CLOUD.pptx

Mobile

12

INFORMATION IN THIS CHAPTER:

- Where is mobility addressed in PCI DSS 3.0?
- What guidance is available?
- How does PA-DSS 3.0 fit?
- Deploying the technology safely
- Case study

Mobility continues to grow in significance as consumers integrate these devices into their personal lives. The whole concept of Bring Your Own Device (BYOD) is fueled by consumer-driven IT movements everywhere. In this case, individuals have leveraged mobility to create efficiencies in their personal lives, which then translates into ideas to become more efficient professionally. While BYOD is outside the scope of this chapter, we will be covering some of the key elements of mobility and how it integrates with PCI DSS 3.0.

WHERE IS MOBILITY ADDRESSED IN PCI DSS 3.0?

PCI DSS 3.0 has a few references to mobility in Requirements 1.4, 4.1, 5, and 11.1. If you are considering adding a mobile strategy somewhere near your payment processing, you will need to consider these as well as many others. Remember, just because the requirement doesn't say "mobile" doesn't mean you can ignore it in a mobile environment. As an example, any security patches released for a mobile device must still be applied within the required periods established in Requirement 6.

Consider mobility, in the context of PCI DSS, as a method to do your normal work as opposed to something different and unique. If you approach it this way, it will fall into PCI DSS pretty clearly and the nuances of the mobile platforms become somewhat irrelevant. Your goal is not to reinvent PCI DSS compliant mobile solutions, but to apply the same precautions to mobile that you would anything else.

NOTE

Choose your mobile platform wisely! Not all platforms are alike, and even ones that people advertise as secure can be used for nefarious purposes. If you are considering accepting payments through a mobile device, be sure to choose a solution that is tested or leverages something purpose built for the capture process. Avoid hand keying in transactions as that can be risky and more expensive to service long term.

WHAT GUIDANCE IS AVAILABLE?

Fortunately, since our last edition the Council has been hard at work releasing documents that address mobility. You can see the list of them by going to the PCI Council's Web site, clicking PCI Standards & Documents, then Documents Library, and then All Documents. Just search for Mobile. As of the time of this writing, the following guidelines are available:

- PA-DSS and Mobile Applications FAQ
- Accepting Mobile Payments with a Smartphone or Tablet
- Mobile Payment Acceptance Security Guidelines for Merchants, v1.0
- Mobile Payment Acceptance Security Guidelines for Developers, v1.0

Over the last few years, they have touched almost all of the key areas where questions have popped up, with various levels of detail. For example, the Accepting Mobile Payments with a Smartphone or Tablet document is only two pages, but describes the basic concept of what they consider acceptable. Unfortunately, none of this has really made it explicitly in the Standard, so you may run into some difficulty depending on the third party you work with. Just remember, use the guidelines provided by the Council to help you build a safe solution, ensure the solution complies with the baseline PCI DSS standard, and use other security Web sites to bolster the security of the solution.

HOW DOES PA-DSS 3.0 FIT?

Two of the documents mentioned in the last section deal specifically with payment acceptance for developers and merchants, and there is only a single reference in PA-DSS 3.0. Requirement 11.1 is the only place where mobility is mentioned, but you still must apply all of PA-DSS to a mobile application. Remember, think of mobility as your method of working and think about it in that frame.

Now, this book covers PCI DSS 3.0, not PA-DSS, so we won't go into the gory details here. Before you build the next payment application for your smartphone, read through PA-DSS as well as the Mobile Payment Acceptance Security Guidelines for Developers, v1.0.

DEPLOYING THE TECHNOLOGY SAFELY

Mobility can be used safely in the field, just like we discussed how cloud can be used safely in Chapter 11. The biggest challenge with any type of technology deployment is finding qualified people who fully understand the security and business implications for said technology. Remember back in the late 1990s when everyone wanted a Web site and we all had that one guy we knew who could do it? Maybe he was a brother-in-law, uncle, nephew, or just some guy at the gym, but we all knew someone who was a web wizard. Of course, reality was somewhat different. Nobody is a web designer after purchasing Dreamweaver for the first time.

There are a few guidelines that you will find all over the web about using mobile applications in the work place, so let's review a few.

Remember that mobile devices are just smaller computers. They can suffer from the same maladies that your laptop can. If someone puts malicious software on the mobile device, it can be considered compromised. Jailbroken devices remove specific controls set by the device manufacturer that are designed to keep stuff like that off of the device, so never jailbreak one you are using for payments. You also might consider only allowing them to use Wi-Fi, and further locking them down with firewall rules, proxies, and policies that are centrally deployed. You will need to do something specific on the device anyway per Requirement 1.4, so consider defining and deploying a security policy for the devices. In addition, don't forget that mobile devices are just another IP device on your network. An attacker can use a compromised mobile device as a launching point to attack your network.

Along those lines, the data stored on the device must be protected. If you don't encrypt the device, an attacker can extract the data from it. Now here's a scary reality that we are starting to see now. Please, please, please do not store payment card information for your best customers anywhere on the device. Branden has seen situations where a merchant puts payment information into a contact record, which syncs with iCloud, which syncs with a desktop or laptop, which syncs with Google, and others. You get the picture.

Finally, by default these devices are, well, mobile. Theft is a serious issue. This is why these devices need a corporate policy and should be managed by a Mobile Device Manager (MDM). You want the ability to track and or wipe a device that leaves your premises, and render it effectively useless.

Some MDM solutions virtually "segment" the device to make secure areas, while other focus on the need to control the entire device. MDM solutions are still evolving, even at the time of publication. While the basic controls will be in place, more advanced controls may not be very mature yet. As an example, renaming a virus can sometimes be enough to bypass the security of a surprising number of MDM and mobile protection solutions.

When you choose to deploy these devices, consider getting a dedicated device for any part of your business that is handling payments—including your Point Of Sale (POS). It's never a good idea to use your mobile POS as an attack platform to throw birds at pigs, to get your clan ready for a clash, or crush some candy. Keep work and pleasure separate!

Regardless, the technology is great! Don't shy away from implementations that use mobility as they can be extremely useful and flexible. All it takes is your creativity to make the technology safely work for you.

CASE STUDY

The case study below shows how a mobile implementation can go well.

THE CASE OF THE SUMMER FESTIVAL

Jennifer's Jungle is a café that specializes in bringing free-trade coffee from small farms and growers throughout Africa and South America. She brews and serves coffee, sells pastries from a local baker, and offers her roasts in one pound sacks for sale. She set up her shop off Main street in her town and enjoyed frequent visitors. Because she is near the historic district of the town, the streets are frequently closed for festivals throughout the year. Unfortunately, during those festivals, her shop is somewhat blocked from the view of the street and people typically do not stop in.

Jennifer decides to set up shop inside the next festival, Summer Fest, where she can sell iced coffee drinks to customers walking along the street. She buys a small table, rents a space, and prepares to serve customers, but she needs to process payments remotely for people who wish to pay for her wares with a credit card. She has heard of ways to process cards through her phone, but is skeptical of the security of the devices given the recent breaches.

She places a call to her credit card processing sales rep, Zack. He informs her that he has two options for her to process payments at the festival. The first is a "sled" that she can plug her phone into. All she needs to do is download an app, drop her merchant credentials into the app, and plug her phone into the sled to process payments. The second option is a payment terminal that looks exactly like the one in her store, but it has a cellular radio in it and a rechargeable battery. She opts for the second, and Zack delivers it to her 2 days before the festival.

During the festival, Jennifer had one of her single biggest days yet. The terminal worked great, and she operated it just like the one in her store. She could even tap into a loyalty program where people can enter their cell phones to get a text receipt and promotions.

As her business grew, she was looking to take the information from the loyalty program and tie it into a larger database of customers. Zack suggested that she purchase a tablet-based POS system that would sit on the counter to track orders electronically. The benefits of the system are significant in that she can remove a paper process in favor of an electronic one, and quickly understand her best products, bundles, features, and customers. She takes cards through an add-on supplied by the bank that plugs into the headphone jack. Because she leveraged a consultant given to her by Zack to set it up, she is confident the system is safe. She pays the same firm a small fee every month to manage the devices and ensure the software is kept up to date.

SUMMARY

Mobility is a big technology driver in both our personal and professional lives, and the ability to transact business over them is expanding. The Council published several documents since our last writing that can be helpful to merchants and application developers alike—some of which are written with the small business owner in mind. Don’t shy away from these technologies, but instead deploy them safely.

PCI for the small business 13

INFORMATION IN THIS CHAPTER:

- The risks of credit card acceptance
- New business considerations
- Your POS is like my POS!
- A basic scheme for SMB hardening
- Case study

One of the key issues we face with respect to Payment Card Industry Data Security Standard (PCI DSS) is the sheer number of merchants that process transactions. The merchants that fall into the top reporting levels (Levels 1–3 for most payment brands) may process a significant percentage of transactions individually and when measured as a group, but the account for a tiny percentage of total merchants (indeed less than 1/100 of a percent). That means if you are reading this book there is a significant chance you are a Level 4 merchant, or another small business, and you are absolutely flipping your lid with the depth of this standard. Both authors have worked with companies big and small—some as small as a few employees. We feel your pain!

And it's not just us either. The 2011 community meeting brought us a big focus on small business with the suggestion of several Special Interest Groups. Some of the final work product is available on the Council's Web site, and this is a continual discussion across the industry. If you are a small business (spoiler alert!), do yourself a favor and seriously investigate a Point to Point Encryption (P2PE) or outsourced solution using EMV terminals to make this easier.

This chapter will explore several ways that you can cope with PCI DSS as a small business, and hopefully never end up in a situation where you are facing fines and fees from a breach. The knee-jerk reactions are two-fold, and they are more of a possibility than you might think. Outsourcing is a biggie: what makes you think you are qualified to own and operate a payment processing company? If you are doing all of your processing on your own, that's exactly what you are doing. The second is choose an alternate method to exchange money. You don't HAVE to accept payment cards. I bet every one of you can think of a cute little diner or bar that is cash only with an Automated Teller Machine (ATM) out front. For you non-US readers, ATMs in establishments are more common, even if you may still be able to pay with a credit card.

THE RISKS OF CREDIT CARD ACCEPTANCE

Small companies tend to think that things won't happen to them. In some respects, they aren't that wrong. It's a numbers game, and criminals want the biggest payoff they can get for the lowest amount of effort and risk. Why would someone target a small business for minimal gains when they can go after a big box retailer and steal tons of information? Therefore, as a small business owner, I don't need to concern myself with information security. I need to focus on making my widgets and preventing fraud or physical threats.

The authors enjoy working with small business owners. Their passion is infectious! When speaking with a franchisee of a major brand, he told us that cashless payment transactions are simply a convenience for his customers. If he has an extra \$15,000 at the end of a month to invest in his business, he'd rather invest in something that brings in revenue versus something that keeps his business safe from hackers. To him, that hack is a "Black Swan" event. He no doubt knows someone who has dealt with it, but probably dismisses that experience as something that won't or can't happen to him.

Small businesses tend to view payment card acceptance as a necessary item to reach their customers. In fact, there are many benefits to accepting payment cards like less cash on hand, quick wire transfers to your bank account, and the opportunity to track buying patterns of your frequent customers. Many small businesses actually prefer payment card transactions to other instruments like checks due to the risks associated with nonsufficient fund charges and the delays in converting the paper to cash. But there are specific drawbacks as well that typically only manifest themselves when you have a problem. Small businesses don't focus on information security, so they typically never know a breach happens until their acquirer makes the "Houston, we've had a problem" phone call.

If you end up on the wrong end of a payment card breach as a small business, you have a significant probability of losing your business. Imagine a small business that nets the owner \$150,000 per year being hit with fines and fees totaling over \$100,000! It's like having your building 75% burned to the ground without fire insurance. Here's typically what happens:

1. Small business is informed by a payment brand (sometimes via their acquirer) that their location has been identified as a common purchase point for a large number of known-compromised cards.
2. Small business must perform a forensic investigation (\$30–70K).
3. Small business faces lawyer and consultant fees to help them navigate the process (\$40–100K+).
4. Fines come down from various payment brands depending on the size/scope of the breach and how good your lawyers are (\$5–100K+).

All this because you entered into an agreement to accept payment cards and the systems you own and operate were not built securely. And possibly because, "That's always how we've done it."

This is why outsourcing becomes an amazingly compelling argument when weighing the cost of operating these systems with the risks of a payment card breach. For the most part, your customers don't see any real effects when their cards are breached. Some will just have new cards show up in the mail while others will notice bad transactions that are immediately reversed and new cards then are mailed. Your loyal customers won't stop patronizing your business, and unless you are in the business of securing payment card data, you probably won't see a major hit in your brand value. There are plenty of examples that prove this point freely available to you with some clever searching.

So why don't more businesses outsource? Because the fees are higher! Think about it this way: if you are required to secure cardholder data, you can either spend the money to do it yourself by complying fully with PCI DSS, or you can incrementally pay an extra point on every transaction to someone else to handle it for you. Small business owners don't want to mess with payment cards, they feel like they must do it for their customers. Imagine the burden you could remove for a measly 1–2 percentage points per transaction! As a small business owner or operator, you must make security and compliance a core part of your competency *if you choose to operate in a manner that puts you in the cross-hairs of regulation.*

NOTE

Are you seeing a trend here? While you will never hear anyone from the PCI Council say this, there are absolutely alternatives to accepting payment cards and if you are not investigating them, you are doing a disservice to you and your shareholders. You don't have to know anything about PCI DSS, but if you actually read the contracts you signed when you got your merchant account, you would know that you have to comply with certain security standards. You obviously can choose to ignore these standards, but you look ridiculous when you point the finger at someone else after a breach. If you are breached, it's easy to blame everyone but yourself, but it's your responsibility to look after your business and your investment. Don't end up on that road!

Small business owners need to know the level of risk they carry to understand how future actions alter that risk profile. For example, an overleveraged company probably wouldn't choose to take on more debt to expand (solvency risk), and a company relying on credit cards for customer revenue should know how security and compliance rules affect their business (compliance risk).

Of course, it's getting harder for business owners to learn about their risks thanks to complex software packages being offered as a service. Business owners are attracted by the glitz and glamor of a fancy piece of software, yet they don't really understand how to use it or understand the liabilities associated with the information stored—that is, until a breach happens.

NEW BUSINESS CONSIDERATIONS

Let's say you picked up this book because you heard about PCI DSS from your previous life, and you are looking to start your own business in the next few months. You know that you will be accepting payment cards, but you are unsure how to deal with

these regulations. Here's a quick guide and some things you should consider when setting up your process for accepting payments.

Your first reaction will be to get out your calculator and play with the bottom line, *how much does this cost me per transaction?* Understand that it is almost impossible to fully calculate down to the penny what it will cost you. Some offerings hide fees and other elements of interchange to where it is challenging to compare apples to apples. That said, cost is still a critical consideration for any small business owner, and you should understand what goes into that cost before you sign up for the service. The authors have helped companies set up their various payment schemes over the last 15 years, and while the offerings and technology has changed, we all still gravitate toward the cost of the solution. Not all solutions are alike, so don't assume that you are making an apples-to-apples comparison when lining them all up in your spreadsheet. Here are five things you should consider while weighing your options:

1. Accepting credit cards costs money but provides convenience and physical security (less cash on hand). If you choose not to accept cards, you should understand the costs of dealing with cash, bad checks, and counterfeiting. You must also consider things like your average ticket size when making this decision. If you sell TVs, you can't live on cash only for the most part. But if you are a restaurant, news stand, or any place where the average ticket size is under \$50, you probably can.
2. Look closely at the plans and understand their differences beyond just the finances. The following six things typically make up the majority of the cost you might see (this list is not exhaustive, but makes up a significant portion of the cost you would pay):
 - a. Will one ISO offer a complete outsourced solution, and take all the burden of PCI DSS compliance off your hands? (WIN!)
 - b. Will they handle chargebacks for you?
 - c. Will they cover fraud if someone uses a stolen card in your shop?
 - d. What cards are accepted?
 - e. How fast do you get your cash?
 - f. How and when do they take their fees?
3. Choose not to accept cards, but provide an on-site ATM or accept PIN-Debit with an outsourced provider (to push compliance back to them for member branded cards that can be used as PIN-Debit).
4. Go exclusive with one provider like Costco or Sam's Club to potentially get a better deal, but ensure you have covered PCI somehow (either yourself or outsourced back to the provider). These are harder to come by nowadays.
5. Offset some of the costs of card processing by offering discounts to cash-paying customers. You cannot require people to pay for the privilege of using a credit card (for the most part), but you can reward customers for paying with cash. You don't even have to give them discounts, it could be additional points in a loyalty

program, a free gift with purchase or some other way to encourage folks to part with paper bills instead of digital cash.

These are all business decisions that are based on the assumption that you have to deal with cardholder data at some point, and you are better served by making it someone else's problem and treating it like a standard overhead cost. Focus on retailing, not processing payments, and you may just find yourself never having to deal with PCI DSS!

YOUR POS IS LIKE MY POS!

Point-of-Sale (POS) software tends to be the biggest focus on payments for small businesses. They understand that tracking the customer's sale and the card swipe plus all the magic that happens afterward starts at the POS. When you first start looking around you may feel like there are too many choices available. The reality is there are not really that many options out there, and the market is largely dominated by a small number of major players. The online or e-commerce world isn't too dissimilar in some respects, but the main danger with this side is that there are many free software packages available to process cards. Free isn't necessarily bad, but most small business owners need commercial support to get things running well. The chances are that if you choose one of the major commercial players, you aren't the first to do so, and you can bet that there are other small businesses nearby that have the exact same system deployed in their stores.

You might be thinking, "Great! That means all the kinks are worked out and I will probably have a great experience with my software." In many respects you are absolutely correct. But there is a hidden snake in the grass that many of us ignore. It's the collateral damage of a POS system being compromised because there is a fundamental flaw in the system, not the way it is deployed.

Remember, criminals want to maximize their payoff while minimizing their effort and risk of being caught. A criminal is less likely to target a single store and more likely to target some kind of common infrastructure shared by hundreds or thousands of stores. If he can find a way to break a POS terminal and can scale that hack from one store to thousands, he's absolutely going to do that. He might even be able to fly under the radar for a bit as the payment brands try to understand where the real issue is. It will start to look like a bunch of unrelated incidents until they see that they are all running the same software and then it becomes a major issue. We can also bet that those small merchants won't be fully compliant with PCI DSS, so even though the breach itself may not have been mitigated by a compliant solution, the fact that they are operating in a noncompliant fashion opens them up for fines and fees. That small business just became collateral damage in a larger, organized attack.

Understand that this chapter is designed to scare you a bit, but with a good outcome. Payment card data left unsecured is a silent cancer waiting to go malignant. You don't need to carry the risk of a breach on your shoulders, you need to make it

someone else's problem. There are plenty of companies out there to choose from, and outsourcing can even make your position stronger from a negotiation perspective if costs from your current provider start to get out of whack.

A BASIC SCHEME FOR SMB HARDENING

So you have been sufficiently terrified of payment card processing and have already started looking to outsource. But since this isn't something you can do overnight, what *can* you do now to help mitigate much of your risk of an electronic intrusion at your store? One element is effective firewall controls that are probably built into the router you have at your location. If you are doing your best to block inbound traffic, have you considered what you are allowing out? Can someone access their Gmail account from a POS terminal? What about a gambling Web site? What about doing File Transfer Protocol (FTP) or Secure SHell (SSH) transfers?

The majority of the guidance you probably have run into on firewalls focuses on how to limit traffic from un-trusted networks into trusted networks. Outbound traffic tends to be much trickier for several reasons like:

- You have to do an analysis of your business critical applications and the traffic passed to the Internet,
- You need to have policies in place governing access,
- You should probably have some controls to prevent employees from going around your policies and rules.

The first one is sometimes the hardest one to accomplish—even as a small business. Traffic analysis is easy if you have the right tools, but small businesses rarely do. You may need to resort to using your firewall to tell you exactly what is typical traffic for your business.

Also, for the record, you should not be allowing Gmail access from a POS terminal, or gambling Web sites, and definitely not SSH and FTP. We're not saying there aren't legitimate business reasons to do any of these things... well, wait. Yeah, we are. Don't do these things from an in-scope PCI DSS device.

TIP

For a detailed post on how to do this for your business, visit this blog post by Branden: <http://j.mp/dAku0U>. In it, you will learn how to handle firewalls in your small- to medium-sized business.

You will also probably want to start limiting traffic to certain Web sites. One easy (and cheap) way to do this is to set up a DNS Blackhole for certain Web sites. There are plenty of resources available online to show you how to do this, but suffice it to say, you can black out massive portions of the Internet by routing lookups to domains to the loopback address of 127.0.0.1. Imagine blocking Facebook with one entry in

your server! It's possible, and will force your employees to use their personal devices to access nonbusiness related sites.

CASE STUDY

Your business may not fully enable you to convert to a cash business, but there are always strategies for reducing your risks. As a small business owner, you have much more to lose than an employee of a big business, so be sure you get good advice, read your contracts, and understand your risks fully before playing in the payment card space.

THE CASE OF THE OUTSOURCING DECISION

Schafer's Sommelier Sanctuary is a new upscale wine shop that caters to choose wine consumers and wine stewards. Michelle has been a connoisseur of fine wines for many years, and started a small wine club to meet and taste fine wines from all over the globe. After much urging from her growing membership, she finally decided to open an official business location. She found a great location that requires minimal finishing costs for her store and is studying how to handle payment acceptance. She knows that cash only is not an option as her average ticket size will be well over \$50, and she wants to provide a concierge level service to her frequent customers to ensure they are serviced with minimal barriers.

She receives bids from several Independent Sales Organizations (ISOs) but is having a hard time choosing as all of their features are vastly different. She makes a table with her top two options to compare the merits of insourcing our outsourcing. She is computer savvy, but does not wish to spend a bunch of time messing with her POS system if she doesn't have to. She ends up with something that looks like this in [Table 13.1](#).

In order for her to choose to outsource her transactions, she must be willing to pay an extra \$20,000 to completely remove the risk of a PCI Breach from having a major effect on her new business. That's a pretty steep charge, but works out to be around \$1,650/month, well under the salary she would need to pay to hire someone to deal with all of her IT and security concerns, and less (marginally depending on the quality) than what it would cost to hire a contractor to maintain the systems. In addition, the outsourced provider will allow her to make profiles for her top customers and include the payment card information there so she can charge her regulars without

Table 13.1 Basic Outsourcing Cost Analysis

Option	Xact Fee	Pct Fee (%)	Ave Ticket Size	Total Sales (\$)	Total Cost
Michelle	\$0.25	3	\$125	\$1,000,000	\$32,000
Outsource	\$0.25	5	\$125	\$1,000,000	\$52,000

even asking for their payment cards. She also has the ability to set up a monthly wine club that automatically bills her customers for the wine they will receive as a part of their membership. Because she chooses to outsource, she is now able to do this without the fear of a PCI breach ending her business.

SUMMARY

Small businesses learning to deal with PCI DSS feel like they have a near insurmountable hill to climb—so much so that many of them choose to ignore the problem and just pray that they will not be the target of a criminal attack. Companies have options when it comes to accepting customer payments, and they may opt to alter their business to accommodate these lower risk changes. For the most part, they absolutely cost more, but there are some interesting advantages that open up when you consider outsourcing this headache to a provider who bases their business around the flow and security of payment cards.

If you find yourself in this situation—be it a new company accepting payment cards or an existing company learning about PCI DSS—consider your options carefully. You have seen an outsourcing theme repeated heavily in this chapter and throughout this book. You should build some financial models to understand the impact of your decisions and be sure you fully understand your current risk and liability if you continue to operate your business as-is.

If you take anything from this chapter, small businesses should remember the three Ds of Safe Payment Processing:

- Delegate (Outsource, you may pay an extra point, but you don't have to worry about PCI DSS if you do it right)
- Destroy (After use)
- Don't Store (Period)

Managing a PCI DSS project to achieve compliance

14

INFORMATION IN THIS CHAPTER:

- Justifying a business case for compliance
- Bringing the key players to the table
- Budgeting time and resources
- Educating staff
- Project quickstart guide
- The PCI DSS prioritized approach
- The visa TIP

You have determined that your organization needs to comply with the Payment Card Industry Data Security Standard (PCI DSS) and, looking at the requirements, you are not sure where to start. Should you jump in and go through the 12 PCI DSS requirements and one Appendix linearly one at a time documenting that all of the requirements are in place? Or should you first figure out at what level you need to validate your compliance? How will you make sure that your fellow associates are on board with the changes you are proposing so that you can effectively and efficiently comply with PCI DSS? Is senior management behind you? How about the IT department that will actually be doing most of the work? How will you make the compliance effort come together? After putting the plan together, how will you ensure that your fellow associates have the training and information in front of them to help keep your company from falling out of compliance? Putting together a comprehensive plan will allow you to manage your compliance project efficiently and, in the end, achieve and maintain PCI DSS compliance as well as efficiently validate it.

This chapter will answer your questions about how to achieve compliance. You will learn how to justify putting in the effort and figure out if you need to comply at all. Once you know you must comply with PCI DSS, we will explore how you will bring all the players to the table to help build and enforce the compliance plan. You will read about tips on how to budget your time and resources so that you can achieve compliance quickly. Once you have your plan in place, you will need to get the message out to your staff and ensure they receive the right training to make sure your organization does not fall out of compliance. By the end of this chapter, you should have a clear plan on where to start with your own PCI DSS compliance efforts and the steps you will need to plan a program to meet compliance.

JUSTIFYING A BUSINESS CASE FOR COMPLIANCE

One of the first steps of any compliance plan is to justify putting in the effort. You must first figure out if you need to comply with the PCI DSS regulation and also figure out if you have overlap from other compliance plans already in place. Once you know compliance is a must, you need to figure out at what level you need to validate (although this should not impact the actions you take when securing cardholder data). PCI DSS compliance mandates apply to up to four different groups depending on volume and the medium by which you accept payments. The biggest question should be, “What is the cost of noncompliance?” Because compliance with the PCI DSS is mandatory, you will be hit with fines today depending on your merchant level, and ultimately your credit card processing capabilities could be terminated. The fines that are rolling down to merchants today should provide concrete numbers for the total annual penalty you can expect. In addition, the upcoming changes with respect to EMV card acceptance may be another way to fund your compliance program. In order to qualify for the reduced liability around chargebacks, a significant amount of your terminals must be EMV capable, which provides an opportunity to deploy a Point to Point Encryption (P2PE) solution right from your terminal. Another form of motivation should come from the fear of living through a breach. Fear, uncertainty, and doubt (FUD) have no place here, but let’s not dismiss the motivational power of fear. If you have never had the opportunity to manage through a major breach, ask around in your industry. There are plenty of individuals who can help you frame your message properly such that you can make a positive impact and get the funding and support from the top that you need.

FIGURING OUT IF YOU NEED TO COMPLY

Your first step with any compliance effort should be figuring out if you need to comply with a regulation. Regardless of the state of the economy, no company wants to waste time putting in measures that they are not required to have. Once you have figured out if you need to comply and what your validation requirements are, you will be in a good position to make your case to management.

NOTE

If you know you have card data in your environment, then you will have to comply with PCI DSS, but are you a merchant or a service provider? Many merchants offer ancillary services to franchisees or even to other local companies to defray the costs of running their payment processing network. By doing this, many merchants end up being service providers and have slightly different reporting requirements for each payment brand. If you are accepting payments from any third party (like a franchisee) for processing, you are most likely a service provider. Consult with your acquiring bank or a Qualified Security Assessor (QSA) to clarify this before you go too far down your compliance project path!

COMPLIANCE OVERLAP

Once you determine that you have to comply, you need to look at the other compliance plans you have in place (if any). One sure way to fast track your PCI DSS compliance program is to leverage investments made for other compliance or security initiatives. Compliance and information security initiatives often overlap (as shown in [Figure 14.1](#)) because most of the regulations are based on good business and security practices. So, pull out your Health Insurance Portability and Accountability Act (HIPAA) of 1996 and Sarbanes–Oxley (SOX) compliance plans, and figure out which components you can reuse for your PCI DSS compliance plan. If compliance is nothing new to your organization, you may have invested in a control set that is unique to you, but maps to all of the major compliance initiatives. You can reuse those controls here as well. Who knows, you might find that you are already compliant but need to show that the measures you have in place are consistent with the PCI DSS regulations. For more information on how common compliance initiatives overlap, check your nearest search engine for a number of sources that show overlap. For a fantastic general overview, check out this article from your nearest library: Constantine Gikas's 2010 article titled *A General Comparison of FISMA, HIPAA, ISO 27000 and PCI-DSS Standards*, published in the *Information Security Journal: A Global Perspective*, volume 19, issue 3, pages 132-141, doi: 10.1080/19393551003657019.

The best place to start to figure out how to leverage your other compliance efforts is to set up a meeting with the team leaders from those projects. You need to get an idea of how the project performed and how it was accepted by the management. The main point is to find out what the other teams have done in their compliance effort

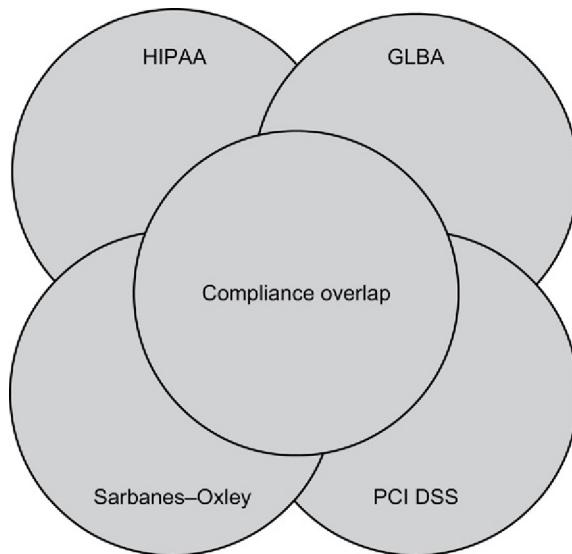


FIGURE 14.1 Leveraging Compliance Overlap

and see what elements you can bring over into your PCI DSS compliance plan. For example, HIPAA and PCI DSS both have rules regarding encrypting data. Can you use your encryption policy and procedure from HIPAA for PCI DSS compliance? That answer will come as you talk to your HIPAA compliance team leaders and review the policy and procedure to see if it already fits the PCI DSS encryption requirements found in Requirements 3.4–3.6. Your company policy for HIPAA compliance should mandate that you have encryption in place as you transmit protected health information across public networks like the Internet. PCI DSS Requirement 4 mandates the encryption of cardholder data as it moves across public networks. In this case, you do not need to recreate the wheel; you might just need to reclassify what type of data is required to be encrypted. Any efforts spent in leveraging your existing regulatory compliance will help to shorten the time it will take for you to become PCI DSS compliant.

NOTE

To help your organization determine how many new policies and procedures you will have to put in place to become PCI DSS compliant, consider completing Self-Assessment Questionnaire D (SAQ-D) in the early part of planning your compliance program. SAQ-D is a good tool to help demonstrate what compliance you already have and can show what you need to do to become compliant. SAQ-D can be downloaded from the PCI Security Standards Council (PCI SSC) Web site at www.pcisecuritystandards.org/saq. SAQ-D is the most complete questionnaire, and may not be the exact one your organization follows. Review the guide available on the PCI SSC Web site to determine the appropriate one for your organization.

THE LEVEL OF VALIDATION

Now that you are on your way to planning your PCI DSS compliance project, you need to figure out at which level you need to validate. Unlike some regulations that present you with an all or nothing stance on how to validate your compliance, PCI DSS validation levels are based on how many credit card transactions a merchant processes or a service provider processes/stores as well as on other related items you already read in Chapter 3. The more transactions that are processed, the more validation activities you may have to perform.

WARNING

Don't let yourself become complacent. If you are a Level 4 merchant across the board and are not required to do anything to validate your compliance with PCI DSS, remember this: by accepting even one card per year, you are still *required to comply with PCI DSS*. Many Level 4 merchants end up in big trouble when they realize they had to comply with PCI DSS regardless of their validation requirements.

For most organizations, validation consists of passing quarterly security network scans and completing an SAQ. If you process transactions in the millions, you need to have a QSA (or ISA if a merchant) validate your PCI DSS compliance through an on-site assessment. Remember, the five individual card brands set the enforcement requirements for PCI DSS, not the Council. It is possible (by volume, but level reciprocity could make the merchant a Level 1 across the board) to be a Level 1 American Express merchant, a Level 2 MasterCard merchant, and Level 3 Visa Merchant all at the same time! To help you determine your merchant or service provider level, you can review the information in Chapter 3. Keep in mind, payment brands frequently alter their programs, so before you go too far into this process, visit the links in Chapter 3 to get the most recent levels from each of the card brands. Or better yet, call your acquirer(s) and ask them to help you determine your level.

WHAT IS THE COST FOR NONCOMPLIANCE?

One question that should be answered during your justification process is: “What is the cost for not complying with PCI DSS?” In all cases, the costs (assuming the worst) associated with a breach far outweigh the costs of being compliant. Can your organization afford the fines and penalties, bad media press, and damage to its reputation? Breaches cost more during recessions as cash reserves are eliminated from companies face dwindling or flat growth.

Your risk managers will tell you that the three things you can do with a risk is to resolve the issue, transfer the risk, or ignore the risk. The way PCI DSS states its 12 requirements—the only way to truly deal with the elements—is to resolve the issue or transfer the risk. Transferring the risk might mean that you outsource or bring in a managed service to deal with that requirement. Therefore, when you transfer the risk, you are still dealing with it indirectly. Ignoring the risk in PCI DSS is not an option. Even one noncompliant item in a Report on Compliance (ROC) or SAQ means you do not comply with PCI DSS. If you are compromised and found to be not compliant with PCI during the investigation, the fines are steep.

NOTE

Breach fines are calculated in different ways depending on the situation. Visa’s Account Data Compromise Recovery (<http://usa.visa.com/merchants/operations/adcr.html>) program describes how Visa fines compromised merchants. They calculate fines based on the number and type of records lost. Your acquirer may be able to give you more information if you are interested in how the formula is calculated. MasterCard now has a similar program, and your acquirer will have those details as well.

Penalties for noncompliance

Did you know that every quarter (if you are a Level 1–3 Visa or MasterCard merchant) your acquiring bank is telling Visa and MasterCard if you have validated

compliance? Both card brands are now actively fining merchants for noncompliance, but the amounts vary widely depending on your level. Let's take a minute to review the current fines that merchants of Level 1–3 can expect to receive.

Visa Compliance Acceleration Program (CAP) Fines:

- Level 1 merchants:
 - \$25K each month for noncompliance (\$300K/year).
 - Tiered interchange penalties, meaning that every transaction will cost you slightly more to process, potentially costing companies millions.
- Level 2 merchants:
 - \$5K each month for noncompliance (\$60K/year).

MasterCard fines:

- Level 1 and 2 merchants:
 - Quarterly escalating fines of \$25K, \$50K, \$100K, and \$200K (\$375K/year).
 - Resets to \$25K on the quarter following the \$200K fine.
- Level 3 merchants:
 - Quarterly escalating fines of \$10K, \$20K, \$40K, and \$80K (\$150K/year).
 - Resets to \$10K on the quarter following the \$80K fine.

While you may never find either of these programs detailed on the card brand's respective Web sites (Visa: www.visa.com/cisp/, MasterCard: www.mastercard.com/sdp/), your acquiring bank will have all the information relevant to your situation. Reach out to your acquirer first, then use the figures here (or the ones provided by your acquirer) to assist in your cost analysis. This information was obtained through previous online publications and through customer relationships, and are subject to change at any time. The fines are global, but you may see differing enforcement of the fines as well as differing monetary amounts. These amounts are based in US Dollars, and could be converted to your home currency for a baseline. Be careful when taking these figures to management. Validate your localization to ensure you don't derail your compliance initiative before it starts. Remember, fines come from both Visa and MasterCard now.

If you suffer a breach and your organization is found to be out of compliance with PCI DSS, the penalties can be severe. Theoretically, the organization could be forbidden to store, process, or transmit credit card information. A more likely result would be stiff penalties from the card brands used to recoup the costs associated with fraud and becoming a Level 1 merchant for one reporting period, thus dramatically increasing your compliance costs. Each case is handled individually on its merits. With the advent of new privacy laws in different states, you might be required to notify your customers of a breach and provide them with credit reporting services. Once notifications go out, you will lose control of the message. Looking at what it takes to comply, it should be easy to see how and why you need to put together your PCI DSS compliance plan.

BRINGING THE KEY PLAYERS TO THE TABLE

Once you have justified your compliance effort, it is vital that you bring all the players to the table to ensure that you are successful in becoming compliant. You need the correct corporate sponsorship; otherwise, senior management could reject any plan you put together. You need to look at your organization from the top down and identify each of the key people who are necessary to put the plan together, forming your compliance team. You need to identify the key members of your team to tackle components of the compliance plan and keep the project moving.

Compliance plans can be won or lost based on the participants you bring in to help you with the project. It is vital to bring the correct people to the table. Be selective of your team, as they will make putting together the compliance plan either a success or a failure. Remember what noncompliance can bring—failure is not an option.

WARNING

Be sure to get a good understanding of the current workload of the members you would like to invite to be a part of your compliance team. Many times, people are enthusiastic to be a part of a new project, but realistically they do not have the time to work on it. At the end, team members miss meetings or deadlines, which may impact deadlines associated with your compliance project.

If you are having trouble visualizing your team's work, consider getting a copy of *Personal Kanban: Mapping Work | Navigating Life* by Jim Benson and Tonianne DeMaria Barry (CreateSpace, ISBN-13: 978-1453802267) and using a kanban board to organize this project.

OBTAINING CORPORATE SPONSORSHIP

Management sponsorship is a critical success factor for any compliance effort. If senior management does not support the process, support from the staff will also be lacking. Why should they comply if your manager does not? As the leader of your compliance effort, you need to first work with your senior managers to make them aware of the issues and let them understand the justification of why they need to comply with PCI DSS. Make them understand the cost of noncompliance, and they will back you up as soon as they realize that the company could be in jeopardy for not complying. Start at the top, because the earlier you gain support from the Chief Executive Officer (CEO), the faster you will get support from the Vice President and other senior management.

Derek, our fantastic technical editor, contributed this very important tidbit. Without the business bought in and involved in the efforts, you will just end up with a bunch of IT controls that are waiting to be broken. It's not just about controls, it's about a culture change in many cases. What good does a password policy do if the culture allows users to write it down on a sticky note and attach it to the bottom of their keyboard?

NOTE

Try to schedule a lunch meeting outside the office with the company CEO or other senior manager, where you would have his or her full attention, devoid of any distractions. Help him or her to understand the cost of noncompliance. Don't treat it like a stick to beat the CEO over the head with—they have plenty of that happening every day. FUD will get you nowhere fast, and the impression you make during these initial meetings will set the tone for the success of your program. Instead, learn about the business, the strategy, the plans, and figure out how to enable the CEO to get there without the burden of PCI Compliance. Executives manage risk every day—some better than others. Don't get kicked off of the adult table to be doomed to spend the rest of your career at the kiddie table because you cried wolf a bit too loudly.

When other employees in the company hear that a senior manager is sponsoring the team, the entire project will get more support which will help drive home the fact that the compliance effort is vital for the organization.

FORMING YOUR COMPLIANCE TEAM

Your compliance team is the focal point of your compliance project and is responsible for the success of the project plan. The best time to create your team is after you have received corporate sponsorship. Many times people who heard about the compliance project from a manager and want to participate will approach you. You need to get a good mix of people on the team to make the most impact. PCI DSS has 12 requirements that can touch different departments in your company, so be sure to include at least one person from each of those functional areas. For example, PCI DSS requires you to build and maintain a secure network; therefore, if you do not get a team member involved from the networking group, you cannot be sure that a firewall is installed or maintained going forward. Don't forget to include the physical security and facilities teams as they help you meet various requirements (including much of Requirement 9) and the business folks when analyzing the various workflows that generate money from customers.

ROLES AND RESPONSIBILITIES OF YOUR TEAM

Your compliance team will help set the pace and scope of your compliance project. The selection of participants will make the project a success, but it is important to make it clear from the beginning what each team member is responsible for by assigning them roles and responsibilities. You will need your team to assist in the following ways:

- Work with managers and other team members to set the scope of the compliance project—mapping credit card data flows is critical and you must include both physical and logical flows,
- Select leaders for each of the areas where you need compliance,
- Analyze information needed for the compliance plan,
- Work with senior management to ensure that the end result is compliance,
- Ensure you properly size and define the cardholder data environment.

GETTING RESULTS FAST

The best way to ensure a successful project and gain the respect from all levels of your organization is to get results fast (or at least score a few quick wins). As you are planning your compliance plan, you need to identify some low-level compliance issues that are relatively easy to fix and have your team tackle those first. People want to see results, and the faster you can show them results, the more confidence they will have in the project. If it takes you months to get the first item addressed, people might wonder if the organization will ever be compliant and become complacent about the effort as a whole, derailing all your efforts up to this point. Getting results early keeps the momentum and support moving in a positive direction for your entire project.

NOTES FROM THE FRONT LINE

To give you a good example of how important it is to select the right team members, here is a real-world story of the first time Karen was on a compliance team.

Karen was approached by her manager, Christina, to help with the company's PCI compliance effort. Christina felt that Karen's knowledge would be an asset to the team. The team leader sent out a meeting request for the 10 team members, and Karen was excited to help make a difference in her organization. She showed up at the first meeting on time, ready to do what was necessary—even if it meant having to put in overtime to get the job done. That first meeting did not go so well. The team leader was 10 min late and only half of the team members showed up for the meeting.

During the meeting, Karen began to realize that none of the other senior managers were briefed on the compliance project, and some even wondered whether they needed to comply with these new regulations. Even though there was no senior management support, the team leader knew the company needed to get into compliance or face trouble. When Karen asked about the missing team members, the team leader thought that it was probably due to the lack of support from upper management.

After weeks of meetings, false starts, and many extra hours, the compliance team finally had senior management involved and then the wheels started to turn. The entire team showed up for a meeting for the first time, but they had to start over from the beginning. Karen and the team leader soon realized that the meeting lacked the right people for the areas need to become compliant.

After a few more weeks, the right people did get involved with the team, and miraculously senior management support was still there. The project took off like a wild fire. Karen's team did a gap analysis and figured out what needed to be fixed and hit the ground running. After months of trying to put the team together, they were able to knock out the entire project in 3 weeks. Just like the expression about needing the right tool for the right job, you definitely need the right team for any compliance project you are attempting to pull off.

BUDGETING TIME AND RESOURCES

In order for your project to be a success, you need to ensure that it is managed correctly and that it does not take too long to complete. As important as it was for your team to get some results early on, you must continue to make sure that you set expectations, goals, and milestones. Figure out early on how you will manage the time and resources of your team and you will have a successful compliance project.

SETTING EXPECTATIONS

Setting expectations is a key factor when budgeting time and resources within your team. From the first stages of your compliance project, your team needs to know what to expect from you, other team members, and management. If this project is a priority one, the team needs to know that all other tasks are secondary until the compliance plan is in place. You also need to be sure you set the right expectations with management, so they know what to expect with the compliance plan. Don't forget the ongoing costs associated with compliance as well. If you approach this as a project, your company will treat it that way. There are significant ongoing costs associated with complying with PCI DSS. Those can include things like hardware refreshes, software licenses, headcount associated with managing portions of your infrastructure, meetings, assessments and audits, and other expenses that you will find as you go along.

MANAGEMENT'S EXPECTATIONS

Knowing from the beginning what management expects out of this effort should be one of your first tasks. Before you bring the team together, you should talk to senior management to make sure you understand what they expect out of the project and the timeline in which the project must be completed. Be sure you understand the criticality of the compliance effort to the organization, as that will help you get a pulse on the project itself.

Once expectations (and the appropriate management sign-off) of the compliance project are in place, you need to document them and share them with all the members of your team. By having all the team members of the compliance project working from the same set of expectations, you are one step closer to having a successful project. If management feels that the project needs to be done in 4 weeks but the team actually needs 8 weeks to complete the tasks, be sure to set the correct expectations.

ESTABLISHING GOALS AND MILESTONES

Once a timeline is in place, it is important to set goals for the team on when key items should be complete. You want to make it very clear when project items are due and when parts of the compliance plan need to be in place.

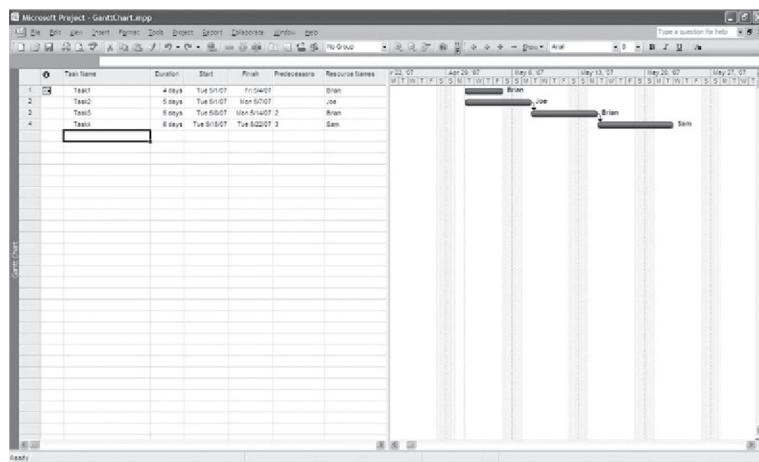


FIGURE 14.2 Example of Gantt Chart

Start by listing the goals of the project and assign those goals to team members. Make it clear when goals need to be met, as some will have prerequisites that must be finished before you can move on to the next task. Having goals in place will keep the project moving in the right direction. Set up milestones for success and publish your plan to everyone involved to keep them up-to-date on the project's status.

A good way to keep your time and resources managed is by using project planning software such as Microsoft Project, which allows you to create Gantt charts that map resources to goals (Figure 14.2). Gantt charts give you a way to easily report on your compliance project. If an item slips or is completed early, the chart will adjust and keep your project in line with the project timeline.

If you don't have Microsoft Project, some open-source equivalents are as follows:

- GanttProject: www.ganttproject.biz/
- OpenProj: openproj.org/openproj
- OpenWorkbench: www.openworkbench.org/
- OmniPlan: www.omnigroup.com/products/omniplan

A Web-based equivalent is:

- Ganter: www.gantter.com/

HAVING STATUS MEETINGS

The key to keeping your project on time is to have weekly (or daily if needed) team status meetings. The meetings should include your compliance team members and each should report on what they have accomplished in the past week and what they will be working on in the next week. These meetings also give team members a

chance to compare notes and bounce ideas off each other if they are stuck on a problem. Be sure to take notes or minutes and assign any new action items that come up. Any items that may impact the project timeline or resource requirements need to be placed into your project plan.

You should also have status update meetings with senior management on a regular basis. Depending on the length of your project, the meetings should be, at a minimum, once per month. During these meetings, you can go over your goals and milestones, and show how the project is progressing. It will also give the senior managers a chance to give their input on the project and reinforce the support you need from them.

Be prepared to hand out copies of your working project plan Gantt chart. It will give a clear picture to your senior management team of where you are in the process and who is working on what issues. It is a good idea to send these charts to the managers beforehand to give them time to review the progress so that they can determine the guidance and support you will need.

If you are leading this team, you must strive for accountability. If deadlines go without recourse, expect the whole project to miss every future deadline you set. One trick Branden used was to send out daily status reports with every senior manager's name listed in a specific color to provide a visual reference to how their area's projects were going. Without fail, any manager who was resisting PCI DSS changed their attitude immediately after seeing their names turned red next to their peers.

EDUCATING STAFF

Training can make or break any compliance project. From the first meeting, ensure there is a training component to make all members aware of how the project will run and make sure they have all the necessary information to move forward with their part of the compliance project. Also, when your compliance program is in place, you need to make sure that part of that program includes training. Actions and plans to meet PCI DSS requirements must be maintained after they have been developed. The only way to do this is through a series of reminders and recurring training classes for your organization's employees. Having a training program in place from day 1 will go a long way in keeping your organization compliant after you have completed your compliance plan.

TRAINING YOUR COMPLIANCE TEAM

When your compliance team meets for the first time, you should review common information for all members. Items should include the following:

- An overview of the PCI DSS,
- An overview of the compliance effort for your organization,

- Why your organization is going through the process,
- A high-level review of the project plan to share goals and milestones,
- A review of any elements the team might be submitting (i.e., how a policy should be written or how often to deliver status reports in some set format).

You could even use this book's Table of Contents as a guide for your training, making sure that you pull out the relevant portions for the teams you are working with.

Training your compliance team will help them understand how the plan came together and how to execute it to make your organization compliant. It will also get all members on the same page about what PCI DSS is and why your organization is going through the effort. You want to remove all myths around the project and level the playing field for your team members, so they can be successful in making your organization compliant.

TRAINING THE COMPANY ON COMPLIANCE

After your project is complete and you deem your organization to be compliant, you need to make sure the rest of the company knows that you need to maintain a level of compliance. You do not want to have a violation in the first week because an employee did not know about the need for compliance.

You need to put together a corporate compliance training program for all new employees and for existing employees to complete annually, which acts as a refresher course and also gives you a chance to present any information that has changed over the past year (Requirement 12.6.1 among others). Consider training anyone who comes into contact with a payment card. If this number seems daunting, try masking or truncating cardholder data for the majority of your users to reduce the number that need focused training. Your IT staff may also need some training to ensure system administrators behave in a compliant fashion. Things like building hardening into the procurement process, weekly FIM checks and acknowledgements, regular antivirus and antimalware agent validations may not be part of your IT staff's DNA. If the new security processes are truly built-in (not bolted-on) then personnel from many IT teams will need to be refreshed on PCI-related processes for in-scope systems.

SETTING UP THE CORPORATE COMPLIANCE TRAINING PROGRAM

Be sure to set up your corporate compliance training program as an element of your compliance plan. Get the human resources department involved early on in the process to make sure that all employees of your organization receive the training. Many times you can leverage existing programs, like your current new employee orientation, to train existing employees.

NOTE

Keep your compliance training program upbeat and fun. Although security might be boring to most of your employees, it is fundamental to the success of your compliance efforts. One idea would be to have prizes at your training classes and offer them to people who get answers right during a question and answer session. People will be more likely to want to attend the training class if they can win a dinner, movies, or a gift card to any number of retail stores.

If you are reading this and laughing at how corny this sounds or thinking “this will never work in my company,” you might be surprised. Both authors have always found it interesting from a human behavior perspective when you find the one or two motivators for influential team members. Others tend to fall in line!

Try this Web site for ideas: <http://www.securingthehuman.org/resources/presentations>

The compliance training program is more than just creating a one-time training class for your employees. The following elements should be incorporated for a successful program:

- Create a new hire training class that all new employees are required to attend. It can be as simple as handing out this book to your new hires, and making sure they understand the content by asking questions about PCI, or as complicated as bringing in a trainer to develop a program for you. The initial training will need to be comprehensive, potentially derived from this book’s Table of Contents. Work with your human resources department to see if this training class can be injected into an existing orientation program, or be sure you are a part of the process, so your training team is notified about new hires.
- Create an internal Web site that outlines key elements from the compliance training, so employees have a good source to review information.
- Create a series of reminders to help keep the compliance effort on the minds of the employees. Good ideas for this are awareness posters, articles in your company’s newsletter, and even “Compliance Days” where you can make a fun event around being PCI DSS compliant.
- Create a recurring annual training program for employees to make sure they are reminded about what they need to do to comply. Recurring training should update your employees on new developments (e.g. in 2009, we saw MasterCard change both validation requirements which were reversed and reinstated later, and add fines), changes in PCI DSS, or covering specific areas that your company struggles with. The recurring training program can work either as a live training class or as a Web-based training class that they can take when time permits. Either way the training is presented, it should be required to keep your organization in compliance.

With the right training programs in place, you can be sure that from the first meeting of your compliance team to the annual recurring training for your associates, your compliance efforts will have a lasting effect on your organization.

NOTE

One of the greatest tools in any compliance awareness program is the use of posters. With the use of posters, you can get the message out quickly. The posters you put out should have simple messages that grab people's attention. For PCI DSS compliance, simple phrases such as, "Ensure your Anti-Virus is Up to Date" or "Keep all Cardholder Data Under Lock and Key" will get the message to your employees quickly. Compliance posters are also a great way to get that first big result. You can create and put these posters up in the first part of your compliance planning efforts to give a kick-start to the project. When senior managers are walking around the office, they will see the posters and know that you are taking the compliance project seriously.

PROJECT QUICKSTART GUIDE

Putting a compliance plan together for PCI DSS can seem like an overwhelming task. You are probably asking yourself where to start. Who should be involved? When do you look at the PCI DSS SAQ? This section will get you pointed in the right direction and give you the first step toward getting your organization compliant with PCI DSS.

THE STEPS

We know how to plan a project to meet compliance, but when it comes to PCI DSS, what are the specifics you should be looking at to become compliant quickly and efficiently? For an overview of the steps, see [Figure 14.3](#).

Step 1: Obtain corporate sponsorship

Once you have corporate sponsorship, you will have the backing for all the steps of your compliance project plan. Be sure to meet with these members of your organization first to get the sign-off and acceptance that your company needs to be PCI DSS compliant. Remember, you need to make sure you get support from the highest level possible in your organization. Getting the backing from senior managers will help to ensure that the rest of the employees will be willing to work with you on getting compliant with PCI DSS.

Step 2: Identify and establish your team

This is a critical step because it could make or break your compliance project. You need to be sure to select your team members from the appropriate areas of your company. Include the business leaders that have to worry about PCI DSS compliance and also the techies in the trenches who are setting up your networks. Having a good mix of key players will help your project succeed.

You should choose leaders for each of the 12 requirements of PCI DSS. If you break PCI DSS up into each requirement, you will be in a better position to complete your effort in a timely and concise manner. You should also set up a training class during your first team meeting to review what PCI DSS is, why your company has to comply, and the initial plan of what needs to be done to get into compliance.

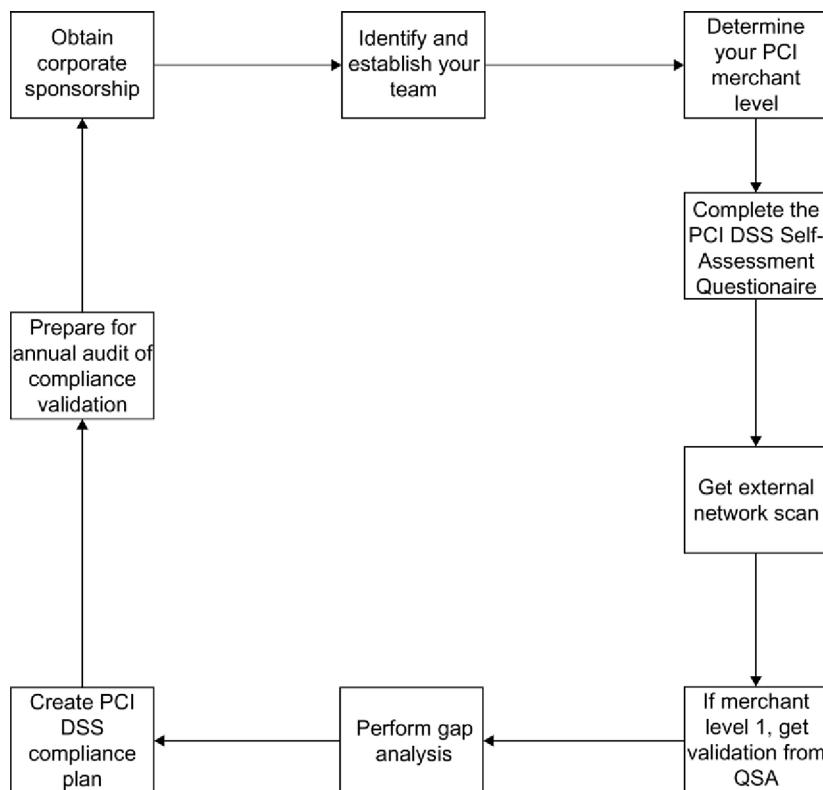


FIGURE 14.3 Steps to PCI DSS Compliance

Step 3: Determine your PCI level

You need to know what your PCI merchant or service provider level is, which will tell you how you need to validate compliance with PCI DSS. Talk with your team members who are from the business side and figure out how many transactions you perform. Or, if you are a merchant, call your acquirer(s). Then, refer to the card brand's Web sites to help you figure out your organization's level.

Knowing your level will set the stage for what exactly you need to do to comply as each level has different requirements for validating compliance. It is important that you determine this early on in the process because as you get closer to Level 1 (or Level 2 for MasterCard merchants), your compliance effort will take longer and involve more resources. If you are not at Level 1 from the start, you want to periodically review how many transactions you are processing—especially if you are on the border. If you jump to another level, you may also jump out of compliance.

Step 4: Complete a PCI DSS SAQ

You need to complete the SAQ most appropriate for your business (many will use SAQ-D) in one of your first compliance meetings because the results of the questionnaire will give you clear guidance on how compliant your organization already is or is not with PCI DSS. The questionnaire can be found at the PCI SSC Web site: www.pcisecuritystandards.org/saq/. If you answer “No” to any of the questions, you are not in compliance. The questions on the questionnaire map directly to the requirements of the PCI DSS. When your organization has the questionnaire complete, it will indicate not only if you are compliant with PCI DSS, but what you need to do to become compliant. Even if you must validate as a Level 1, this exercise can provide a quick baseline for you.

Step 5: Set up quarterly external network scans from an approved scanning vendor

Compliance with PCI DSS requires a quarterly network scan from an Approved Scanning Vendor (ASV), but Level 4 Visa, MasterCard, and Discover merchants may not have to submit their scans to their acquirer. All externally exposed Internet Protocol (IP) addresses must be scanned for vulnerabilities by an ASV, which means performing your own external scans will not make you compliant. The PCI SSC maintains a list of ASVs at https://www.pcisecuritystandards.org/approved_companies_providers/index.php. For more information, see Chapter 9.

At the end of the network scan, the ASV is required to provide you with a report that will show you if your Internet-facing network is PCI DSS compliant. If they discover a vulnerability of a high enough severity, they will typically point you in the right direction toward a remedy.

One quick tip on your scans: do them monthly. Sometimes it can take a month to coordinate fixes.

WARNING

You must select your ASV from the list that is maintained by the PCI SSC. If you do not use an approved vendor, any results you have, no matter how good they appear to you or your organization, can invalidate your PCI compliance efforts. Remember that you must have clean, quarterly external scans, except for your initial PCI DSS compliance where all you need is a recent, passing scan and documented policies and procedures requiring quarterly scans. Submitting scans with vulnerabilities that must be fixed for compliance proves you are not compliant with PCI DSS.

Step 6: Get validated by a QSA (or an ISA)

This step is only required if you are at a merchant level that requires this (currently, Levels 1 and 2 in some cases) or your acquiring bank mandates it. You can also send some of your own employees to Internal Security Assessor (ISA) training, and perform some activities without the use of a QSA. You want to engage the ISA or QSA to help you with Step 7 below. PCI Assessments are an annual process, where

all components that are a part of how your company stores, processes, and transmits cardholder data are assessed. You can find a list of QSAs at www.pcisecuritystandards.org/pdfs/pci_qsa_list.pdf. Keep in mind, Level 1 merchants are NOT REQUIRED to use a QSA, and in fact can self-assess after having an employee complete the ISA program. Many companies opt to go the QSA route, but remember that hiring a QSA does not transfer your liability to that QSA if you have a breach. You are still ultimately responsible for your ongoing compliance.

Step 7: Perform a gap analysis

After your team has gone through the SAQ, the network scan results, and potentially the reports from your QSA including a ROC, you now must prepare a document that lists out the gaps in your compliance. Your gap analysis document will set the stage for the creation of your compliance plan. To assist with your gap analysis, you should put together a worksheet that lists each requirement and indicates whether you are compliant or not. You can also use the worksheet to initially assign the requirement to a compliance team member.

Part of your plan should include building out some very basic documentation such as a data flow map for all of your card data to a list of in-scope systems. For those of you that must validate as a Level 1, you will already be required to do this. For those of you who are filling out a SAQ, this is required both for compliance and to drive your compliance process.

Step 8: Create PCI DSS compliance plan

Following the steps above, you now have the steps needed to create your PCI compliance plan. As discussed throughout this chapter, you should take all these elements and bring them into your compliance plan. Your plan should include the gaps that are standing in the way of your PCI DSS compliance efforts and what your organization plans to do to stay compliant year after year. Once all the gaps are closed, your compliance plan will be the live document that ensures you stay compliant with PCI DSS. If you need guidance on which gaps to tackle first, consider looking at the Prioritized Approach for PCI DSS v3.0 and the accompanying tool (found here: www.pcisecuritystandards.org/security_standards/documents.php).

Step 9: Prepare for annual assessment of compliance validation

To maintain compliance, you should start over at Step 1 and begin the process again every year. The good news is that most of what you need to do is already complete, and you are mainly validating that you are still PCI DSS compliant.

THE PCI DSS PRIORITIZED APPROACH

For those companies that feel lost among the mountain of remediation that needs to be done, the PCI Council may have some help for you. In early 2009, the Council released a Prioritized Approach for PCI DSS (www.pcisecuritystandards.org/security_standards/documents.php) [1] which was updated to version 3.0 in June 2014. While

the approach provides guidance to those individuals responsible for steering a PCI compliance project to completion, it needs to be customized for each organization to most efficiently meet your needs. At the above URL, there are two documents available for download—a written document outlining the approach and an Excel spreadsheet to help manage your process.

NOTE

How can you use the PCI Prioritized Approach to make PCI DSS easy for you?

Use the document to plan your PCI project from current state to compliant and secure state.

Use the Excel spreadsheet for ongoing planning of the next steps and identifying weak areas/next area to handle.

Use the Excel spreadsheet to track status and create a report of compliance status for senior managers.

The PDF document describes the approach and gives some background on why the approach was created, it describes objectives, and it outlines the six milestones in their plan. The other document is a Microsoft Excel spreadsheet that contains the entire PCI DSS with a milestone number next to each requirement. Most companies that use this tool will add more columns to it to bring in their assessment data and will change the milestones to be in line with their particular project milestones. If you have no milestones defined, use these as a reference. Remember, you will probably need to adjust the milestones to fit more appropriately into your company's current compliance plan.

THE VISA TIP

While the Technology Innovation Program (TIP) is only from one of the payment brands, we felt it necessary to cover it here in this chapter as it is potentially a nice shortcut to obtaining compliance. One of the loudest complaints we hear from folks who feel that PCI DSS is forced upon them goes something like this: "If the payment system would just be secure, we wouldn't need this thing!" Religious wars of thought aside, everyone shares blame here. A secure payment system would certainly help, but so would secure enterprises. Visa recognizes this, and is now pushing for EMV card processing here in the US. Globally, Visa recognizes companies who have invested in the use and deployment of EMV, and reward them by giving them shortcuts to compliance.

As of this writing, the Visa TIP allows companies to bypass certain annual reporting requirements by meeting a certain minimum set of standards. Unlike the CAP which used fines and interchange penalties to motivate merchants to comply with PCI DSS, there is no true financial incentive to participate in the TIP today. The closest resemblance to a financial incentive is the domestic and cross-border counterfeit liability shift. For more information on this, contact your acquirer. Essentially,

merchants that cannot accept an EMV or contactless card when presented one by a customer will bear the liability of a fraudulent transaction instead of the issuer after October 1, 2015.

In order to qualify for the benefits of the program, merchants must meet the following baseline:

- Validate PCI DSS compliance within the previous 12 months or have submitted to Visa (via their acquirer) a defined remediation plan for achieving compliance, based on a gap analysis.
- Confirm that sensitive authentication data (i.e., full contents of magnetic stripe, CVV2 and/or PIN data) is not stored, as defined in the PCI DSS.
- The merchant must not be involved in a breach of cardholder data. A breached merchant may qualify for TIP if they have subsequently validated PCI DSS compliance.
- 75% of your transactions must *originate* from “*enabled devices*.” An enabled device under this program means that your payment terminal must accept and process an EMV or contactless card (not just be capable of doing this). Note that the transaction itself does not need to be EMV or contactless, but the terminal must be capable of processing both of those payment types. Merchants that have upgraded their terminals with a slot for EMV and a contactless reader but cannot accept either in exchange for goods and services do not meet this requirement.

The TIP does not address card-not-present transactions but that transaction volume is included in the numbers used to determine the 75% qualifying level. If this is an issue, consider any number of strategies that could remove those transactions the counts such as outsourcing card-not-present processing or creating a separate merchant relationship for your online store.

Merchants that choose to take advantage of the TIP may in fact be able to talk their acquirers or processors out of validating compliance entirely. This obviously would depend on the relationship you have with your providers as well as the way you process payments, but most acquirers are only required to report compliance back to the payment brand—not send along the ROC or Self-Assessment Questionnaire. Therefore, participating in the TIP could ultimately remove your annual PCI Assessment requirement if you convince your acquirer to report your compliance in lieu of an ROC if you participate in the TIP.

The three main outcomes that Visa is shooting for as a result of these programs are as follows:

- Bring the backend processing technology up to more secure levels as is seen in other parts of the world as dynamic authentication will reduce the amount of card present and counterfeiting fraud.
- Ensure cardholders that travel internationally can use the Visa card in countries where EMV is prevalent. Some cardholders have had trouble recently because their US-based issuers have not provided them with EMV capable cards.
- Push merchants to accept dynamic data methods for payment by excusing them from liability in the case of a domestic or cross-border counterfeiting fraud.

SUMMARY

Planning a project to meet compliance can be so overwhelming that you wind up having false starts or not begin the project at all. Your compliance efforts do not have to end this way. By putting together a good compliance project plan, you will have what it takes to make your organization PCI DSS compliant.

From the start of your project, you need to take a close look at why you need to become PCI DSS compliant. Simply figuring out if you need to comply can save you weeks of time and effort that could be devoted to other compliance initiatives. Once you determine that you must comply, spend time understanding your current level, type of company (merchant or service provider), and what exactly you have to do to validate your compliance. Once you know your level, either you will have a QSA perform an assessment against your company or you will fill out one of the nine SAQs relevant to your business model. The Council's Web site has information on all nine SAQs, and what must be submitted for compliance validation purposes (www.pcisecuritystandards.org/saq/). You also need to figure out what is the cost to your organization for noncompliance. Can your organization afford the risk? With new legislation and fines coming down pike, nearly all situations will yield a firm "No."

Once you determine that you need to be PCI compliant and cannot afford the risk of noncompliance, you need to bring all the players to the table. You first want to obtain senior manager (C-Level) sponsorship to get the backing you need to complete the project. The corporate sponsorship process will also help you form your compliance team. Your compliance project starts by getting your team together and working through the planning process.

You must guide your team in the right direction and help them budget their time and resources effectively. First, you need to set expectations with your team and management about what the compliance effort entails. At this point, you can set up goals and milestones to help keep the project on a timeline and define when the project should be completed. It is important to have status meetings with your team and management during the process to keep everyone informed and moving forward.

As you start your compliance planning project, make sure that your team members get the correct training by providing an overview of what PCI DSS is and why your organization is going through this compliance effort. You should also train all the employees in your company, so they know what it takes to be compliant and to stay compliant. Setting up a corporate compliance training program will have a lasting effect on your organization—not only in keeping PCI compliant but also keeping your workforce thinking about security at all times.

Then, we outlined the nine steps you should take to become PCI DSS compliant. If you go through each of these steps, you will complete the first round of your compliance effort. Knowing that you are PCI compliant will help allay the fears of noncompliance by management. If you find yourself still needing a place to start, you can try out the new PCI SSC Prioritized Approach to compliance. Remember, the information provided may not directly apply to your organization, so you must customize it to make it effective.

At the end of your compliance effort, congratulate the team and encourage them to continue to keep your organization PCI DSS compliant.

REFERENCE

- [1] PCI Security Standards Council. Prioritized Approach for DSS 3.0. <https://www.pcisecuritystandards.org/documents/Prioritized_Approach_for_PCI_DSS_v3_.pdf>.

Don't fear the assessor

15

INFORMATION IN THIS CHAPTER:

- Remember, assessors are there to help
- Dealing with assessors' mistakes
- Planning for remediation
- Planning for reassessing

The title of this chapter might shock you a little bit. Why? Have you noticed that the words “audit” and “auditor” in reference to Payment Card Industry Data Security Standard (PCI DSS) are copiously missing from this book? That’s because the correct terms are “assessment” and “assessor” when referring to PCI DSS. While your QSA may be a CPA, it is not a requirement, and most QSAs are not; instead many come from IT security domain; some may come from IT audit. The procedures an assessor uses to validate your compliance with PCI DSS are called the Security Assessment Procedures (not the Auditing Procedures). It’s amazing what the change of a word will do to get you a more complete assessment. Imagine if your Internal Audit group changed their name to the Primary Assessment Group (or even to “actually useful assessment group”), and everyone changed their title to Assessor.

Sure, it’s a psychology trick, but part of the goal here is to use the right terminology with sound advice to follow it. Your internal audit group should be involved in the PCI DSS program from a self-assessment perspective, but remember, PCI DSS is assessed, and you work with assessors.

Whether it’s your first on-site assessment or your first vulnerability scan, it’s pretty easy to find gaps to compliance. And while this may not be the case for you, you should have a plan in place to deal with this if it happens. This may happen because you interpreted a requirement slightly different from an assessor, or it may be that you simply missed something that an experienced assessor would catch. When things go wrong, it’s easy to blame the assessor. Having the right attitude can make all the difference.

Generally, assessors should not allow you an easy pass unless your environment is truly one of the best managed and secure. If they do not properly report real gaps in your PCI DSS compliance they can lose their QSA status—or go into the dreaded PCI Council remediation process where they are closely monitored by the Council. In addition, your company will be dragged through the mud after the breach as yet another example showing how ignoring PCI and only paying lip service to it will get you not only breached, but also fined by the card brands.

REMEMBER, ASSESSORS ARE THERE TO HELP

When dealing with on-site assessors or approved scanning vendors (ASVs), most people fit into one of three groups.

1. Some people are intimidated by assessors. They see assessors as people with a lot of power, and they hope they will say and do the right things, avoiding the pain of a gap.
2. Some look at assessors as their enemy. They believe they must wrestle with the assessor and hopefully win in the end (this is where the “auditor–auditee” mindset rears its ugly head).
3. Some people treat the assessor like a consultant (“a mandatory consultant” as the case may be) they’ve brought in to help bring their company into compliance. They respect the assessor’s opinions and keep the assessor in the loop as they work out solutions.

While it might surprise you, the last group will get the most out of their assessor and will have the best overall experience—and likely the most secure cardholder data environment as well. They will quickly be able to bring their company into compliance with the least amount of hassle, and will usually gain quick upper management support for funding required to resolve the issues identified during the assessment.

As hard as it might be to believe, assessors are there to help you. After all, you are the one paying for their services. It’s important to know how to work well with assessors so that your assessment will go smoothly and efficiently, and ensure that you get your money’s worth. A good assessor will go over your company’s systems, practices, and policies with a fine-toothed comb, and tell you what you can do to improve your security. Hopefully, your primary goal in becoming PCI compliant is to have your company become more secure and decrease the likelihood of the card data theft or loss. When you realize that assessors provide you with a valuable service and that you’re both on the same team working towards a common goal, you will have the right attitude. Remember that assessors have moral and professional obligations to follow the guidelines and procedures they’ve been given for the assessment, despite the fact that you are paying for their services. It is not appropriate to ask them to compromise those obligations, just for that reason. The assessor’s integrity is more important to them than your fee. Assessors are trained and likely have performed many assessments, and they can give you great advice on what you can do to bring yourself into compliance.

When you choose your assessor, interview them! This assessment is probably one of the most important security projects you will embark on this year. Getting out of the gates with the wrong assessor will make your pain so much greater. This is where having a person with an Internal Security Assessor (ISA) certification or a former QSA who knows how assessors are trained and how they operate helps a lot.

First off, set up an interview with someone from your prospective assessor. Understand their methodology, and how they will strive to ensure you are not one of the companies that exited their assessment process thinking they were compliant,

only to find out (sometimes DAYS later!) that they had been breached, and their assessor did a poor job (sadly, there are some examples of this very event, which are well-covered in the media). Good assessors will bring a team of at least two QSAs to every engagement to make sure you get the most accurate result. Good assessors will come on-site and will not just interact with you via e-mail. Have you ever tried to explain to someone how to build a Lego Millennium Falcon over the phone or, worse, e-mail? Of course not. So don't expect your assessor to get a good view of your infrastructure over the phone, either. Finally, you get what you pay for! You don't always need to choose the most expensive bid, but if you get bids of \$20K and \$200K from the exact same scope, something is not right. Don't get stuck in an apples-to-oranges comparison. You will end up with prune juice every time! In many cases, the cheapest assessor will end up being the least competent who may in fact not even look (or look—but then overlook) at whole parts of the network and control environments.

When you have the right attitude you will find ways to use your assessor to improve the security of your company. Seasoned assessors have a wealth of knowledge, even outside of payment security, and can be leveraged when bridging gaps in compliance. They have seen many technologies, policies, and practices others have put into place to mitigate risks, and should be able to give you choices to help you meet requirements that work best for your situation. For example, if cost is your main concern, an assessor may know of a low-cost or open-source tool that you can use to comply with certain requirements (the QSA may be efficient, "cost effective" and "cost-smart"—or "cheap" and cut corners everywhere).

On the contrary if time is more important, the assessor may know of a solution that is quick to set up that will bring you into compliance. As you work on your remediation, it's important to keep your assessor in the loop. This way he can give opinions on what you've chosen to do and can give further advice. It will also likely make your next assessment much easier for both parties involved.

Don't forget the old business adage... Pick any two of the following when asking someone to provide you with a good or service: good, fast, or cheap! For QSA services, picking "cheap" with either of the remaining two often leads to spectacular (and costly!) trouble later.

BALANCING REMEDIATION NEEDS

Do your homework when looking at ways to bridge compliance gaps. Depending on the problem you're trying to solve, there may be open-source tools, managed solutions, off-the-shelf software, or hardware appliances to consider. When looking at products and services that can help bring you into compliance, there are usually four main factors that you should consider.

- Effectiveness: Will the solution you're looking at really solve the problem and allow you to pass your next assessment? If it won't, it should be ignored.
- Cost: Normally cost is a factor in any decision made by a business. Sometimes decisions are based solely on initial cost, but costs of maintaining should also

be considered. While one product or service is cheaper upfront, it may end up costing your organization much more in the long run.

- Time to install: You probably want to get into compliance quickly. If you're not in compliance you may be facing fines, but will definitely have gaps in your security begging for a hacker to exploit.
- Time to maintain: Many times, the time used to maintain a product will be the most expensive part of adding it to your organization. It may end up that a solution you choose will be more expensive in the long run, because it takes a lot of time to maintain (i.e., purchasing a log management solution instead of outsourcing the management to a managed service provider).

Depending on your exact situation, some of these may be more important than others, but they should all be considered when choosing a solution.

HOW FAIL == WIN

In some cases, failing an assessment ends up being a huge win for the security of your company, and not even for just payment security. In many organizations, the security staff (or security minded IT staff) would like to put certain security measures in place, but have been blocked by upper management because of cost or the notorious "other priorities." Remember, upper management's job is to help the company make money, not spend money. Even after you have done a careful cost–benefit analysis and have determined that the benefits outweigh the costs, upper management may still say "no." A failed assessment may be the perfect time to help them to say "yes." If the assessor is requiring that you add something to comply with PCI DSS, you can use that as leverage with upper management to get it put into place. Again, submit a cost–benefit analysis, adding the cost of noncompliance to the total cost. Let them know that the assessor says you will not be compliant without that measure.

DEALING WITH ASSESSORS' MISTAKES

Assessors are human. Humans make mistakes. Thus, assessors make mistakes. While this does not happen often, there is a right way to deal with it when it does. The first thing to do is to talk to the assessor and have him explain how he came to his or her conclusion. Many times someone misunderstood a requirement or believed a compensating control mitigated a problem, but the assessor doesn't agree. Having good open dialogue about what you believe is a mistake, will often solve the problem quickly.

Many assessors find their roots in security (some find it in auditing, which can, on occasion, make this more difficult). Sometimes an assessor will "make up" a requirement because it just makes good sense, but when you ask the assessor to show you where in the PCI DSS that requirement exists, they will realize their mistake. Notice how that last sentence was phrased. Ask them to "show you where the requirement is in PCI DSS," don't ask them to "PROVE IT!"

Before you go to the step in the next paragraph, consider why you think the assessor made a mistake. Is it because you personally have an attachment to a particular system or control? Was the assessor rude? Have you tried to fix the issue the assessor identified with no success? Or, was he trying to sell you something that his company happens to offer? Assessors make mistakes, but don't assume that because an assessor feels a certain way that he is alone on an island where no other assessor would dare sail. Before pushing back on the assessor, ask **yourself** this: "If I were breached tomorrow, could this be a cause?" If it is, don't waste time bullying the assessor. Take some time to research and look up the issue; don't argue because you happen to dislike his advice or don't want to deploy a particular technology or change a certain "bad habit" of your organization. Use that energy to fix the issue and close the issue that an attacker may use to break into your system.

On the contrary, if you know that this does not increase your risk and do not seem to be expressly spelled out in PCI DSS or supplemental guidance documents, maybe the argument should be continued and you need to push back further.

NOTE

You may feel like you have compensating controls in place to solve a problem but the assessor doesn't agree. If the assessor does not agree with the control and you are a merchant, try working with your acquirer (sorry service providers, you do not have this option). If your acquirer chooses to accept the risk, they can absolutely do so. Most acquirers will side with an assessor; however, be sure if you want to go down this route that you present a good case. Most of the time it's easier to follow the requirement exactly than to try to get a mitigating control to fix the problem.

In some cases, you may need to push back on your assessor. Pushing-back is when you challenge an assessor's results. When you push back, be polite. Simply explain to the assessor your point of view and why you believe there was a mistake. If the assessor disagrees, ask him to explain his reasoning. If the assessor has explained why you didn't pass and you don't agree with his reasoning, you may need to talk to his manager or a practice lead about the situation. Explain your situation to the manager and why you think a mistake was made. Most of the time, the manager will talk to the assessor to get his side of the story before coming to any conclusions. If the assessor's manager agrees with the assessor, you will need to fix the problem to be validated as compliant.

Sometimes an ASV scanning tool will report a "false-positive", nowadays an uncommon but entirely possible event, especially if unusual systems are scanned. This is when an assessment shows you have a vulnerability such as a missing patch or vulnerable system that really is not there. This seems to happen more with remote scans, since they have less access to systems. Any good assessor or ASV knows how to keep false positives to a minimum. When you do get a false positive, your ASV should be able to work through it with you; many have a mature, automated process for dealing with scan mistakes and other false positives. They may want to get more

details from you so they can verify it as a false positive, and then fix the system so the error does not come back in the future.

WARNING

Don't forget, you get what you pay for! Most ASV scanning engines today are based on either Qualys (<http://www.qualys.com/>) or Nessus (<http://www.tenablesecurity.com/>), but not all ASVs will produce the same results. Interview your ASV and ensure they are involved in the process, not just setting you up in a database for scans. You may need help interpreting the results or addressing false positives, and those features may cost more.

Some ASVs only run automated tools with very little human checking. This generally works well most of the time, but sometimes the scans can be complicated, and a false positive may end up in a report. If you get a report listing a serious vulnerability, first act as if it's true and see if there's something you can do to remedy the problem quickly. Please do not start arguing before you have the data: after all, just because you told somebody to "go and patch the vulnerability", but a scanner still shows it does not necessarily mean the scanner is wrong...

Depending on the situation, it may be a good idea to do some tests on your own. You already have a scanning solution on-site for the internal scans mandated by Requirement 11.2, so use it against the target in the external scan report (a good trick if the tools are not the same!). Depending on the type of vulnerability that was reported, you may be able to do some manual testing. For example, if your report says that a patch is missing, you may want to manually check the system to validate the finding. If you are unable to find the vulnerability after your testing, it may be time to challenge your scanning vendor's findings and report it as a false positive. They should do additional tests to determine why the false positive happened, and fix the problem or remove this finding from your final PCI scan report.

PLANNING FOR REMEDIATION

A good rule of thumb when doing remediation is that it should be as transparent as possible, so that it has a minimal impact on the business. Sometimes business or user impact is impossible to avoid. For example, implementing a much stricter password policy or disabling group accounts may have an effect on how people perform their jobs. For the most part, patches and system updates should be transparent to users. The more transparent your remediation, the fewer problems you're likely to have implementing it. As you plan your remediation process, always keep transparency in mind.

The first thing you should do in planning for remediation is review your gap analysis with your assessor. Your gap analysis describes the difference from where you are now to where you should be to be compliant. Ask your assessor which risks

he considers high priority. This is there the part about the QSA being your consultant come in: you are paying him, so ask for his advice! Unlike auditors, a QSA may well share his experience with remediating and analyzing similar problems for other clients or in his past life as a security professional.

For example, if the assessor feels that you have urgent risks that could easily be exploited at any time, you would want to work to remedy those first. In a few cases, an assessor will find a risk that is being actively exploited. In this case, the assessor should let you know as soon as he finds the problem and not wait until the rest of the assessment is done. This would then become your top priority, and you should follow your company's incident response plan. See [Figure 15.1](#) for a visual representation of this process.

Now that you have your results and understand what needs to be done to comply with PCI DSS, it's time to prioritize your risk. With the help of your assessor, work to determine which problem can be exploited easiest and can cause the most damage. These are the ones that should be fixed first. If there are not any "gaping holes," the conversation should turn to the items that you can address that will (1) get you some quick wins, and (2) give you the biggest bang for your compliance buck. There are many tools that can be used to help you classify risks, including the many vulnerability Web sites. Here are some that you might find useful

- Common Vulnerability and Exposures (CVEs): This is the industry standard, and much referred to, listing of vulnerabilities in products. Many products use CVE numbers to reference vulnerabilities (<http://cve.mitre.org/>).
- National Vulnerability Database: Supported by the Department of Homeland Security and has a great database of many types of vulnerabilities (<http://nvd.nist.gov/>) with scored vulnerability severities using CVSS method.
- Open Source Vulnerability Database (OSVDB): This community run database of vulnerabilities will give you a lot of great information on a vulnerability, including references, ways to test your system, and how to mitigate the problem (www.osvdb.org).
- Security Focus Bugtraq: A well-organized site that will give you a lot of information including what versions are affected, an overview of the problem, and examples of exploits. It uses Bugtraq IDs (bids), which are supported in many products (www.securityfocus.com/bid).

FUN WAYS TO USE CVSS

The Common Vulnerability Scoring System (CVSS) is a standard for scoring vulnerabilities that has become more widely used. ASVs were mandated to use CVSS scores instead of PCI scores from June 30, 2007, for any vulnerabilities that have a CVSS score. PCI DSS 3.0 retains the use of CVSS for vulnerability severity (see Requirement 11). Most vulnerability databases, such as NVD, will list CVSS scores, which are great in helping you determine the impact of a vulnerability. There are some vulnerabilities that may not have a CVSS score, but NIST provides a tool to help you estimate them, which can be found at <http://nvd.nist.gov/cvss.cfm?calculator>.

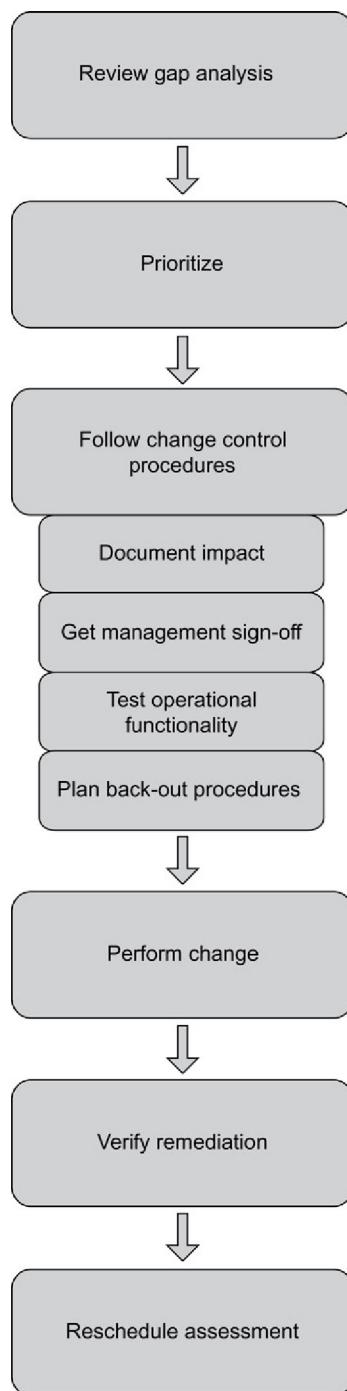


FIGURE 15.1 Remediation Process

For example, let's say that your report shows that you don't have your credit card area physically secured. Since this is not a specific vulnerability with a specific system, there won't be a CVSS score for it, but you can use CVSS to help you determine the priority.

In this example, we'll use a physical security issue to show you how this works. While this system is mainly for computer security issues, it works pretty well for physical vulnerabilities as well.

Jeff is the afternoon manager for Teri's Tapas To Go, a small tapas bar near midtown Manhattan. When Teri built out the location, she found certain constraints as to where electric and telecommunications wiring could be placed. Thus, she has a fax machine near the bathroom to receive faxes containing orders with cardholder data on them. Because the fax machine is not visible by Jeff (or any employee) unless he is in front of the counter, he cannot closely monitor it. There are often times when Teri's staff is busy with customers and are not watching the fax machine. If they are too busy, they may not hear the fax machine and therefore delay checking for new orders. Anyone passing by the bathroom could easily grab a fax.

On the calculator page, Teri would start with the Base Scoring Metrics. This gives CVSS a base score to work off for the vulnerability.

Related exploit range is where an attacker would have to be able to exploit this vulnerability. If an attacker can compromise the system over the Internet or some other remote means, then it would be remote. In our case, with the credit card area not being physically secured properly, it would be **Local**.

Attack complexity is how difficult the attack is to pull off once an attacker has found the vulnerable target. If the attack requires other factors to be in place for it to work, it may make it complex. In our case, we'll say that this is **Low** complexity. Once an attacker knows where the credit card data is, it's easy for them to get to it.

The level of authentication needed is if an attacker must be authenticated to pull off an attack. This means that there is some test to verify who the user is that must be bypassed to attack the system. An example would be something like a fake badge to get access to the fax machine. In this case, an attacker would not because the fax machine is in a public area, so the level will be **Not Required**.

Confidentiality impact describes how the exploit will affect the confidentiality of data in question. In our case, if they can access cardholder data by walking into a protected area and wheeling a file cabinet with all cardholder data in it out the door, it would be complete. Normally a heavy filing cabinet is pretty safe, but since Teri has faxes coming in with cardholder data and there is little to no protection of that data once it hits the fax machine. In this case the confidentiality impact could be **Partial** (as in you are not getting **ALL** of the cardholder data), or **Complete** (as in you did get the complete card number). For illustration purposes, we'll choose **Partial**.

Integrity impact describes how the attack will impact the integrity of data. In our case, it's not likely that integrity will be compromised, so we'll use **None**.

Availability impact describes the measure of how the availability of systems and data is affected. Since the attacker can walk off with a fax, the data is no longer available, so we'll mark that as **Partial**.

The Impact value weighting allows you to give more weight to confidentiality, integrity, or availability. In our case, the biggest problem will be confidentiality, because the attacker just walked off with cardholder data, so we will chose Weight confidentiality. At this point if we click **Update Scores**, we will get a base score of 3.7.

Next we will do the temporal score metrics.

Availability of an exploit lets you determine if an exploit is actually available or not. In our case, we'll say that a functional exploit exists since the attack would work much of the time, but there may be times when one of Teri's employees would catch somebody.

The type of fix available allows us to specify if there is currently any way to remediate the problem. We'll say that Teri has asked employees to keep an eye on the fax machine, which is a **Temporary** fix until she finds a better home for the fax machine.

Level of verification that the vulnerability exists allows us to specify how sure we are the vulnerability is actually present in the system. In our case, we know that the vulnerability exists so we'll choose **Confirmed**.

Finally, the environmental score metrics section. Here we will score the kind of damage that will happen.

Organization specific potential for loss allows you to specify the physical impact the attack could have on your systems. In our case, one credit card number stolen on a fax won't bankrupt Teri, so we'll say it has **Low** (light loss) potential for loss.

The percentage of vulnerable systems allows us to choose how many of our systems are vulnerable to this attack. In Teri's case, this is her only fax machine so we'll say all choose **High** (76–100%).

Now that we're done, we click the **Update Scores** button and get an overall score of 3.9.

There are many ways to prioritize risks—more than we could review in the scope of this book. Don't spend a huge amount of time and effort prioritizing risks, since in the end they all need to be fixed. But it's good to have a general idea in order to start where the fire burns the hottest....

PLANNING FOR REASSESSING

As you are working through your gap assessment, include your assessor! Not only can she give advice on how to mitigate some risks and bring yourself into compliance, she can also help you set realistic completion dates. As you run into roadblocks, she can help you adjust the dates and remediation plan, and be there to support you through the process.

After this is done and everything is in place, plan to reassess yourself. We provide some self-assessment tips in Chapter 15: You're Compliant, Now What? Validate that the gaps are closed to save time and money with your assessor. Provided you included your assessor during the remediation process, your reassessment should be quick and painless. Then you will finally be able to have your PCI DSS compliance party!

SUMMARY

Don't feel bad if your first assessment does not end in a compliant report. Instead, use it to your advantage to better your company's security posture. Work with your assessors instead of against them. Remember you and your assessor are on the same team and the process of assessing should feel like a partnership. By following your assessor's recommendations, your assessment should be less painful and go by quickly. You should involve your assessor as you work to bridge your compliance gaps. The more you involve your assessor, the easier your reassessment will be.

The art of compensating control

16

INFORMATION IN THIS CHAPTER:

- What is a compensating control?
- Where are compensating controls in PCI DSS?
- What a compensating control is not
- Funny controls you didn't design
- How to create a good compensating control
- Case studies

Few payment security professionals can find a hotter Payment Card Industry Data Security Standards (PCI DSS) topic than compensating controls. Ironically enough, this is probably more true today as companies seek to incorporate new technologies that the standard struggles to address. They often look like a mythical compliance accelerator used to push PCI compliance initiatives through completion at a minimal cost to your company with the added bonus of consisting of little or no effort.

Compensating controls are challenging. They often require using a risk-based approach that can vary greatly from one Qualified Security Assessor (QSA) or Internal Security Assessor (ISA) to another (note that QSA and ISA can be used interchangeably for the remainder of this chapter). The Council doesn't prescribe any specific risk-based model or thresholds, so it often defaults to a QSA and acquiring bank to approve and stand behind the control. There is no guarantee that a compensating control accepted today will also work 1 year from now, and the evolution of the standard itself could render a previous control invalid.

The goal of this chapter is to paint a compensating control mural. After reading this chapter, you should know how to create a compensating control, what situations may or may not be appropriate for compensating controls, and what land mines you must avoid as you lean on these controls to achieve compliance with the PCI DSS.

WHAT IS A COMPENSATING CONTROL?

In the early years of the PCI DSS, and even one author's experience under the CISP program, the term compensating control was used to describe everything from a legitimate work-around for a business challenge to a shortcut to compliance. If you are considering a compensating control, you must perform a risk analysis on the

scenario for which the control would be implemented and have a legitimate technological or documented business constraint before you even go to the next step. The best tip we can give you is to remember the word *legitimate* and the phrase *perform a risk analysis* before proceeding to the next step. “Bob” being on vacation is not a legitimate constraint, and a 10-min discussion of the gap and potential control is not a risk analysis. Your QSA should ask for the documentation from your thorough examination of the issue during a compliance review, and having it ready to go will make sure that you are efficiently using their (and your) time. If they do not, you can bet that your assessment isn’t thorough, accurate, or really worth the paper the attestation came on.

If you think that compensating controls are easy, please read the note below.

NOTE

The compensating control polygon has four specific points that must be met. For a compensating control to be valid, it must

- Meet the intent and rigor of the original PCI DSS requirement;
 - Provide a similar level of defense as the original PCI DSS requirement;
 - Be “above and beyond” other PCI DSS requirements (not simply in compliance with other PCI DSS requirements);
 - Be commensurate with the additional risk imposed by not adhering to the PCI DSS requirement.
- Appendix B in the PCI DSS Requirements and Security Assessment Procedures document contains this information as well as some additional notes for those of you tasked with creating said controls. Appendix C contains a worksheet to be used when creating your compensating controls. For an example of a completed compensating control, review the end of Appendix C of the PCI Requirements and Security Assessment Procedures.

An example of a valid control might be using extra logs for the *su* command in UNIX to track actions executed under a shared root password back to an individual administrator. In the fourth edition of this book, this may not be the case anymore. In rare cases, a system may not be able to use something like *sudo* to prevent shared administrator passwords from being used. Keep in mind, this is not a broad license to use shared passwords everywhere in your environment. Nearly every system has the ability to use something like *sudo*, or “Run As” either as a free add-on or as something built into your operating system. In rare cases, you may need to purchase a commercial tool for your platform, but at this point in our information technology history, that should raise red flags about the platform’s ability to meet other requirements that are core to PCI DSS. In those instances, you should look for alternative platforms to support your PCI environment as you will most likely have other PCI DSS challenges.

As stated earlier in this section, before immediately running down the compensating control route you should be sure that you have done your research and make sure that you legitimately meet all of the requirements for a compensating control. Seven to 10 years ago, companies relied on compensating controls because most

platforms did not have readily available solutions to certain components of the PCI DSS. Today's security environment is tremendously different. As a rule of thumb, if the operating system can meet the patching requirements in 6.1 it will probably have everything you need available to comply with PCI DSS.

WHERE ARE COMPENSATING CONTROLS IN PCI DSS?

Compensating controls are not specifically defined inside PCI but are instead defined by you or your QSA. That's where the trouble starts!

Thankfully, the PCI Council provides an example of a completed compensating control at the end of Appendix C of the PCI DSS Security Assessment Procedures, as well as a blank template to fill out. Appendix B provides all the guidance they feel necessary to design a compensating control.

WARNING

Pay close attention to subbullets A to C under list item 3. In our experience, this is where most companies go wrong and end up scrambling around assessment time to re-architect systems to support PCI DSS compliance.

As long as you have met the requirements in Appendix B for a compensating control, you should be able to build that control into your environment and satisfy PCI DSS. Compensating controls are ultimately accepted by acquirers or the card brands themselves (start with your acquirer), so even after putting all of this information together you could face the rejection of your control and a significant amount of expense rearchitecting your process to fit the original control. This is where an experienced QSA can really help you to ensure that your control passes the "Sniff Test." If it smells like a valid control, it probably will pass. If you need examples, look later in this chapter under the section titled "Funny Controls You Didn't Design" and in the case studies.

WHAT A COMPENSATING CONTROL IS NOT

Compensating controls are not a shortcut to compliance. In reality, most compensating controls are actually harder to do and cost more money in the long run than actually fixing or addressing the original issue or vulnerability.

Imagine walking into a meeting with a customer who has an open, flat network, with no encryption anywhere to be found (including on their wireless network, which is not segmented either). Keep in mind, network segmentation is not required by PCI, but it does make compliance easier. Usually in this situation, assessors may find a legacy system that cannot be patched or upgraded, but now becomes in-scope simply

because there is no segmentation in the network. This usually triggers the conversation about compensating controls. Now imagine someone in Internal Audit telling you not to worry about this massive problem with PCI DSS compliance because they would just “go get some compensating controls.” Finally, imagine they tell you this in the same voice and tone as if they were going down to the local drug store to pick up a case of compensating controls on aisle five.

In their original incarnation, compensating controls were never meant to be a permanent solution for a compliance gap. Encryption requirements on large systems were made unreasonable early in this decade (or you could argue that these same encryption requirements forced innovation and progress on large systems). Not only was there limited availability of commercial off-the-shelf software, but also it was prohibitively expensive to implement. For Requirement 3.4 (Render PAN, at minimum, unreadable anywhere it is stored), card brands (largely Visa at the time) were quick to point out that compensating controls could be implemented for this requirement, one of those being strong access controls on large systems.

In the old days, assessors would typically do a cursory walk through of the mainframe’s controls and continue to recommend an encryption solution at some point for those systems. At one point, compensating controls were deemed to have a lifespan, meaning that the lack of encryption on a mainframe would only be accepted for a certain period of time. After that, companies would need to put encryption strategies in place.

Compensating control lifespans never materialized. Compensating controls can be used for nearly every single requirement in the DSS—the most notable exception being permissible storage of sensitive authentication data after authorization (Requirement 3.2). There are many requirements that commonly show up on compensating control worksheets—Requirement 3.4 (Render the PAN unreadable anywhere it is stored) being one of them.

Even with no defined lifespan, compensating controls are not an eternal-free pass and shouldn’t be viewed or sold to executive management that way. Part of the annual assessment process is to review all compensating controls to ensure that they meet the four requirements as currently defined by the PCI Security Standards Council. Remember that the requirements can change between versions of the standard, the original business or technological constraint must still exist, and the control continues to be effective in the current security threat landscape. If certain types of attacks are on the rise and a certain compensating control is not effective in resisting those attacks, it may not be considered acceptable during your next assessment.

To further cloud the situation, it is up to the QSA performing the assessment to decide to accept the control initially, but the acquiring bank (for merchants) has the final say. Substantial documentation and an open channel of communication to your acquirer are essential to ensure money is not wasted putting together controls that ultimately do not pass muster. If you are a service provider, you won’t have that same authoritarian beyond your QSA. The reason for this is service providers typically do not have a relationship with a payment brand directly. They work for merchants directly. This is why selecting a great QSA that *understands the underlying technology he is assessing* is critically important.

Don't get discouraged, though! Compensating controls are still a viable path to compliance even considering the above caveats and descriptions of why you may not want to use them.

The authors would not be true security professionals if there were not a fun story or two based on experiences coaching companies or individuals to better security. No names will be used, and the details will be altered to protect those who were most likely being forced to try the old "Push Back on the Assessor" routine. We hope you enjoy reading them as much as we enjoyed listening to them.

FUNNY CONTROLS YOU DIDN'T DESIGN

Some of the most cherished stories and experiences come from customers and vendors that had the right intentions, but never seemed to follow the basic doctrines listed above on how good compensating controls are made. By the way, if you read this and think, "Hey!! They are talking about ME!," we're not. Pinky swear.

Before one of the authors was heavily involved in PCI, he did some IT auditing for a bank that was owned by his employer. He knew the drill of responding to audit findings. They usually start with a meeting, bringing all the key stakeholders together to mull over a comprehensive spreadsheet detailing all of the deficiencies. Findings are separated out in the "To Fix" pile, and the "To Push Back" pile, each individual item being assigned to an expert to either fix the issue or push back on the auditors. "We don't need that control because of a control over here," and "This gap does not apply to our environment," are the common phrases uttered in these meetings. Eventually, a happy (potentially unhappy) medium is established, and the audit or assessment is closed out.

The same process is often applied to PCI DSS, and the compensating control dance that commences.

Before we poke fun at the following examples, please understand that we are only illustrating a point. At no time were these suggestions made by people who didn't understand both the requirement and the capabilities of the technology in question. These people were professionals; based on their credentials and experience, they should have known better.

Encryption has always been a hotly debated topic. Our favorite failed compensating control for Requirement 3.4 comes from a vendor that called an author late one afternoon. He brought in their product team and tried to convince the author that RAID-5 was essentially an equivalent to encryption. Their argument stated you could not take any one drive and reconstruct useful data that could be considered compromise worthy; thus, their Redundant Array of Independent Disks (RAID) cards should be considered valid to sell to companies as an encryption mechanism.

To their point, if one drive (probably damaged) falls off of a truck during transport, the technology does prevent someone from reconstructing all of the data from that system. If the system was large enough, chances are that the data on the drive may not provide any tangible use to nefarious individuals either. But that's not

really the goal of the requirement, is it? Physical theft prevention is covered in other areas of the standard. The point of the requirement is to render the data unreadable anywhere that it is stored. RAID may render parts of the data unreadable (or unreconstructable) on one physical drive, but it does not render it unreadable in any other circumstance. A simple compromise of one area of the system could lead to the access and theft of massive amounts of unencrypted data.

Speaking of encryption, disk-only encryption inside data centers is not very useful either, unless additional user credentials are tied to the decryption process. Another favorite was a vendor that offered PCI compliance through an encryption appliance that was completely transparent to the operating system. So basically, the vendor was only protecting the data as it sat on a disk, in a secured facility, with gates, cameras, and Buck, the not-so-friendly security guard that embodies an ex-bouncer of a dance club—but now with a Taser®. If cardholder data sat on disk drives installed in the unlocked part of a post office, then any reasonable technologist would see the value of encrypted data while physically on disk. But since we have physically hardened data centers, the solution doesn't really do anything other than line a sales person's pocket.

Contrast this concept with whole disk encryption for a laptop. Laptops are frequently lost or stolen, and whole disk encryption (depending on configuration, of course) would absolutely serve the needs of keeping data secure when its physical container is not.

But even after all these fun stories about encryption, it's really not the big problem with Requirement 3—key management is. Once companies figure out that encryption technologies are available for their platforms, they realize that key generation and management is a whole different problem that may require additional hardware and software to fully enable as well as lots of process.

NOTE

One vendor, who apparently thought a severe case of weekend-itis had firmly set in, made a case for using the COBOL Random Number Generator (RNG) to spit out 16 digits (technically 128 bits of data) for use as an encryption key. People can come up with some really creative ideas when the fear of failing an assessment looms.

Now, the vendor in the note did actually have some good intentions. Yes, they were trying to be random and they will end up with a 128-bit key. However, anyone with a basic knowledge of encryption will quickly find the problem with that approach. The problem isn't that COBOL's RNG is less than random, but instead you have eliminated a giant section of possible key space! A 128-bit key generated in the manner described above is the equivalent of (approximately) 53 bits of encryption, thus making it computationally feasible to brute force that key. With a basic calculation of today's computing power (not including some increasingly popular GPU parallel processing), a collection of 50 computers could brute force that key

in less than 1 year. Who needs that when you can just submit ciphertext to one of those cloud-based cracking solutions to get it done quicker? Even better, lease some computer time from a large botnet, and you could be looking at weeks or even days.

HOW TO CREATE A GOOD COMPENSATING CONTROL

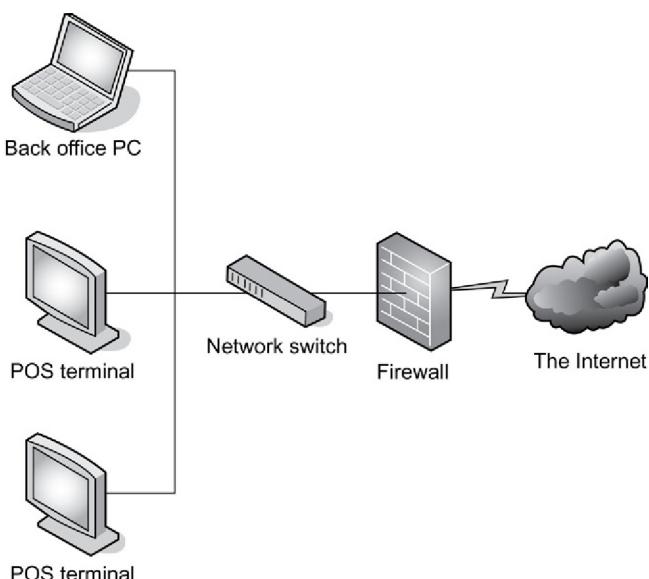
We've spent quite a bit of time setting up this section. We talked about what compensating controls are, what they are not, and some of the best-misguided attempts to create them. Before we discuss these good examples, please remember they should be used for illustrative purposes only. We have oversimplified the scenarios for brevity, and things are rarely this simple in the corporate world. Ultimately, compensating controls must be approved first by a QSA, or barring that, your acquiring bank. As one author recalls being a former assessor, he hated someone bringing an article about PCI to an interview during an assessment to bully his way through to compliance, so please don't do that with this chapter. Now let's walk through a couple of examples of how to create a good compensating control.

Let's start with a common compensating control that QSAs will define and implement at a customer. A Level 1, brick and mortar retailer with 2500 stores has some systems in their stores that do not process cardholder data. These systems are a high risk to this customer's cardholder environment because they may access both the Internet through a local firewall and the corporate intranet and Web mail system, and users log-in to that machine with the default administrator account. Store managers and retail operations claim that the systems are required for day-to-day business because each store is empowered to customize their operations to better fit the local market. The corporation believes this drives innovation and helps them maintain a competitive edge over their peers. See [Figure 16.1](#) for a simplistic view of the network.

If the retailer chooses not to segment the network, all of the systems in the store are now in-scope, and they must meet all of the applicable requirements of the PCI DSS. Doing this will add significant expense to the IT infrastructure, and will probably force a call center to be staffed up to manage the volume of calls that will come in for things like password resets.

What do you do? Do you crush the retailer aspirations to innovate by telling them they must deploy active directory to these machines, lock them down Department of Defense tight, and staff a call center? That is one option. But, if you made that recommendation, then you missed something important—understanding the business and limiting the impact that your compliance recommendations make. Instead, consider this compensating control.

Any number of network components could be used to create some segmentation in this environment. Let's say that we have a virtual local area network (VLAN) aware switch at the location that can have access lists (ACLs) tied to it. Why not create a new VLAN for just the point-of-sale (POS) network? Then create some ACLs around it to make it look like it is segmented behind a firewall. Now the threat of the

**FIGURE 16.1 Flat Store Network**

in-store PC is effectively mitigated provided that the ACLs are appropriately secure. See [Figure 16.2](#) for an example of what that might look like.

NOTE

"Wait a second," you might say. "ACLs? Those are not supposed to be used for compliance with PCI!" They most certainly can be used for compliance as we discussed in Chapter 5. Requirement 1.3.6 only refers to external connections, not internal connections. Using ACLs internally is perfectly acceptable. Where you can (technology permitting), use reflexive access lists (RACLs) that will basically look and feel like a stateful inspection firewall. You may need to review the memory and overall capacity of your switches when going through this analysis, but keep in mind that most switches developed for business in the last 3 years have this capability and more.

"But my store networks are different in every store," you say. "I can't just slap something in there like that and expect it to work globally!" If this is the case, is your store support group is overloaded with break-fix calls? Maybe this could be an opportunity to shore this up and make each store based on a consistent footprint. Keep in mind where your scope lies with any sort of jump server and the networks those servers connect to.

Barring that, how about this twist?

Let's say that you are running a Windows variant as the operating system powering your POS. You are already required to put some kind of antivirus and malware

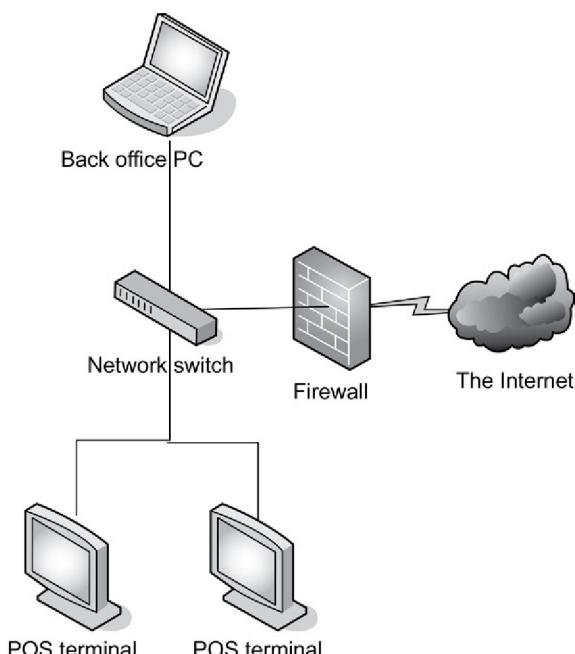


FIGURE 16.2 VLAN With ACLs Segmenting the Store Network

removal tools on there. Most of those come with software-based firewalls that could be administered remotely. Deploying firewall capabilities to the POS itself could be viewed as appropriate segmentation depending on the policy attached to that firewall. It is neither a transparent solution, nor is it very pretty, but it works.

The first solution above is really less of a compensating control and more of a way to reduce the scope of PCI. The best thing you can do for your company is reduce the scope of PCI (or any compliance initiative) to the bare minimum required, and then manage that subset of your infrastructure for security and compliance. The second example above truly is a compensating control. It meets the original intent and rigor of the original PCI requirements and provides a similar level of defense as the original requirements (reduce the vulnerability to payment systems), goes above and beyond the base requirements of PCI (firewalls are not required on devices that do not leave the premises), and it is most definitely commensurate with the additional risk imposed by not meeting the original requirement.

Take a closer look at those two suggestions. The first may be “free” to your company depending on what is already in place! You will need to adjust business process and prepare your IT community to deal with the change, but you may not need to spend any hard dollars rolling this solution out (unless your equipment cannot do this in the first place). The second suggestion, which is actually the compensating control, probably requires capital outlay for software licensing and training or

consulting to build out the environment. Upon rollout, things will break that will result in potential losses to the business.

WARNING

Without fail, companies that push major changes into large environments often face some kind of hardware or software error during the final rollout. Keep in mind, large environments always vary just a little bit among locations. Be sure to have a solid contingency and rollback plan.

Are you starting to get the hang of this thing? How about another example?

A Service Provider has a large mid-tier UNIX, like Solaris or AIX, installation that runs critical areas of the payment process, including long-term data storage. For various reasons, encrypting the data is not an option on these machines. How do we make this service provider compliant with PCI Requirement 3.4?

This is a real-world example that comes up frequently. Encryption implementations have come a long way since early in this decade. The words “my platform does not have a solution for encryption” is no longer valid for platforms that can comply with PCI. When presenting the following control to customers, it is shocking how fast they find a way to encrypt their data.

Most mid-tier UNIX operating systems have the ability to switch from discretionary access control to mandatory access control (MAC). MAC will cause that mid-tier UNIX machine to act like a mainframe using RACF or ACF2, and managing those controls is now a massive chore for the employees charged with it. Additional effort aside, converting the appropriate systems to MAC and potentially adding some segmentation could effectively render cardholder data unreadable and meet PCI Requirement 3.4.

Things are never that easy. Some security professionals inside companies love the idea of converting to MAC as it allows them to have more granular control over their systems and data. Practical ones know that converting an existing system requires so much effort that the costs typically outweigh the benefits. This is a perfect example of how a compensating control might look good on paper (it’s only three words when you use the acronym! “Convert to MAC!”), but in reality would be much easier to just meet the implied requirement to encrypt that data.

One more example, and then it’s time for you to get creative!

A medium-sized retailer with less than 500 stores is struggling with Requirement 10.2.1 to log “all individual accesses to cardholder data.” All of their data is stored in a large DB2 database that runs on a mainframe. They run massive batch processes at regular intervals, and their space constraints prevent logging every single access to a row. Do you tell them to go back to their board for a Capital Expenditure (CapEx) request to buy lots and lots of drive space to store logs?

Before we proceed, consider the intent of the requirement. Reliable logs are valuable in investigating a breach quickly. Without them, it may take forensic examiners days, or even weeks, to determine the source of a breach. Once the source has been

identified and analyzed, forensic companies must attempt to determine how many card numbers may have been exposed. If there are no logs, the assumption is that everything could be exposed, meaning that fines will add up pretty quickly.

The idea is not necessarily to make a log record that includes every single card number that is accessed, but to be able to identify which cards are accessed through the data contained in the logs. If we were to log the actual query performed against the database during a batch process, with knowledge of the date and time that the query was run and exactly what that query will do, we should then be able to determine, with reasonable certainty, which cards were accessed. Common batch processes run on a daily basis, usually using the data from the previous day to produce its output. If we must determine what could have been exposed from January 1 to January 8, we could look at the data that would have been accessed by that batch process during those days.

Logging the query, and all the other elements required by 10.3 about that action, would generate a reasonably accurate list of records that would use a fraction of the drive space required by creating an entry that has every single record exposed.

CASE STUDIES

Now that we have explored examples of what some humorous (yet invalid) compensating controls look like and what acceptable ones might be, let's walk through a couple of case studies to help us further illustrate the process.

THE CASE OF THE NEWBORN CONCIERGE

Nora's Newborn Nursery is a small, successful chain of day care centers specializing on infant and newborn care, with a small medical staff on-site to assist with minor issues that can come up while providing ongoing and routine child care. Her customers tend to be affluent and busy professionals that can sometimes have strange schedules and benefit from a concierge service designed to target professionals with young children.

Nora founded her business on the principle that her customers should never have to worry about the transaction process. Once a customer signed up for the service, they would leave a credit card on file to be prebilled for services to be rendered during the following week, month, quarter, or year. Her customers simply drop off the newborn, briefly discuss any problems or issues that are going on, and get on with their day. Nora invested in some basic IT systems and an iPhone app that allows her customers to get reports on their children while care is happening, as well as schedule additional services like routine checkups, wellness care, and seasonal immunizations. For those customers without an iPhone, her systems can alert or update her customers via text message to their cell phone.

Most of her customers pay monthly or weekly, so her transaction volume is projected to make her a Level 2 merchant in the next 12 months. As a Level 4 merchant,

she heard about PCI DSS through a presentation at the local Chamber of Commerce, but has not implemented anything at this point to comply with the standard.

Because she has a small IT staff, building sophisticated networks simply isn't an option.

She calls a consultant and sets up some time to meet. During the first conversation, the consultant describes how a centralized database and processing system could be valuable for her to invest in so that each location doesn't have to worry about on-site storage. In addition, she is looking at Healthcare Information Portability and Accountability Act (HIPAA) compliance issues with the healthcare data she inevitably stores during the course of her business.

Nora, with advice from her consultant, decides that the best course of action is to invest in a hardened, centralized computing infrastructure that houses both the applications and data that her locations will use. She will continue to store information about her active customers in an encrypted format, and ensure that hardened environment meets both HIPAA and PCI DSS compliance. For her employees, they will connect to that environment through a virtual desktop protocol like RDP, Citrix, or VMWare View. This allows her the freedom to implement any number of IT solutions in her locations, such as removing PCs in exchange for iPads or other tablet computers, or even allowing her employees to bring their own devices into the workplace. The connection between the centralized locations must be encrypted, and there must be adequate controls in the software to prevent the theft of sensitive information through screen scraping or even lost credentials.

Nora now complies with PCI DSS by building this compensating control for all the machines and network devices in her diverse environment that may not be able to directly comply with PCI DSS without a massive investment, and she knows that her IT future is both flexible and secure.

THE CASE OF THE CONCIERGE TRAVEL AGENCY

Sally's Sojourns is a medium-sized travel agency that focuses on personal getaway travel. Because they never process payments on their own, they are a classic concierge-like service provider that holds payment card information for their customers and passes it along to the various merchants that make up the vacation experience. Because Sally prides herself on arranging travel excursions to remote parts of the globe, several of her vendors are outside the United States and can only accept credit card information through e-mail, she must find a way to add some security into her business processes to meet Requirement 4.2. She knows that she cannot invest in all of these vendors to provide computing services for them, but must find a way to continue to do business with many of these merchants as they provide a personal touch that has earned Sally a fantastic reputation in the industry.

Sally decides to invest in some Data Loss Prevention technology coupled with an in-line e-mail quarantine program that will look for and encrypt any e-mail found to contain cardholder data. The remote users will get an e-mail with the sensitive data redacted, and a link that directs them to an Secure Sockets Layer (SSL)-enabled

e-mail Web application where they can view and process the end customer's payment card. In addition, any inbound e-mail found to contain cardholder data will be immediately quarantined into the same system, and the original securely erased from the original system that processed the e-mail.

Now she is able to meet Requirement 4.2, and has narrowed the scope of PCI DSS as her e-mail infrastructure may be removed from the scope of PCI DSS.

SUMMARY

What a pretty mural we have painted throughout this chapter! Good compensating controls are the result of a marriage between art and science. We've discussed what compensating controls are, what they are not, some funny examples of how to go wrong, three solid scenarios from which we created good controls, and two case studies that illustrate the discovery process.

Compensating controls are not the golden parachute of compliance initiatives. They require work to build effective ones that will pass the scrutiny of both a QSA and an acquiring bank (or card brand). Rarely do they reflect a lower total cost and compliance effort to the organization than simply meeting the original requirement. PCI DSS is based on many good (not best) standards of practice for security, and should be viewed as a baseline by which to operate not a high water mark to which you aspire. Compensating controls may help you lower the bar of compliance in the short term, but remember, only you can prevent a security breach.

You're compliant, now what?

17

INFORMATION IN THIS CHAPTER:

- Security is a process, not an event
- Plan for periodic review and training
- PCI requirements with periodic maintenance
- PCI self-assessment
- Case study

Congratulations, you made it! Your Report on Compliance (ROC) or SAQ is completed and you are ready to complete your Attestation of Compliance (AoC), your vulnerability scans came back clean, and compliance status is validated. You are DONE! Depending on where you were when you started, you may have worked long and hard to get here. So now you can kick back, relax, and enjoy your flight until you land at your next annual assessment, right? It would be great if it were that easy, but unfortunately it's not. Security (and PCI compliance in particular) requires constant vigilance, both for new controls deployment and for event monitoring. In this chapter, we will discuss how you can best spend your time now to ensure compliance in the future. First, we will discuss why you should think about security as a process instead of an event. We will then make suggestions on periodic review and training that should be happening in your organization. Lastly, we will outline some suggestions on performing a self-assessment on your own network.

SECURITY IS A PROCESS, NOT AN EVENT

Compliance is a process, including Payment Card Industry Data Security Standards (PCI DSS) compliance, which is intended to induce people to get to security.

Security is not something that can be achieved and then forgotten about. Contrary to some security vendors' claims and management hopes, you cannot install a magical device on your network that will make you eternally secure. Security is a process of constantly assessing your risks then working to mitigate them to a reasonable level. These risks are ever-changing, so processes and technology to address them should be ever-changing as well. To put this concept another way, the concept of security is akin to a journey, not a particular destination.

One thing to keep in mind is that you are never 100% secure. Even if you've done everything to secure your systems, an attacker can still find ways in (sometimes through low-tech ways like social engineering). In fact, it's actually very difficult to prove that you are secure, while it's relatively easy to prove that you are insecure. To prove that you are secure, you must prove that every possible threat (remember, these are constantly changing) is addressed. To prove insecurity, you only have to find one attack vector that allows you past the rest of the security controls in place. Thanks to more zero day attacks, that vector could be something you had not even considered. It could be an exploit known to only one attacker in the world; if that attacker decides to target your company, then despite all you have done it could successfully compromise your network.

First, the more complex a system, the harder it is to secure. It is very difficult to have a system that's completely risk-free while actually doing something useful, like serve a Web page or allow a user to send e-mail. In general, today's systems are very complex, and therefore hard to secure.

Second, risks, technologies, and your organization are changing constantly. New attacks are invented constantly. Geopolitical movement opens and closes threats against your organization. Hactivism is a real threat to your ongoing security posture. New technologies and software are implemented in your network on a regular basis. New people come to your company and current employees forget things from time to time. People need to be trained regularly.

Third, it's impossible to be PCI compliant without approaching compliance and security as a process. All of the requirements require some sort of maintenance. Logs need to be reviewed, systems and policies need to be updated, and security assessments need to be performed. These are all part of the security process that keeps your company as safe as possible from attack.

NOTE

All of the requirements mandate some sort of maintenance. Please remember that while your compliance is validated by a QSA or ISA, it is maintained by you and you alone. Validation is a moment when you feel like you accomplished something and you did. Security is an ongoing state of vigilance, with compliance as a by-product.

PLAN FOR PERIODIC REVIEW AND TRAINING

It's important to plan now for future review and training. Working with technology in an organization can get very hectic, and if you put off planning then you are far less likely to do it. It's important to review your security polices and practices often to verify they're actually in place. In fact, many of the PCI DSS requirements mention monthly, quarterly, and annual processes.

NOTE

Here are some examples from PCI DSS requirements and testing procedures that mention ongoing process:

Requirement 1.1.7: "Requirement to review firewall and router rule sets at least every six months."

Requirement 3.1.b testing procedure: "(Verify) either a quarterly automatic or manual process is in place to identify and securely delete stored cardholder data. The quarterly automatic or manual process is performed for all locations of cardholder data."

Requirement 11.1: "Implement processes to test for the presence of wireless access points (802.11), and detect and identify all authorized and unauthorized wireless access points on a quarterly basis."

Requirement 11.2: "Run internal and external network vulnerability scans at least quarterly and after any significant change in the network."

More examples are covered in the next section of this chapter called "PCI Requirements with Periodic Maintenance."

Many times companies write great policies but they never enforce them, and so they are never actually followed. Train and test your employees often so they are aware of your security policies and re-emphasize their importance so that employees are more likely to follow them.

NOTE

Perform training sessions that are brief and frequent. For example, a short 15-min reminder session several times a year will probably be better than an hour-long review session once per year. Here are some ideas of things you may want to review with employees at your organization:

Passwords: What makes a good password? Remind people never to share their password with anyone for any reason. Warn employees of common mistakes such as writing passwords on a post-it note and sticking on the computer monitor.

Social Engineering: Don't let people fool you. Make policies for visitors clear to ensure that a malicious visitor won't leave with information they shouldn't have. Also, you should review policies for verifying an employee's identity when they make requests (such as password resets) over the phone or in some other non-face-to-face situation.

Physical Access: Verify that everyone knows what a visitors badge looks like and knows what the company policies are with regards to where visitors are allowed to go and where they are not allowed to go.

Correctly Storing and Destroying Sensitive Material: Help employees keep up-to-date with company policies that require sensitive data be destroyed. For example, it's important that employees are trained on destroying paper and electronic media that contains confidential data when it's not longer needed.

Your Information Technology (IT) staff also needs to be regularly trained on security. For example:

- *Secure Coding Practices:* Software engineers don't necessarily need to be security experts; however, it's important that they understand secure coding

practices. Anyone working on a Web application should be aware of cross-site scripting (XSS) and Structured Query Language (SQL) injection bugs, which are shockingly prevalent given their age and ease of remedy. Programmers should also be aware of unsafe functions that may be available in their language, and their safer alternative functions (e.g., printf() vs sprint()). Requirement 6.5 mandates this.

- *Systems Training:* Systems administrators should be kept up-to-date with secure practices that are related to the systems they administer. They should know how to securely install and configure these systems. Many of the Requirement 2 items mandate that.

Finally, security professionals at your company must be trained regularly. Depending on the size of your company, this may be a few or several employees. These people are responsible for securing your systems every day. They must receive periodic training to help them be aware of new technologies and new attacks.

Regularly review the PCI requirements and compare what is asked with what you have at that moment. Focus on the requirements that cause your company trouble. Set aside a day per month for your review. A good rule of thumb is to review all PCI requirements at least quarterly. Reviewing on this schedule should keep you in great shape for your quarterly scan and annual assessment as well as keep you up-to-date with any changes in the PCI requirements. Your self-assessment process can be as detailed as walking through the testing procedures of the PCI DSS, or reviewing Self-Assessment Questionnaire D.

PCI REQUIREMENTS WITH PERIODIC MAINTENANCE

PCI DSS has many requirements that mandate ongoing actions with varying outcomes. Some requirements have documentation outputs that are reviewed during your annual assessment, and other requirements actions are in fact the compliance activity. Finally, some requirements don't have an actual maintenance requirement, but there is documentation that must be updated before an assessment, so we'll cover those as well.

WARNING

Please, if you remember one thing from reading this entire book, let it be: you are never “done” with security. Also, while validating your PCI compliance is your QSA’s task, maintaining compliance and data security is yours and yours alone.

Before we continue on with this section, the framers of PCI DSS v3.0 made a pretty dramatic push away from using specific language around how often to review certain controls. The instances of the term periodic increased from 8 to 20. What we

will cover here in this section are requirements that have a defined review term. Open up PCI DSS 3.0 in your PDF reader and do a search for the term “periodic” to find all the other more ambiguous requirements.

BUILD AND MAINTAIN A SECURE NETWORK

No major updates here in PCI DSS 3.0. Requirement 1.1.7 requires a review of all firewall and router rules and configurations every 6 months. That means that when your annual assessment is due, you should have documentation from at least two of these reviews; that is what your QSA will be asking for. The reviews should be detailed enough to show that an engineer checked every item and validated that it was still needed.

Your assessor will also review documentation for two other requirements in this domain, and while they may not have to be updated through a formal process, many companies have failed parts of their PCI Assessment because they forgot to do something simple like update a configuration standard that referenced outdated, or known vulnerable software. Requirements 1.1 and 2.2 fall into this category. Be sure that during your normal review for Requirement 1.1.7, you review and update your firewall and router configuration standards for Requirement 1.1. Do the rule set review first and note anything that is not in the standard, then go update the standard so that it matches what is in the rule set. While it may seem overly picky, an assessor would be right if he noted outdated configuration standards for Requirement 1.1.

In addition, go back through all of your in-scope system types and ensure that their configuration is up-to-date for Requirement 2.2. Most companies fall short on this requirement when they reference out-of-date software packages or versions that have security vulnerabilities in them. For example, if your configuration standards still say to start with a stock build of Sendmail 8.9, an outdated and vulnerable version of Sendmail, you would not be able to pass that requirement.

PROTECT CARDHOLDER DATA

Requirement 3 has two key items to address, and one that may just need to have a fresh set of eyes. First off, Requirement 3.1 mandates that you create retention requirements for cardholder data. Requirement 3.1 has a quarterly requirement to purge old data by way of manual review or automated disposal process. Even if your process is 100% automated, take the time each quarter to make sure that the processes are in fact functioning correctly. During your assessment, expect your assessor to ask you to produce your oldest set of retained data. They will then check to make sure that it does not exceed the retention requirements set forth in the aforementioned review.

WARNING

Our guidance from Chapter 7 still stands: it is usually easier to not store the data than to protect stored data. This is not a traditional way to think about data security, but it is one way of thinking about which is extremely useful for painless PCI DSS compliance.

Consider expanding that quarterly process to include a review of the actual requirements to retain data. The business, regulatory, and legal environments can easily change more than annually (maybe not all at the same time, but one of the three will happen at least more than annually), so use that time to be sure that you do not need to alter your data retention requirements. If you do, republish the information and perform a new review to make sure you are in compliance with your own policy.

NOTE

Time and time again, companies write policies in good faith and then completely ignore them in practice. As an example, Matt works for a large regional grocery chain that is a Level 2 merchant. His company's corporate policy is actually more restrictive than PCI, and requires that all security patches be installed within 10 days of release, and cardholder data must be encrypted internally over the wire. When Matt's assessor was performing the assessment, she noticed that one process for the florist shop inside his stores did not encrypt the cardholder data over the wire. When Matt confronted the department responsible for the florist shop's systems, the response he got back was "Well, it's not a PCI requirement so we just did the minimum," thus completely ignoring the corporate policy. Usually you don't find that just one policy is violated—there tends to be many. This would be a clue for his assessor to dig deeper to ensure that policies requiring the minimum PCI requirement are actually being followed. Moral of the story? If you have a corporate policy that exceeds the base requirements of PCI, FOLLOW IT. There is a reason why that policy is in place!

PCI DSS 2.0 gave us some reprieve on key rotation and PCI DSS 3.0 honors that change. Requirement 3.6.4 states that keys should be rotated once they have "reached the end of their cryptoperiod (e.g., after a defined period of time has passed and/or after a certain amount of ciphertext has been produced by a given key), as defined by the associated application vendor or key owner, and based on industry best practices and guidelines." This formerly was an annual requirement to rotate all keys used for the encryption of in-scope data. Expect some challenges with your assessor here. They may still want to see annual changes, and you may need to educate him on what an acceptable cryptoperiod is for your particular implementation. Your assessor will first look at your key management processes and make sure the requirement to rotate is documented, and then will check your implementation to see if you actually did rotate the key per the document. This becomes challenging on new installations with longer cryptoperiods (say a new installation with a 3-year rotation requirement), so again, expect an education session. While you are reviewing your documentation for this, take a look at the overall key management processes and procedures and ensure they are up-to-date for Requirement 3.6. Common gotcha's here include forgetting to update the key management processes when you change encryption technologies. That's a big, red flag for an assessor, and it is pretty easy to spot.

MAINTAIN A VULNERABILITY MANAGEMENT PROGRAM

Vulnerability management is a task that is easily seen as ongoing. New vulnerabilities, viruses, and malware are found every day, and companies need to take the appropriate precautions to ensure that they are protected. Requirement 5.2 uses that

indefinite term periodic. To comply with this requirement, you must perform periodic scans of your in-scope machines.

TOOLS

We cover the vulnerability scanning tools that help with external and internal network scanning in Chapter 9.

With the ever-changing landscape of attacks and malware, you should probably scan your systems at least once a month, and likely more frequently. More of the quality vulnerability vendors offer unlimited scanning, or you can use freeware tools such as Nessus as well for internal scans and “unofficial” external scans (those not performed by an ASV). To determine how often you should scan your systems, perform a risk analysis of these systems and include elements such as the criticality of the systems, amount of processing power, business hours and uptime requirements, bandwidth, and the frequency by which you update your antivirus definitions. Such risk analysis is in fact prescribed in Requirement 12.2.a and must be performed annually (security policy must include “an annual process that identifies threats, and vulnerabilities, and results in a formal risk assessment”). PCI DSS does not give you a time period for performing a full scan, but most assessors should minimally accept quarterly as an acceptable period if you were including on-demand scanning as well. It’s a slippery slope with many variables, but more frequent scanning followed by the ongoing remediation is clearly better. If you do your homework, you can take your assessor on a journey through your logic, with documentation to back it up.

Security patches must be installed within 1 month of their release. This particular requirement will quickly ruffle the feathers of system administrators, and particularly those of mainframes and large databases. These systems have high availability requirements and typically are not brought down frequently to install patches. When looking at Requirement 6.1, be sure that the patches you are considering installing are actually security related. Only those patches must be installed according to this schedule. Alternatively, if you have a way to mitigate the vulnerability without installing the patch, do that. Given the nature of some virtual systems, it may be possible to patch without taking the actual service down, so don’t get lazy. Be vigilant with your security and patch! Since PCI DSS version 1.2, the note below Requirement 6.1 allows you to take a risk-based approach on your patches, and prioritize critical systems first. Then you can patch the remaining systems within 3 months (quarterly) or longer instead of one.

NOTE

A good example of an (old) action that could be used to mitigate a security vulnerability instead of deploying a patch is the Microsoft Graphics Rendering Engine vulnerability from early 2006 (www.microsoft.com/technet/security/Bulletin/MS06-001.mspx). In their notes, unregistering the DLL mitigates the vulnerability until the patch is deployed. This type of mitigation is acceptable for PCI DSS compliance.

Finally, be sure that you scan your public-facing Web applications at least annually, or after any changes for Requirement 6.6. Your assessor will review your policy and procedure documents, as well as review the output from processes like the above to make sure it is occurring as prescribed. If you do not want to go through this process, Requirement 6.6 allows you to use a Web Application Firewall (WAF) in lieu of the above process. Most companies end up choosing a hybrid of the two. They may have a WAF protecting their applications, and also perform an automated Web application security scanning as needed. Keep in mind that both scanning and WAF monitoring are ongoing tasks; scanning is periodic while WAF management is continuous. Also, keep in mind that scanning is not terribly useful unless vulnerabilities found by the scanner are actually fixed!

IMPLEMENT STRONG ACCESS CONTROL MEASURES

Strong access controls are key to preventing unauthorized access to your data. While Requirement 7 does not specifically have a mandate to review access granted to cardholder data, consider adding it to your quarterly Sarbanes–Oxley review (or if you do not do these, review cardholder access quarterly). This will help to keep only those with a business or job-related requirement to access in-scope data part of the access groups that grant this type of privilege.

For those that do have a requirement to access in-scope data, be sure that users inactive for more than 90 days are disabled or removed (Requirement 8.1.4). To save time, build an automated process to do this, and assess the process quarterly. Be sure to document everything you find! While you are doing this, think back to the last time you changed your password. Did all systems you access force you to do it at least quarterly? If not, follow up on that now. Don't wait for your annual assessment to learn that your systems are not requiring quarterly password changes (Requirement 8.2.4).

Do you use off-site storage for tape backups or hard copy media? If so, be sure you visit it at least annually and review its security. Some of these facilities have annual assessments performed and can provide the documentation to you upon request. If they do this, be sure you have a chance to review the methodology used by the assessment company, and it includes common security controls in the scope of the review. Statement on Standards for Attestation Engagements (SSAE) 16 assessment reports are often provided for this type of due diligence. You may want to do this in conjunction with your annual media inventory process for Requirement 9.5.1.b.

Finally, another one of those requirements without a stated review period is Requirement 9.2. You developed procedures to visually distinguish identification provided to both visitors and employees. Make sure it is up-to-date, addresses all personnel, and is still accurate as part of the hire and fire process.

REGULARLY MONITOR AND TEST NETWORKS

It's time for one of our more frequent periodic review elements—daily log review for Requirement 10.6. Chances are you don't have a team of people driving themselves insane by reading every log generated by every in-scope device. You most likely have

some tools to collect, aggregate, normalize, and correlate these logs. As part of that process, the log management tool should be intelligent enough to find items that are not part of normal operations and create alerts to your staff for follow-up. Have your internal assessment group periodically assess this process to make sure it is working as designed. It is more important to review an ongoing process for log review and to make sure it is current and active than to review every single log manually.

Next comes Requirement 11. Almost all subrequirements under Requirement 11 have some kind of periodic action associated with them. Starting with Requirement 11.1, quarterly running a wireless analyzer at all of your locations to search for unauthorized wireless activity. As we discussed in Chapter 8 doing a quarterly walk-through may not be the best idea from a resource and threat mitigation perspective. If you only have one location the logistics might work, but from a security protection perspective you are not doing everything you should. If you have chosen to deploy Wireless Intrusion Detection System (IDS)/Intrusion Prevention System (IPS) technology instead, perform a quarterly assessment of the systems to ensure they are properly alerting and deploying defense and containment measures that you configured.

Next comes the quarterly scans. Requirement 11.2 requires that both internal and external scans are performed at least quarterly, with clean scans being the desired outcome for both of those operations. Save each of those clean quarterly scans in a special place because your assessor will need them as part of the on-site assessment review. Here is another requirement that can benefit from network segmentation as a scope reduction exercise. Companies are required to scan all in-scope internal systems, and all external IP Addresses. PCI DSS scoping instructions state that systems connected to networks that process cardholder data are in-scope for PCI DSS, even if they do not process or store cardholder data themselves. Thus, putting firewalls between these systems will reduce the amount of work you need to do for your internal scans. Externally, you can reduce the scope of your scans by air-gapping networks (meaning that the networks that process cardholder data externally are physically disconnected from ones that do not) or other strong segmentation techniques. Every situation is slightly different, so it is best to check with your QSA or ISA on which method works best for you.

Although those quarterly scans should address most of your vulnerabilities, you must also perform a deeper inspection of your systems and applications at least annually with a Penetration Test (Requirement 11.3). Two common mistakes companies make are forgetting to include applications (Requirement 11.3.2) and neglecting to perform the penetration test from an internal perspective as well as an external perspective. New in PCI DSS 3.0 is the requirement to test that segmentation actually works as part of your penetration test. If you claim large areas of your network are out of scope due to a firewall or other type of segmentation, be sure you have a penetration tester prove that segmentation is effective. The penetration test is one of the most critical periodic requirements used by PCI DSS. The quality of your assessment will directly impact the likelihood of being caught up in a breach due to an external (or internal) attack. Your penetration test should include the following elements:

- Network attacks,
- Application attacks,

- Social engineering,
- Modem penetration testing (if applicable),
- Wireless penetration testing (if applicable).

Once your test is performed, be sure to address any findings and have them re-checked to make sure your fixes worked.

Finally, Requirement 11.5 has a periodic requirements to consider. The first is to perform critical file comparisons at least weekly, meaning that all in-scope machines (including the point of sale) should have some kind of file integrity monitoring run on its files. Whether you go with a commercial solution, or script something using a hashing utility like *sha5sum*, be sure that you run those comparisons at least weekly and for the second part of the review, follow-up with any exceptions that come out of the process. The only reason an exception would be generated by the file integrity process is if you deployed a system update or patch and can track the exception back to a change control ticket, or if something bad was happening (maliciously or not) to the system. Both should be followed up on and closed appropriately.

MAINTAIN AN INFORMATION SECURITY POLICY

If you have not had enough with documentation, here's some more periodic requirements specifically for your information security policy and framework. Requirement 12.2 mandates an annual risk assessment, and potentially as a part of that you can conduct your policy review. Your organization probably already performs some kind of annual risk assessment as a part of normal operations. You may be able to tag along with that, but be sure to include an assessment of risks relevant to PCI DSS. If you ignore all in-scope systems and don't include the risks associated with storing cardholder data, the risk assessment is not much help to your PCI efforts. Including your policies in the risk assessment might be a nice way to kill two birds with one stone. Be sure to update your policies to address findings in the risk assessment! Your assessor will look for this!

One of the more abstract requirements attached to the end of every major requirement is the operational security review. This should include (but not be limited to) things like following up on log alerts, searching for new threats to systems, and checking on vulnerability data—basically anything you are doing in any of the other requirements. Your assessor will want to see your definition of these items, and will want to see evidence that you are following the process. A simple (but probably not scalable) way to do this is to have a “SOC Analyst Diary,” a text file on a shared drive or other common area where personnel write to when they check things—even when nothing out of the ordinary is found.

Security awareness training is something that is often overlooked, but critical to maintaining a secure company (Requirement 12.6). You must demonstrate this ongoing educational process occurs at least annually and for all new hires. Limiting this type of training to only once per year will meet PCI DSS, but it is a bit too relaxed to be considered a security best practice. Find fun ways to engage your employees on

information security. Do demonstrations or exercises often. Consider posting things in break rooms or common areas where employees congregate. Add something to their pay stub. Or even involve them in a demonstration like “Find the Bad Guy” where you have an employee walking around with a fake badge, or maybe no badge at all. Offer prizes like Starbucks or iTunes gift cards. You will be amazed at how well this will improve the weakest link in your security chain: people.

None of us want to invoke an incident response policy, but like your last will and testament, it needs to be defined and planned in case something bad happens. Requirement 12.10.2 mandates annual testing of the incident response policy. Instead of hiring someone to steal something from your company, consider doing a tabletop exercise whereby you simulate a relevant and creative incident, and walk through all the steps in your plan. This will help your employees understand their roles and will ferret out problems or gaps in the process. In addition to doing this, you must train your staff with security breach responsibilities so they have the specific skills needed to respond. This may include sending key employees to forensics training, or Continuing Professional Education classes around breaches and their aftermath.

PCI SELF-ASSESSMENT

In addition to the elements called out earlier, take the time to review all of the requirements at least once before your assessor shows up. You should be going into an assessment by a QSA or ISA knowing that you will pass. Don’t rely on the assessor to find all of your gaps. Instead, show your assessor that you have been compliant all year long. You’ll be amazed at how much faster your assessment goes, and how much more confident your management will be when asked about your current PCI DSS posture.

The PCI Security Standards Council (www.pcisecuritystandards.org) provides some great documents to help you with your self-assessment. For example, Self-Assessment Questionnaire D (for either merchants or service providers) can help you to determine your company’s current compliance level in a Yes or No format. You should periodically review these documents and use them in your own assessment processes, and look for ways to improve your company’s security posture.

CASE STUDY

You have read plenty of case studies in this book where someone made a mistake that lead to a breach. It’s not that there are no GOOD programs in this world; it’s just that we (should) learn from others mistakes. Those lessons will hopefully prevent us from walking off the proverbial cliff. However, in this chapter, we’re turning the tables and presenting a case study where a company got it right. Not only right, but knocked it out of the park!

THE CASE OF THE COMPLIANT COMPANY

Peggy's Pad Thai Palace, a national Thai food chain, enjoyed quick success with its multiple brands and quality food. Peggy's vision was to create two sides to the business and started with a high-quality dining experience in freestanding restaurants. Once she had her recipes down, she figured out how to prepare something very close to the same quality and taste in a mass-produced way, and opened hundreds of stores in shopping malls across the country. Her success spread like a wildfire, and before she knew it, she was managing a billion-dollar enterprise with more than 3000 locations.

She began accepting credit cards in her mall locations 5 years ago, and quickly ascended through the levels to become a Level 1 merchant. Before she became a Level 1 merchant, she attended many conferences and seminars geared toward restaurateurs, and heard alarming stories about how people lost their restaurants because a hacker breached their point of sale systems and stole credit card data. Peggy was determined not to let that happen to her.

Peggy began investing in security soon after hearing these stories and had a full enterprise security assessment performed to see how she stood up against the ISO Standards on information security (17799 at the time, 27002 now). She further invested in a framework to address the gaps discovered during the assessment, and within 2 years had a mature ISO program, complete with compliance management. She hired a small staff of individuals to handle her information security and compliance needs, and after completing several internal assessments against PCI DSS, she set out to find a QSA.

Her QSA came on board during the next quarter, and was duly impressed at how organized her processes and systems were. Through segmentation and other scope reduction techniques, Peggy had reduced the scope to the POS terminals and one additional machine per store, three machines at the corporate office, and a few routers and switches in between. Overall, it was less than 1% of her total infrastructure. She also was sure to complete internal preassessments before the QSA arrived, and had all the documentation built and ready to go.

Peggy approached compliance as a cost of doing business, and was sure that she invested in security to cover her compliance needs. Her company went into each assessment KNOWING the outcome would be positive, and never had an issue passing her compliance assessment, and building security best practices to cover security threats where PCI DSS did not.

SUMMARY

Security is fleeting. You've got it 1 min and it's gone the next, but there are some steps that can be taken to keep yourself as secure as possible. Working with management and employees, you can keep your company in a good position to combat many attacks now and in the future. Things you can do to maintain your security include

keeping your policy up-to-date, periodically assessing your security, and periodic training.

Please remember that it really doesn't matter how many times the author team will say "security first!" If some organizations don't get security with all of its complexities and ignore it for years, "compliance first" becomes a real choice for them. At least they can understand it. And then later, "compliance first" becomes "compliance ONLY," "checklists" replace "risk awareness," "flowcharts" replace thinking about their threats and vulnerabilities. And then hackers get them!

Reading this chapter should have reminded you that maintaining ongoing PCI DSS compliance and data security is your job, not ASVs, not QSAs, not your bank's. It is yours and yours alone! You can't achieve security; it's a never-ending process; you must constantly be assessing and working to mitigate risks. Also, you should plan now to train and review the employees and IT staff regularly to keep them reminded of and up-to-date on company policies.

PCI DSS changes as well: review the PCI requirements regularly to keep yourself up-to-date.

But more often: systems and attacks change. Regularly assess your systems to ensure they are still PCI compliant and secure. Similarly, regularly review your policies to verify they are up-to-date and are working at your company.

Emerging technology and alternative payment schemes

18

INFORMATION IN THIS CHAPTER:

- New payment schemes
- Predictions
- Taxonomy and tidbits
- Case study

After getting this far in the book, you may be thinking that PCI DSS compliance is both insurmountable and unsustainable. We disagree on both counts, but we're writing this from our side of the page, not yours. Both authors have seen many companies where PCI DSS just wasn't a good fit. Be it culturally, technically, or simply the sheer willpower to get it done, PCI DSS may be one of those things that you choose to avoid.

If this is you, don't despair! There are some interesting emerging technologies and alternate payment methods that could be useful to you. Keep in mind, as a society we have been trained to reach for the plastic. No doubt, your customers are similar. Before you can rock the boat, you need to consider things like average ticket size, how often you directly interact with the customer, and your customer experience life-cycle (how long from entry to payment?). Each of those things will affect your customer's behavior while patronizing your business. For example, everyone can move to a cash-only business, but if you are selling furniture and televisions where the average ticket price is over \$100, putting an Automated Teller Machine (ATM) out front may not get you the customers you need at the rate you need them to survive.

NEW PAYMENT SCHEMES

Getting customers to part with precious Dollars, Pounds, Yen, and Euros isn't new, but some of the ways we do it are. For the last 10 years, we've seen a tremendous advancement in technology and interoperability between handheld devices and fixed ones. From scams to legitimate business needs, there are new problems, solutions, and opportunities to attack this issue.

EMV

OK, before the non-US folks yell and scream how behind the times we are here in the States, it's important to discuss what EMV means for us in general. When this

becomes the norm in the US, the use of magstripe will begin to decline dramatically. There are some places where it will continue to be used in the US, specifically with Automated Fuel Dispensers or other unattended payment terminals, but their usage will decline. With this decline we will see a shift in fraud to the Card Not Present channel (online/mobile). What this means for your Web site is that you will have a new breed of attacker that is voraciously going to throw everything they have at your technology to steal your data. Be prepared!

You can also be prepared for there to be confusion for card present transactions. EMV is commonly known as Chip & Personal Identification Number (PIN) outside the US, but here the closest tagline we can use is Chip & Choice—not very elegant. The reason for this is that any intermediary between the cardholder and the issuer can choose to lower the security of the transaction down to what will likely be Chip & Sign. So for you EMV cardholders in the US who travel internationally, you will present your chip like every other citizen in that country, but instead of using a PIN a receipt will print out for you to sign. The holder of the terminal will look confused for a bit until it becomes expected. EMV transactions that ride the Debit rails may end up being closer to that Chip & PIN variant we are familiar with, but we will have to see how things end up in implementations.

Let's be clear—EMV is not hack proof. In fact, there is some great research out there that shows hacks for both online and offline transactions. That said, it is a huge step forward in transaction security. For more information, check out Dr. Ross Anderson's page (<http://www.cl.cam.ac.uk/~rja14>) for a good number of papers on this topic.

At a minimum, now is the time to ensure that your terminals have an EMV slot, and that it is soon to be fully active. By October 2015, you will need these terminals to be eligible for a liability shift that is happening for chargebacks. Check with your acquirer for more information on this shift and how it will impact you.

MOBILE

Let's continue with mobile. While we discussed the technology and its applications in Chapter 12, there is still much to discuss from an emerging payments perspective. Smartphones are ubiquitous now. Businesses realize that these devices are everywhere and are always on the lookout for ways to monetize it.

One of the simplest mobile payment options doesn't even require a smartphone. SIM-based payments are commonly used in the US to attract charitable donations after natural disasters, but they can also be used to pay for taxis, parking, and even items in vending machines. They typically require the user to send a text to a 5- or 6-digit number with some code associated with it, and the charge will show up on their mobile phone bill at the end of the month. As a business, you may choose to accept SIM-based payments in this realm, or soon as part of a Near-Field Communication (NFC) scheme (more on this in a bit).

SIM-based payments use the same SIM chip that authenticates your phone to your carrier to figure out who purchased something, and bill them accordingly. It's a neat solution for on-the-go payments; or for something that is self-service in a manner where the "store" is automated or electronic. You must have the ability to

receive such confirmation of payments in your store, which will require you to both have a contract to accept them and a reliable Internet connection. Just like an online payment card transaction, if your Internet connection goes down you won't be able to receive the payment information unless you invest in sophisticated over-the-air technology (which still must have two-way connectivity to function).

From a PCI perspective, if your business ONLY accepted SIM-based payments, you wouldn't need to worry about PCI DSS unless you were somehow getting payment data back from the provider (unlikely this is the case). It's an interesting alternative, and depending on the ticket size and how someone's phone plan is set up, it might work. For individuals with company phones, however, this type of payment scheme may not work, or may cause problems for the individual at the end of the month.

NOTE

You must understand that regardless of your situation, any time money is exchanging hands there is a possibility for fraud to occur. There is no silver-bullet solution, and part of doing business is taking on risk. There are several variables to consider when deciding how to accept payments for your business. Those are as follows:

- Ticket size: How big is your average transaction size? Certain thresholds may be conducive to some methods over others, and you may notice more fraud from certain types of schemes depending on the ticket size. Payment cards are good for higher ticket sizes, cash for low, and others fall somewhere in between.
- Transaction volume: How many transactions are you running per day?
- Customer volume: How many customers do you see per day (each with a potential to pay in their own special way)?
- Average sale time: How long does it take for someone to run through the line? For high-volume, rapid transactions you may not be able to pick certain schemes over others.

With any change in payment acceptance you can expect you will see a change in fraudulent activity. Just like someone can spoof a magstripe card by capturing and reprinting the data, some SIM cards can be spoofed in the same way. You may end up accepting a fraudulent payment and allowing someone to exit the store to end up having the equivalent of a chargeback hit you without much recourse. Ten years ago this type of spoofing was very difficult to do and typically required equipment purchases into the six-figure range. Now with some parts you can get from eBay and knowledge, you can do this for under \$2000. If your storefront is unattended, this may not be an issue if the ticket size is small. It's no different than any other kind of fraud. But if you are using SIM-based payments in places where your average ticket size exceeds \$10, you may find yourself learning quite a bit about fraud (the painful way).

NEAR-FIELD COMMUNICATION

NFC technology has been used in the payment space for years as an alternate, dynamic payment mechanism. Payment brands released products like Blink (Visa) and

PayPass (MasterCard) where Radio-Frequency IDentification chips were embedded in traditional plastic, magstripe cards as an alternative to swiping. You simply wave your payment card over the reader and walk out with your goods. No signing, no PINs, nothing static to capture and replay. In fact, this is one of the two technologies you can use to qualify for Visa's Technology Improvement Program (TIP), which can effectively eliminate the requirement to validate compliance annually (see <http://brandenwilliams.com> for thoughts on killing PCI assessments). Some NFC payments might be SIM-based as well, thus going through a carrier instead of a payment scheme. Some geographies have different adoption rates than others, so you should understand your options when building your payment strategy for each area you are doing business.

NFC payments have a huge advantage to merchants as they are an element that allows you to qualify for Visa's TIP program along with accepting EMV transactions. For those of you reading this outside the US, you can also qualify for the TIP program if you already accept EMV or Chip & PIN. It might even work to your advantage, especially if you can deploy some level of point to point encryption in your network to encrypt the information that EMV considers "routing," but can in fact be used to push fraudulent transactions through in other parts of the world. Keep in mind that the Visa TIP only works for Visa, and you may have a couple more hoops to jump through for other payment schemes for the same benefit.

Many smartphones are now coming with NFC capabilities built in to replay your existing cards over the phone transmitter. For example, Google Wallet (at the time of this writing) allows you to register your Citibank MasterCard and use your phone to pay for things anywhere that PayPass is available. This doesn't change anything for you as a merchant as it will use your existing infrastructure to complete the payment.

NFC carries its own risks as well. While the information is dynamic, computing power increases at incredible rates. Cryptanalysis is not only possible, but becomes feasible in some cases when bad guys put their efforts to try to reverse these algorithms. What does that mean for you? It means that while the risk of a spoofed card is reduced to almost zero with EMV, it is still possible and can still happen. The rate is so low that you can probably write it off as a cost of doing business, but you should understand your liability if you end up in that situation.

SQUARE (AND OTHERS)

We would be remiss if we didn't mention this creative little scheme (<https://squareup.com/>)! This solution allows you to both accept payments with phones and tablets, and create the equivalent of a digital wallet whereby users can put payment information into an app on their phone and use that app to pay for goods and services. It's not quite the same as say the Starbucks app where you can register your Starbucks card and pay via a barcode on your phone as you still are passing your actual payment card information to the merchant for processing. Merchants like the technology as it can enhance their ability to provide top-notch service to their customers. Consumers

like it because they can almost get to the point where they just need a smartphone to live their lives.

One element of the Square scheme is the tiny reader that can capture a magstripe and send it along for processing. This makes payment acceptance for small businesses a breeze when they are doing events outside of their standard storefront. Think about all the outdoor festivals you have been to and all those extra wireless terminals you see. As a merchant, I'm definitely looking to that device as a way to save money and time. No longer do I have to take the card to the back of the store for processing, I can swipe it right in front of the customer and get them out the door.

Keep in mind, using Square won't excuse you from your PCI DSS responsibilities. Depending on how you have it set up, you may have greatly reduced responsibilities and risk, but it's not as complete of a solution as some of the other options out there. Your biggest risk with Square may simply be the concept of acceptance. If people are unwilling to use the app on their smartphone, or allow you to swipe their card through the fancy reader, you won't be able to close the sale.

Square may have been one of the first on the scene, but they are not the only game in town. Paypal, Intuit, and many others have released competing solutions, not to mention the myriad of apps that can accomplish this as well.

GOOGLE CHECKOUT, PAYPAL, AND STRIPE

Let's spend a few minutes talking about card-not-present transactions. Online retailers have to deal with PCI DSS just like brick and mortar ones do, but the security at online retailers tends to be much better because they live with electronic fraud every day. Brick and mortar stores tend to focus on theft and fraudulent payment devices more than they do an external attacker constantly pounding on their door to steal things. Regardless, there is a fantastic solution for eTailers to adopt that will allow them to be exempt from PCI DSS.

Both Google and Paypal offer products to merchants that take the burden of payment processing away. These may be offered in multiple delivery methods, but the way to ensure you are exempt from PCI DSS is to ensure you are redirecting your users to Google and Paypal during the payment portion of the checkout process. If you are accepting the card information and passing it to Google and Paypal on the user's behalf, you are still in the middle of the transaction and must comply with PCI DSS. Both allow you to set up a recurring payment for subscriptions as well, so businesses with those models can take advantage as well.

Essentially in these schemes you would get your user all the way to entering payment information and then pass them to Google or Paypal with key information related to the transaction. The user would then choose how to pay within those systems (there are multiple methods) and then be routed back to you after the payment has completed. You will get cash in batches but without the payment card information attached.

Some disadvantages include losing track of the user while they check out. Some eTailers focus on their user experience so much that passing their customers to a third

party during the most critical part of the transaction may be unacceptable. Google and Paypal aren't new technologies anymore, so I would largely argue that using these services won't dramatically increase your cart abandonment rate. Another disadvantage may be the cost to process. Doing the processing directly may cost less per transaction, but in theory, those savings should be used to secure your network and deal with PCI DSS compliance. With some analysis, you may learn that it is cheaper in the long run to outsource your online payments to any number of companies like this. Some things to look out for include the following:

- How long does it take for cash to show up in your account?
- What is the chargeback (or equivalent) process like?
- Do you have any minimum transaction volumes to adhere to (dollars or numbers)?
- What is your liability if money is exchanged on a stolen account?

Stripe is a service built by developers, for developers, and attempts to address many of the issues with losing control through a great API and fantastic service. It's a great alternative to consider when looking at outsourcing card not present transactions. PCI DSS 3.0 has some new requirements for companies filling out the SAQ (specifically SAQ A-EP) that may bring some elements previously considered out of scope to be now in scope, so find your nearest PCI DSS expert to assist you in your unique situation!

PREDICTIONS

We wanted to take an opportunity to give you our thoughts on who the leaders might be when it comes to all this fancy technology. You are only allowed to read this portion of the book if you understand that the authors do not have a crystal ball, and we cannot tell the future. All we can do is pontificate and let things happen naturally. That said, here are a few thoughts on where things are going.

The smartphone has revolutionized how we live our lives. Not only is it annoying at times when our carriers refuse to work or they crash, but they have become the centerpiece for our digital world. Businesses know this, and they are doing everything possible to integrate with our devices in a way that both makes doing business with the consumer easier and can use analytics and personal service to increase ticket size and loyalty.

Because of this, the concept of a physical payment card may have a shorter lifespan than we think. As avid travelers, both of us would love to consolidate our loads by carrying fewer things in our pockets. Branden lives for the day when all he has to carry around is his smartphone and some chap stick and he can drive his car, pay for dinner, present on a fun security topic, and run through security to hop on a plane home without problems. Any way you can tap into this device without reinventing the wheel will probably work to your advantage. The key here is to not invest heavily in a technology that locks you in for the long haul. If Square goes belly up or never

has wide scale adoption for your customers, you want the ability to change your back end to accommodate some other scheme that will.

Along those lines, magstripe cards are at the end of their life. If you are a small business owner and are looking to technology to differentiate and power your business, skipping EMV and going right to the smartphone might be your best bet. Smartphones are proving to be a disruptive technology in the payment space, and we feel that they will become centric to payments like they have become centric to our digital lives.

TAXONOMY AND TIDBITS

In this section we will go over some basic taxonomy and definitions that are useful to those of you who feel a bit overwhelmed by what you are reading.

EMV

EMV, known as Europay, MasterCard, and Visa when you spell it out, is a global interoperability standard for payments meant to boost assurance in payment card transactions, specifically at the point of interaction (POI—the point where the information is read off of a customer's payment card). Some areas of the globe use Chip & PIN, but Chip & PIN is just one possible implementation of EMV. It's incorrect to use EMV and Chip & PIN interchangeably. The chip contains cryptographic algorithms to authenticate the card, and can be presented by themselves or used with a cardholder signature or PIN. PINs are optional and may not be used for every implementation. The majority of EMV transactions are done online, but offline transactions may happen in some geographies.

EMV is not yet globally implemented, so you may see different implementations in different places depending on where you go. Here in the US, a large number of processors simply cannot handle an EMV transaction today, even if their merchants have EMV-capable terminals for customer use. Most new terminals deployed today have the hardware to do NFC or contactless, EMV, and traditional swipe transactions, but the presence of hardware doesn't mean the presence of capability. If you are doing a technology refresh, go with the fully capable terminal and turn on the specific features as your processor is prepared to handle them. In Visa's case, you must be able to accept both a contactless and an EMV transaction at a payment terminal, *and process that transaction correctly* if a customer presents either of those at 75% or more of your total terminals to qualify for the TIP. There are other nuances with this program as well as other payment brands' programs, so check with your acquirer for details.

EUROPE VS THE US VS THE REST OF THE WORLD

Europe is a bit different than the US or the rest of the world when it comes to payments, including the EMV topic we discussed above. If you are a global business, you may be struggling with the differences between Visa and MasterCard in Europe and how those must be balanced with other global operations. In some cases, you

might find some requirements relaxed while others are more stringent. Rest assured, you will need to fully investigate these differences and it's a bit beyond the scope of this book. It's not that we don't want to take the time to write about it; it's more like the situation changes so frequently that we fear that this would become rapidly outdated and not useful to the readers.

In addition to requirements the penalties vary from place to place. You will not only find selective enforcement by some payment brands, but you will also find inconsistencies between each brand on how they enforce and how big the penalties are. We discussed some of those penalties earlier, and those should be used as a benchmark. You could end up paying more or even less than what you read about. Your acquirer may even refund some of those penalties, further invalidating the numbers. Do yourself a favor and check with your acquirer for any up-to-date information on global enforcement.

CUSTOMER EXPERIENCE

Lastly, let's explore the concept of customer experience. Our readers are largely charged with securing the infrastructure or complying with PCI DSS, but you will have another force that will dictate how you do business—the customer experience. This may dictate more about how you build and manage your payment systems than any other single force in your company. Many of the authors' customers are experimenting with all of the emerging technologies above with varying impact and success. Collecting money from a customer is an important part of the customer experience, but mostly to the business. You can expect that your company will begin to experiment more with this in the coming years in ways that will challenge your compliance status. Wireless terminals, kiosks, festivals, third parties, pay-by-cellphone, and integrated physical and virtual shopping experiences will impact everything you are doing with respect to payments.

Are you responsible for a call center? Consider using new software to direct callers to enter in their credit card number using the touch-tones on their phone. Doing so would take the majority of the Voice over Internet Protocol (VoIP) system out of scope, saving you money and time demonstrating compliance with PCI DSS.

Your challenge (you must accept it or you will find yourself looking for work!) is to meet the business folks with solutions to their payment acceptance problems. Don't put the "NO" in inNOvation, put the YES in succYESs (ok, that was a stretch but the "y" is silent...). Be creative. Learn how the business wants to operate. Learn what is important for them. Learn about the customer. Only then can you adequately put your brainpower to use to come up with solutions that solve both the business and security needs with the compliance requirement.

CASE STUDY

If you read the next case study and think, "Jeez, these sound crazy far out in the future," they might be for you but someone is doing this today. Concept stores, like concept cars, are fully functional and in limited use, but help businesses push

themselves to the fringe to figure out what works and what doesn't. In fact, some of this may be happening right under your nose!

THE CASE OF THE CASHLESS COVER CHARGE

Melissa's Mainstage is an intimate concert hall in SOHO, Manhattan. Melissa focuses on bringing local, regional, and some national acts through her venue but keeps the seating under 300 people. Her goal is to charge under \$10 for cover for every act, some times as low as just a few dollars, and donate some portion of the fee to a local charity for the needy. She has had a few instances of cash missing from the daily take as well a few customers wanting alternative payment options to cash when coming through the door. She accepts major credit cards once inside, but the door is cash only. She decides to take advantage of SIM-Based payments and on a trial basis and allows customers to send an SMS with the event name to a five-digit code that will place a charge on their cell phone bill. Since the dollar amount is typically very low, she doesn't see any major customer challenges. The first week she put the plan into effect, she had an amazing 97% adoption rate effectively reducing her cash risk by the same amount. She could track each cover by phone number and validate any transaction by simply checking the cell phone of the customer at the door. For the 3% that chose cash, she could choose to motivate them by making a cash cover charge slightly higher than the SMS payment. Or, as long as the adoption rate remains high, she could just opt to accept cash as a non-preferred alternative.

To fully understand Melissa's decision, you must realize that this change does not in any way affect her PCI compliance posture. She still does her normal compliance work for her bar and merchandise sales, but she now has a cashless solution for the door as well. She knew that by piloting and ultimately implementing this solution she would not create a PCI headache in the process.

SUMMARY

You get it and we get it—the world moves pretty fast just like Ferris Bueller said. By the time you are reading this book, these technologies will be even further down the path to maturation and probably embedded in more than a few retailers around the world. Remember that you can pilot or even adopt different payment methods that may or may not have any impact to your existing PCI DSS compliance. If you use a Qualified Security Assessor (QSA) or Internal Security Assessor (ISA), you will probably be educating him in this process as well. When you prepare for your meeting, be sure to give yourself plenty of time to walk through the technology and implementation, and bring an expert along with you that can answer any specific questions that assessor may have.

Go forth and experiment! The authors are consumers too, and we're pretty excited about the creative ideas you all will implement.

Myths and misconceptions of PCI DSS 19

INFORMATION IN THIS CHAPTER:

- Myth #1 PCI Doesn't Apply
- Myth #2 PCI Is Confusing
- Myth #3 PCI DSS Is Too Onerous
- Myth #4 Breaches Prove PCI DSS Irrelevant
- Myth #5 PCI is All We Need for Security
- Myth #6 PCI DSS Is Really Easy
- Myth #7 My Tool Is PCI Compliant
- Myth #8 PCI Is Toothless
- Case Study

As we previously discussed, Payment Card Industry Data Security Standard (PCI DSS), now updated to version 3.0, has transformed the way many organizations practice information security. While we've heard that something will take information security from the wire closet to the boardroom many times before, PCI actually accomplishes this for many organizations – both large and small. While it should be clear to our readers that following all of the PCI DSS guidance will not magically make your organization secure or prevent all incidents, the standard contains many of the common security requirements that are essential for protecting cardholder data and can be useful for other types of sensitive data as well.

As of today, "PCI DSS applies to all entities involved in payment card processing—including merchants, processors, acquirers, issuers, and service providers, as well as all other entities that store, process or transmit cardholder data." The aforementioned quote from the PCI DSS document reminds us that the applicability of PCI DSS is in fact nearly universal.

In this chapter, we look at common PCI DSS myths and misconceptions. We will also dispel those myths and provide a few useful tips on approaching PCI DSS.

Let's get to the myths.

MYTH #1 PCI DOESN'T APPLY TO ME

Myth #1 is pretty simple, but, sadly, very common: "PCI DSS just doesn't apply to us, because we are small, or we are a University, or we don't do e-commerce, or we outsource 'everything,' or we don't store cards, or we are not a permanent entity, etc."

More recent versions include “we use tokenization,” “we use EMV” (yeah right! – most of our US-based readers would say – even if that may change after 2015) or “we encrypt end to end.” “We outsource everything and thus have no PCI responsibilities” may in fact occasionally be true, but in most cases that is just that – a myth.

This myth takes over an organization and makes it oblivious to PCI DSS requirements and, almost always, to information risks and security requirements in general.

Another example is more blatant: health care providers have been so busy with Healthcare Information Portability and Accountability Act (HIPAA) that many became oblivious of PCI DSS arrival. A paper in “SC Magazine” called “PCI-DSS: Not on health care provider’s radar” [1] (notice the incorrectly hyphened “PCI–DSS” in the title...) reports:

However, since Medicare reimbursement is not at risk with PCI-DSS compliancy, it has been virtually ignored. It doesn’t help that major health care publications are openly misinterpreting the PCI-DSS standards for health care providers, with statements such as: “[Providers] do not have to worry about compliance with PCI standards... they aren’t storing any card numbers” [1].

A PERFECT EXAMPLE OF MYTH #1 AT WORK!

PCI DSS is not about storing cardholder data; it is about those who accept payment cards or capture, store, transmit, or process such card data. Want to guess whether most health care providers accept cards? Didn’t think so – the number is probably close to 100.00%, as most US readers can attest from their experiences. Indeed, the paper mentioned earlier [1] confirms: “In 2009, virtually all health care providers take credit cards—and virtually none of them are PCI compliant.” Now in 2014, the situation has barely changed. While HIPAA enforcement seems to have increased across health care providers, PCI DSS still remains “a big black hole” for many of them. Additionally, most such health care providers do not run a compliance program that can accommodate the needs of multiple regulations. They deal solely with HIPAA and adjusting the controls and practices to another regulation becomes fairly hard for them.

NOTE

Question: If I only accept cards from June to August each year and I only use a dial-up terminal, I am “safe from PCI,” right?

Answer: Wrong. Even though your scope of PCI DSS validation is very, very small, you are definitely subject to its rules because you – surprise! – accept payment cards. PCI DSS applies to those who “accept, capture, store, transmit, or process credit and debit card data.” If you do, it applies to you – end of the story. No myths can change that.

Interestingly enough, one of the data elements required to be protected under HIPAA is customer payment information, which often means “credit card data.” This means that HIPAA technically preceded PCI DSS when it comes to cardholder

data security! However, this doesn't stop health care providers from ignoring both regulations in one fell swoop.

NOTE

Question: If I use external tokenization and cardholder information never enters my environment, am I "PCI OK?"

Answer: Possibly! If your merchant agreement does not mention PCI DSS, none of your employees can see the data, and it is not handled anywhere on your systems, your PCI responsibility might be nonexistent.

The reality, as we mentioned earlier is pretty simple: PCI DSS does apply to your organization if you accept payment cards or capture, store, process, or transmit any sensitive payment card data (such as Primary Account Number (PAN)) with no exceptions. If the data touches your systems, they are in scope for PCI DSS assessment and, obviously, your organization has PCI DSS responsibilities. Whether you cure, educate, rent, offer, sell, or provide services doesn't matter – what matters is whether you charge! If you do, PCI DSS does apply. Hopefully, if you picked up this book while being unsure whether PCI DSS applies to your organization, reading this book convinced you that becoming compliant and secure is indeed in your future if you deal with payment cards.

Admittedly, different things need to happen at your organization if you have absolutely no electronic processing or storage of digital cardholder data compared to having an Internet-connected payment application system. The scope of compliance validation will be much more limited in the former case and so your PCI project will be much, much simpler. For example, if a small merchant "does not store, process, or transmit any cardholder data on merchant premises but relies entirely on third-party service providers to handle these functions" he is only responsible for validating a small part of PCI DSS. Specifically, he would be responsible for the parts of "Requirement 9: Restrict physical access to cardholder data" as well as a small part of "Requirement 12: Maintain a policy that addresses information security for employees and contractors" via a small self-assessment questionnaire (SAQ) Type A.

Let's explore this example in more detail. As we covered in Chapter 3 payment card brands such as Visa and MasterCard label merchants that process fewer than 20,000 e-commerce transactions a year or fewer than 1 million card present transactions as "Level 4." As you now know, such merchants currently are recommended to validate their PCI compliance using an SAQ.

In addition, as described in PCI DSS standards, if a merchant matches the criteria below, he is considered to be "validation type 1" and needs to fill the SAQ Type A (the shortest). The criteria are as follows:

- Merchant accepts ONLY card-not-present (i.e., eCommerce) transactions.
- Merchant does not store, process, or transmit any cardholder data on merchant premises but relies entirely on third-party service providers to handle these functions.

- The third-party service providers handling storage, processing, or transmission of cardholder data is confirmed to be PCI DSS compliant.
- Merchant retains only paper reports or receipts with cardholder data, and such documents are not received electronically.
- Merchant does not store any cardholder data in electronic format.

Explained simply, the aforementioned criteria describe a situation where a merchant accepts credit cards as payment, but does not have any electronic storage, processing, or transmission of cardholder data. Think about it for a moment! PCI DSS doesn't apply if you do not store, process, or transmit any card data on your premises (or your systems located off your premises such as outsourced, hosted or shared cloud systems) at all! This example highlights that fact that card acceptance is sufficient to make the merchant to fall under PCI.

The exact scope of its validation as covered by SAQ Type A, which can be obtained from www.pcisecuritystandards.org.

The merchant needs to validate part of Requirement 9 and part of Requirement 12. Specifically, sections of Requirement 9 cover the storage of physical media (printouts, receipts, etc.) that has cardholder data. For example, quoting from PCI DSS SAQ Type A [2]:

- 9.5 Are all paper and electronic media that contain cardholder data physically secure?
- 9.6 Is strict control maintained over the internal or external distribution of any kind of media that contains cardholder data?
- 9.6.3 Are processes and procedures in place to ensure management approval is obtained prior to moving any and all media containing cardholder data from a secured area (especially when media is distributed to individuals)?
- 9.7 Is strict control maintained over the storage and accessibility of media that contains cardholder data?
- 9.8 Is media containing cardholder data destroyed when it is no longer needed for business or legal reasons?

All of the above deal with the physical media such as printouts that may contain card data. The merchant is also subject to one section of Requirement 12, which covers the merchant's relationship with service providers that actually handle data (again, see PCI DSS SAQ Type A [2]):

- 12.8 If cardholder data is shared with service providers, are policies and procedures maintained and implemented to manage service providers, and do the policies and procedures include the following?
 - 12.8.1 A list of service providers is maintained.
 - 12.8.2 A written agreement is maintained that includes an acknowledgment that the service providers are responsible for the security of cardholder data the service providers possess.
 - 12.8.3 There is an established process for engaging service providers, including proper due diligence prior to engagement.

- 12.8.4 A program is maintained to monitor service providers' PCI DSS compliance status [2].
- 12.8.5 Information maintained about which PCI DSS requirements are managed by each provider and which are managed by you.

All of the above deal with the responsibilities of the third party that handles processing, storage, and transmission of data.

Overall, the choice is pretty simple: either you comprehend PCI DSS now and start working on security and PCI requirements or your acquirer will make it clear to you at some point when you won't have much room to maneuver.

A subtle point brought to life by an increasing use of EMV Technologies needs to be clarified: payment card brands may relax some of the PCI DSS validation requirements if the merchant uses new (and presumably more secure) payment methods; however, merchants will still be required to maintain PCI compliance at all times. Now in 2014, many merchants dread the coming "liability shift" of 2015 (officially known as "Global Point of Sale Counterfeit Liability Shift") when merchants not installing EMV systems may become liable for fake cards transactions.

MYTH #2 PCI IS CONFUSING AND AMBIGUOUS

MYTH #2 IS Just as pervasive: PCI is confusing and not specific. At first, it might seem like it is true: after all, this whole book is written about it! However, this is actually a myth, despite the fact that people who didn't invest enough time into learning about PCI compliance might be more than a little confused about it.

In addition, this myth seems to be purposefully propagated by some people to "muddy the waters" and thus to make PCI DSS seem impossible to achieve and thus not worthy of even trying. Said people frequently have something to sell to merchants, something that presumably simplifies PCI compliance to nearly nothing. For example, when confronted with the need to change their business process to avoid storing the card data (simpler task as you now know) or with the need to secure card data (a harder task compared to not storing it), many smaller organizations go into the "ostrich in the sand" mode and try to pretend the problem is unclear, unsolvable, or confusing, instead of tacking it head on.

Namely, those under its influence often proclaim things such as:

- "PCI just confuses us – we can't do it."
- "We don't know what to do, who to ask, what exactly to change."
- "We don't know whether we are compliant and hiring somebody to tell us is expensive."
- "PCI is confusing; until you give us something better, we will not do anything to protect the data."

Sometimes it also devolves into the following:

- "Just give us a simple task list and we will do it. Promise!"

The reality is quite different: PCI DSS documents explain both what to do and then how to validate it. Apart from people who propagate this myth, you just need to take the time to understand the “why” (the spirit of the standard and cardholder data security), the “what” (the list of PCI DSS requirements), and “how” (common approaches and practices related to PCI).

PCI is actually much easier to understand than most other existing security and risk management frameworks and regulatory guidance in spite of its length and breadth. Looking at some of the advanced information risk management documents (such as ISO27005 “Information security risk management” or NIST 800-30 “Risk Management Guide for Information Technology Systems”), with their hundreds of pages of sometimes esoteric guidance, is a refreshing reminder that PCI DSS is, in fact, pretty simple and straightforward.

Most of the gray area talk you hear about PCI DSS typically comes from companies that are trying to find creative ways not to do anything to their business or their technology or take risky shortcuts with other peoples precious data. Most of these can be solved by removing the emotion from the situation, and logically laying out the costs and risk associated with making a change versus staying in violation of the rules until caught and punished.

Let's compare what PCI says and what some other guidance documents say:
PCI DSS in Requirement 5 (quoted from PCI DSS 3.0) states:

“5.1 Deploy anti-virus software on all systems commonly affected by malicious software (particularly personal computers and servers).

5.1.1 Ensure that anti-virus programs are capable of detecting, removing, and protecting against all known types of malicious software.

5.1.2 For systems considered to be not commonly affected by malicious software, perform periodic evaluations to identify and evaluate evolving malware threats in order to confirm whether such systems continue to not require anti-virus software

5.2 Ensure that all anti-virus mechanisms are maintained as follows:

Are kept current,

Perform periodic scans

Generate audit logs which are retained per PCI DSS Requirement 10.7.”

ISO27002 states [3]: “Precautions are required to prevent and detect the introduction of malicious software, such as computer viruses, network worms, Trojan horses, and logic bombs.”

NIST 800-53 document “Recommended Security Controls for Federal Information Systems,” itself a 174 page tome, states [4] “The information system implements malicious code protection.” Additional NIST documents, such as 800-83 “Guide to Malware Incident Prevention and Handling: Recommendations of the National

Institute of Standards and Technology,” do provide useful additional guidance on malware protection, but that is another 101 pages to read (!).

The aforementioned example demonstrates that even though PCI DSS needs some focused attention from the merchants, it is not confusing, but more specific than other industry security guidance. While the original PCI creators had BS7799/ISO17799 in mind, their standard ended up being much simpler than its ISO-created inspiration, but lost of few things that people consider important (e.g., all the security project management requirements).

Still, there are definitely areas in PCI security guidance that can be made more specific. For example, when PCI DSS document was updated from version 2.0 to version 3.0, the updated document contained dozens clarifications and explanations, brought up during the review phase (see the PCI DSS Summary of Changes document available on the PCI Council Web site for details).

As PCI DSS grows up, all the remaining ambiguity should be reduced through the efforts of participating organizations and by adopting to the needs and requirement of the merchants as well as the evolution of the threats.

Finally, security cannot and will not ever be reduced to a simple checklist. Even today some criticize PCI DSS for being a manifestation of “checklist security,” which does not account for individual organization’s risk profile. PCI guidance is as close to a checklist as we can get without actually leading to increased, not reduced, risk.

MYTH #3 PCI DSS IS TOO ONEROUS

The next myth, Myth #3 is closely related to the above: PCI DSS is just too hard. Sometimes, it becomes too expensive, too complicated, too burdensome, just too much for a small business, about too many technologies or even simply “unreasonable” and “too much security.”

Sometimes this myth is close to reality, but not for reasons people think. “PCI is too hard” becomes true if merchants are treating their card data in a particularly negligent way. Think giant text files with unencrypted PANs on a Web site or payment terminals allowing remote access open with no password!

For example, we introduced a concept of “flat network” in Chapter 4. To remind, it means a network that is not segmented by firewalls or filtering routers into zones or segments. Such network design means that if a single system handles card data, the whole network is in-scope for PCI DSS. Yes, that means all servers, all desktops, all network devices and everything else connected to it becomes subject to all PCI DSS requirements! Such situation, even if relatively rare nowadays, has been known to happen at large and medium merchants and thus make PCI DSS compliance very hard as the scope grows from a single system or a handful of systems to hundreds or thousands of systems. Every such system will need to be scanned for vulnerabilities, have its security configuration verified (these include the password length and complexity, use of encryption, etc.), have logging enabled and monitored, and so on.

In this case, “PCI is hard” will not be a myth, but PCI DSS guidance will not be the reason to blame: the operators of such a network will be.

WARNING

If you think PCI is too onerous, please stop and consider this: do you think that wearing a seat belt while driving is too onerous? How about brushing your teeth every day? How about actually operating your business?

Tasks unlike ones you have been doing are often first seen as onerous. When you were 3 years old, it is very likely that brushing your teeth was seen as a huge task – and an annoying one as well.

So, if you happen to think PCI and data security is too onerous, you simply need to start paying attention to it – and it will stop being onerous after you get good at it.

Similarly, if a simple change in a business process will lead to removal of stored cardholder data, continuing to operate without such change and engaging in attempts to protect such data, rather than remove it will be shortsighted and lead to PCI being indeed “too hard.” As we say in the book “delete the data—and then secure what you absolutely CANNOT delete.”

On the other hand, the reality is that PCI DSS exemplifies common, baseline security practices, which every organization needs to take into account when planning their IT and business operations. PCI only seems hard if you were not doing *anything* for the security of your data before; hopefully this book will help to make PCI DSS easier and actually useful for your organization. PCI might not be easy for a large, distributed organization, but it is clearly much easier than creating and running a well-managed security program based on a good understanding of your risk across all types of data.

As observed by one of the authors, people who complain about PCI DSS being too hard and emitting loud calls for “Make PCI Easier!!!” are often split into two camps:

- “Please, please make PCI easier by letting us skip the requirements; or, better, they just let us ‘JUST SAY YES ON THE SAQ!’ camp. We can call them “laggards” or “losers” for extra hilarity.
- “We know that our security program makes us PCI compliant; please make it easier for us to prove it!” camp. We can call them “leaders” or “snobs,” if you wish.

As you can guess, the organizations that fit into the first camp and those that fit into the second camp are very different. While some in the first will miss the joke in *ScanlessPCI* (an old joke site set up by a band of security experts to poke fun at some of the organizations that ignore security), the second camp is often concerned with relating their “risk-focused” approach to PCI’s mostly “control-focused” approach.

It is possible to make PCI easier even for those in the first camp, those people who just “want it gone”: make doing the right thing easier for them (while making doing the wrong thing harder): not storing card data, outsourcing processing, using

tokenization, and the like. For example, a lot of merchants store card data because they are under the mistaken impression that such data falls under the “Financial Rule of Thumb” of storing financial records for 7 years. Such actions are making PCI DSS much more difficult or, in the case of storing Card Verification Value 2 (CVV2) or other prohibited data, impossible. Even for chargebacks, which obviously don’t go back 7 years, storing the PAN and other sensitive data is just not needed today.

On the other hand, while in the second camp, one sometimes hears things like “we have a good security program and we manage our risk well!—why should we spend time on that PCI thing? We are probably in good shape already!” These organizations are likely doing a good job with security and want to use all that too quickly “prove compliance.” In this case, making PCI easier will include making it easier to assess, validate, and prove compliance and overall make the whole “assessment experience” a little less painful. Still, it will not be harder than building that risk management program that they already have built.

So, as we mentioned, you can make PCI harder for yourself by making the wrong decisions. For example, developing your own Web application complete with credit card processing will increase your PCI scope likely beyond your ability to handle. On the opposite, using a third-party checkout service will do just the opposite and make PCI and data security easier. Review the previous section on Myth #2 and notice that not touching the cards will make your PCI experience so much easier – without going into the application security esoterics such as OWASP guides, SQL injection, cross-site request forgery, and other things better be left to those who enjoy writing books about PCI DSS compliance.

MYTH #4 BREACHES PROVE PCI DSS IRRELEVANT

Myth #4 seems mostly driven by the media: it claims that “Recent card data breaches prove PCI irrelevant.” We suspect it stems from the fact that reporting failures and other “bad stuff” typically draws more listeners, readers, and watchers compared to reporting successes and thus attracts more media attention. Target retailer breach was the largest for a few months since late 2013 – until the Home Depot breach of 2014, that is!

Note how much press time the catastrophes, corrupt politicians, weather anomalies, and various technology failures get; and the bigger the better!

However, it encourages some organizations to develop a negative, destructive mindset and thus to do a bad job with PCI DSS and data security. As a result, they would suffer from a devastating data breach, which is more than a little ironic because they were trying to “focus on the breaches, not on compliance.” Breaches of organizations that validated their PCI compliance should get more people to focus on security as well as on maintaining compliant and secure state, not the other way around. If a similar organization to you was breached, you now need to do a better job to not end up like them.

It is worthwhile to mention that QSAs carry part of the blame as well: cases where blatant security mistakes, severe configuration weaknesses, forbidden data

storage, and even compromised systems were missed by the “easy grader” assessor who focused much more on writing a report quickly enough to make his margin and not enough on actually collecting the data for the report. In selecting a QSA – your “mandatory security consultant” – it helps a lot to focus on value delivered to an organization and not simply on price. Try to squeeze every penny of valuable security advice during that face time in the assessment process. In some cases, you can “get your own QSA” – have somebody from your organization certify as “ISA” – a new Council certification for internal security assessors.

WARNING

It is often assumed by security professionals that people outside of security industry will not take advice to protect data from mainstream media. Sadly, people often do and sometimes even IT people and IT managers do so. In light of this, the author team would make this into an explicit warning.

Dear reader! Newspapers, magazines, and even IT “trader rags” and information security press are NOT a reliable source of security guidance, whether for technology or for policy security guidance. Read them to know the news, but go to experts for detailed guidance on how to secure your data.

Today the media is much obsessed about “cyber war” with a lot of truly idiotic advice being passed around this topic. If you made it to RSA Conference this year, you probably noticed that the conversations had changed from basic compliance or antivirus to threats intelligence and advanced security analytics. The discussions are starting to happen, which means the money and behavior will follow in time.

Again, the reality is exactly the opposite: data breaches remind us that basic security mandated by PCI DSS is necessary but not sufficient. You have to start from the basics before you can advance in your security education. As you learn more about security, you usually come to realize that nothing guarantees breach free operation. Let’s pick a few examples from PCI DSS and check whether they are a good idea as well as whether they guarantee the absence of breaches. [Table 19.1](#) shows excerpts from PCI DSS as well as their relation to data protection and data breach prevention.

[Table 19.1](#) allows us to conclude that PCI DSS controls are necessary to prevent card data loss and theft, but few of them can be considered sufficient.

NOTE

Recent breaches remind us that PCI DSS is insufficient – and it is. Only ongoing and adaptable risk management program with quality strategy and execution can be considered an end-state, but it is an ongoing end state that never really “ends.”

Please remember – you are never “done” with information security!

Finally, one of the authors’ colleagues likes to say that every breach proves that PCI DSS is even more necessary. PCI DSS is a great start for security, but a really bad finish, as we discover in the next myth.

Table 19.1 PCI DSS Requirements for Data Security

Requirement Number	PCI DSS Requirement	Important for Data Security?	Guarantees Lack of Card Data Loss?
2.1	“Always change vendor-supplied defaults before installing a system on the network—for example, include passwords...”	Yes, default password is a frequent avenue for attackers	No
2.2.4	“Remove all unnecessary functionality, such as scripts, drivers...”	Yes, sample scripts, etc are often abused to get access to systems	No
2.3	“Encrypt all non-console administrative access”	Yes, guessing an admin password exposes the entire systems to attackers	No
3.2	“Do not store sensitive authentication data after authorization”	Yes, theft of such key data allows account abuse	Yes! No data means no data breach
5.1	“Deploy anti-virus software on all systems commonly affected by malicious software”	Yes, viruses, spyware and other malware leads to theft of card data	No, despite antivirus tools, the malware can still spread to the systems
7.1	“Limit access to system components and card-holder data to only those individuals whose job requires such access”	Yes, stolen access credentials makes “hacking” the system very easy	No, as other means of breaking in can be used by attackers
8.2.3	“Use passwords containing both numeric and alphabetic characters”	Yes, such passwords are much harder to guess and then to be used to steal the data	No, even complex passwords can be guessed or systems compromised without the passwords

MYTH #5 PCI IS ALL WE NEED FOR SECURITY

Myth #5 is probably the scariest one of all: PCI is all we ever need to do for security. People in the grasp of this myth would proclaim things that will shock every security professional; for example:

- “We have handled PCI – we are secure now.”
- “We worked hard and we passed an ‘assessment’; now we are secure!”
- “Our QSA told us we are secure.”
- Or even, in its more extreme form,
- “I filed my PCI compliance documents; now I am compliant and secure for a year.” (insanity, you say? Such views are heard even now in 2014 when these words are being written).

The above maxims remind the authors of how many executives in the late 1990s proudly proclaimed, “I have a firewall, I’m secure” and then refused to pay attention to “overblown” security concerns. At the very least, a firewall might block some suspicious connections, while report on compliance paperwork on its own does not provide any protection – and in fact only introduces additional risk of fire.

Recently, one of the authors was shocked to read the following on what purports to be a security blog post that talked about the now-infamous Heartland Payment Systems data breach: “Is complying to PCI not enough anymore?” It seems that even security professionals can somehow fall victim to this myth. Another common manifestation is an organization that doesn’t pay much, if any, attention to data security is being suddenly jabbed by PCI DSS requirements. More often than not senior management will “make” IT do “that PCI security thing.”

It often leads organizations to focus on “pleasing the assessor” and then forgetting that a happy assessor does not mean that your organization is protected from information security risks.

Moreover, this myth is actually wrong on multiple levels! First, validating PCI DSS via an assessment or self-assessment does not mean that you are done with PCI DSS as you now need to maintain compliance – something that needs to happen daily (!). Also, it certainly does not mean that you are done with security. In addition, it also doesn’t mean that you are secure, just that you validated PCI compliance and hopefully made an honest step toward reducing your risk; now you need to maintain your compliant status and data security! As a reminder, while your QSA validates your compliance and is responsible for the Report on Compliance, merchant is solely responsible for maintaining compliance status between assessments.

Again:

Validating PCI compliance via on-site assessment or self-assessment does NOT mean that:

- You are “done with PCI” and now can ignore it.
- You are “done with security” and now cannot do anything to protect your data.
- You are “secure” or as even secure as you need to be.

The reality is again different: PCI just mandates minimum security, not maximum or optimal. And PCI validation attests that such minimum security was present at the time of assessment (an astute reader will make a note that the word “present” needs to be replaced with “shown to or identified by a QSA,” but we will not succumb to such cynical urges).

PCI is basic security; it is a necessary baseline, a low watermark, which was never meant to be the “end state” of guaranteed secure data. No external guidance document, even well-written and followed with utmost diligence, can guarantee that – just as excellent police work can never guarantee “crime-free” environment. People don’t expect police, prosecutors, and laws to “end crime” – so why do some people think that QSAs, approved scanning vendors (ASVs), security vendors, card

brands, and PCI DSS document will end cybercrime? This is a useful thought to keep in your head as you are finishing that PCI validation.

WARNING

If you hear somebody lament the fact that he was PCI compliant and then got breached and had all card data stolen, please pause before bashing PCI DSS for this unfortunate course of events. It is hard to say for sure without having the details, but in most case the authors have seen in the field, such breach was the fault of the merchant and not of PCI DSS, PCI Council, or any other regulatory entity. In fact, even when they say “they were compliant,” they rarely mention WHEN in fact they think they were compliant.

Finally, PCI is about cardholder data security, not the rest of your private or regulated information, not your organization intellectual property, not identity information such as Social Security Number (SSNs), not the availability or fewer Internet-facing services or employee productivity. It only covers confidentiality, and not availability of cardholder data. These quick examples show that there is a lot more to data security than PCI DSS, and there are clear areas where PCI does not focus – for a good reason.

Specifically, perfect PCI DSS compliance and validation will not:

- Make your “out of scope,” noncardholder data such as SSNs secure.
- Make your trade secrets and other intellectual property secure.
- Make your “out of scope” systems and networks secure.
- Make your card data, other data and information systems more reliably available.
- Make you automatically compliant with any other international, national, state, or industry regulation.
- Make employees use Internet productively.

It might only minimally affect:

- Security of your other data – the implemented network safeguard might help and so might the change of organization culture due to security.
- Your readiness to address other regulations.

Thus, you are certainly not “done with security” even if you maintain ongoing PCI compliance. For example, one of notable PCI QSAs likes to say that you likely need “PCI+” or even “PCI++” to deal with risks to your systems and data today.

MYTH #6 PCI DSS IS REALLY EASY

The next myth, #6, is the opposite of myth #4: PCI is easy: we just have to “say Yes” on a questionnaire and “get scanned.” As merchants become more familiar with PCI DSS, some start to feel that PCI is not that scary – it is about getting a mysterious “scan” from an entity called “an ASV,” then answering “Yes” to a bunch of lengthy

questions and paying an “outrageous” fee of \$8.95 with their monthly merchant account bill. Given that their view of PCI DSS as “annoying but tolerable” such merchants have fully succumbed to this myth. Their misconceptions are expressed in claims that PCI is about getting a scan and answering some questions or that it is a paperwork exercise or even that it can be bought for \$8.95 monthly compliance fee (or, occasionally, a \$100/year noncompliance fee) ...

For merchants and service providers that don’t have to go through an on-site QSA assessment, PCI DSS compliance is indeed validated via external vulnerability scanning by an ASV and via filing an SAQ consisting of 13–230+ questions. However, it is worthwhile to mention that there is some work involved before many of the merchants can truthfully (!) answer “yes” to those questions and would be able to prove this, if requested by their acquiring bank.

WARNING

PCI DSS is not easy. It is definitely not easy for a large company that needs to collect all the evidence for compliance validation from different systems and then maintain such evidence between assessments. However, these activities, if not easy, are actually useful for security and overall manageability in that company.

PCI DSS is pretty easy if all card processing is outsourced to a reliable and secure processing provider who does not allow you to touch the data. Some external tokenization solutions offer similar benefits for simplifying PCI compliance.

Here is an example: Requirement 1.1.2 mandates a “1.1.2 Current network diagram that identifies all connections between the cardholder data environment and other networks, including any wireless networks.” We show a few examples of such diagrams in Chapter 4. Answering “Yes” to this question entails:

1. Having a diagram available,
2. Making sure that the diagram is indeed current and all network connections are reflected on it,
3. Knowing the locations of cardholder data (itself a massive project at many organizations!),
4. Verify that there are no other locations of cardholder data (PCI DSS 2.0 places additional emphasis on such discovery efforts),
5. Confirming that it shows all connections to all locations of cardholder data,
6. Knowing which wireless networks are deployed and where, including those deployed by divisions and other business units (and possibly rogue as well),
7. Maps out connections from networks and systems that can manage systems housing cardholder data.

Clearly, this takes more than reading a book (we suggest “PCI Compliance”, 3rd edition!) or clicking “Y” on a computer. This is what turns minutes into months and simple tasks into projects.

A slightly simplified reality is that a typical small merchant who processes cards online would at least need to do the following:

- a. Get a network vulnerability scan of the external systems from an ASV, resolve the vulnerabilities found, and then rescan to verify.
- b. Do the things that the SAQ questions refer to and maintain evidence that they were performed.
- c. Answer the questions affirmatively (providing details were needed) and retain proof of that validation.
- d. Keep up with periodic maintenance and other requirements until you no longer wish to accept credit cards, that is, maintain compliance.

In other words, achieve PCI DSS validation and then maintain PCI DSS compliance for as long as you plan to accept cards. You can only answer “yes” if you have grounds for saying “yes” on the questionnaire and can prove it, even with no assessors or acquiring banks looking over your shoulder.

NOTE

Question: My management told me to ignore everything, including security hardening of my servers and even response to an ongoing hacking incident and go focus on “pleasing the QSA.” What should I do?

Answer: Apart from changing a job, you mean? Sorry, bad joke. What you need to do is try to tie all the key security things you know you need to do (we are assuming here that you actually know what those things are) and tie them to PCI DSS requirements. Under such PCI umbrella, the management should be more accepting of what you actually need to do for security. Result: achieve both security and PCI DSS compliance. Some people might object to such tactics as “gray area” or “unethical,” but there is nothing unethical in use in security regulation – PCI DSS in this case – to improve security.

Specifically, even on the vulnerability scanning side, the typical perception that “get a PCI scan and you are done” is essentially misguided. PCI DSS requires you to run both internal and external network vulnerability scans at least quarterly (in reality, twice a quarter since you’d need to fix the vulnerabilities and then rescan to confirm it!) as well as after every major network change. Internal scans can be run by in-house security staff, while the external scans must be performed by an ASV, and these are then used to satisfy your PCI Validation Requirements and are submitted to your acquiring bank. By default, all Internet-facing IP addresses are “in-scope.” For larger businesses, companies can work with their QSA and ASV to reduce the scope of that external footprint if certain controls are in place and enforced.

Furthermore, the specific requirements placed on the depth of such scans. There is also a requirement to remediate all vulnerabilities, scored higher than 4.0 with CVSS for external scans or all “High” vulnerabilities as defined in PCI DSS Requirement 6.1 are resolved for internal scans. In essence, this is not “scan and done,” but “scan-fix-maintain” lifecycle.

NOTE

It is a common misconception that the PCI DSS scanning validation requirement applies only to systems that process card data. If you are subject to PCI DSS scanning requirements (which happens when you have systems connected to the Internet), all of your externally visible systems are subject to security scanning (by an ASV) as well as those systems involved in processing, storing, and transmitting the data and the ones “directly connected” to them (by an internal team or a consultant). Many organizations have a lot of challenges scanning systems with cardholder data especially when such systems are virtual or located at other environments, or not under full control of the merchant.

MYTH #7 MY TOOL IS PCI COMPLIANT THUS I AM COMPLIANT

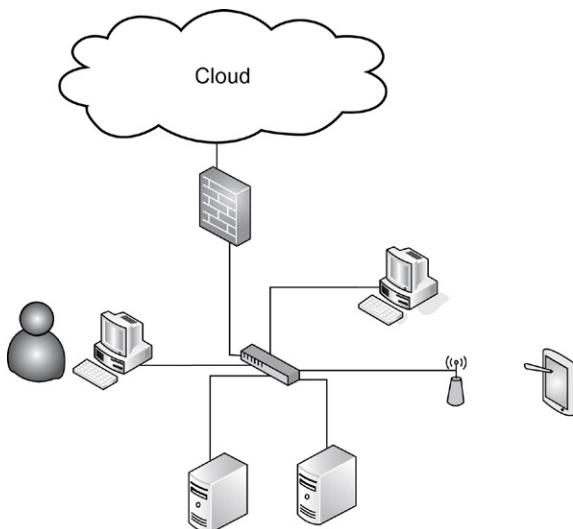
Myth #7 is in believing that your network, application, tool is PCI compliant with the resulting conclusion that this achieves compliance for your organization. This myth manifests itself in statements from merchants such as “My payment application vendor said his tool is ‘PCI compliant’ or “They put together a network and it is PCI compliant.” However, no tool can make you compliant. In fact, people often confuse PA-DSS certified application with PCI DSS compliant organization, which literally have little to do with each other, even though both come from PCI Council. PCI DSS 3.0 states that “Use of a Payment Application Data Security Standard (PA-DSS) compliant application by itself does not make an entity PCI DSS compliant, since that application must be implemented into a PCI DSS compliant environment and according to the PA-DSS Implementation Guide provided by the payment application vendor.”

Just to remind you, 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” (PCI Council Web site [5]). On the other hand, PA-DSS is “to help software vendors and others develop secure payment applications that do not store prohibited data, such as full magnetic stripe, CVV2 or PIN data, and ensure their payment applications support compliance with the PCI DSS” (PCI Council Web site [5]). The former applies to an entire organization while the latter applies to a payment application only.

TOOLS

PA-DSS list contains all of the validated payment applications that can be used by merchants. It is not uncommon for an acquirer to refuse to onboard merchants that use applications which are not on the list or even the versions of the applications which are not on the list.

However, using this list will get you a list of tools, but you still must deploy the tools in the manner prescribed by the DSS document as well as maintain them as required by PCI DSS.

**FIGURE 19.1**

Is This Network PCI Compliant?

In reality, there is no such thing as “PCI compliant tool, application, configuration or network,” regardless of what a vendor’s marketing department says. PCI DSS compliance applies to organizations only. You can struggle toward, achieve, and validate PCI DSS compliance only as an organization. Using PA-DSS compliant, application is only a small piece of the entire puzzle.

Despite that, the authors have been asked multiple times by various industry colleagues about certain pieces of IT infrastructure being PCI compliant. Here is one recent example – [Figure 19.1](#).

[Figure 19.1](#) was shown to one of the authors with the question of “Is this PCI compliant?” Even though some people will start debating this with crafty arguments for “yes” and “no,” the immediate (and correct!) response was “no way to tell.” There are simply too many things that can make or break PCI DSS compliant status of an organization. For example:

- What are the password policies on these servers? Are the compliant with the whole host of requirements 8.1 and 8.2, such as “8.1.4 Remove/disable inactive user accounts at least every 90 days” or “8.2.4 Change user passwords/passphrases at least every 90 days.”
- Is there logging performed? Is such logging as good as Requirement 10.2 mandates. Namely, is logging sufficient to reconstruct the prescribed events.
- Is vulnerability scanning done? Are scans “run [...] at least quarterly and after any significant change in the network?” Are discovered vulnerabilities remediated?

- Is file-integrity monitoring software (now referred to as “change detection software” in PCI DSS 3.0) deployed to “alert personnel to unauthorized modification of critical files, and to perform critical file comparisons at least weekly” (see Requirement 11.5)?
- Moreover, are the security policies “established, published, maintained, and disseminated” (Requirement 12.1 in PCI DSS)? Or is there “a formal security awareness program to make all employees aware of the importance of cardholder data security” prescribed by PCI DSS Requirement 12.6? What about the operational procedures for all the security tasks?

WARNING

Security policy is pretty darn important for your PCI DSS compliance program (since it is mandated in Requirement 12 which we cover in Chapter 6) as well as for data security. However, here is the dirty truth about policy: it does nada, none, zero to secure the data unless it is actually followed in reality and enforced by security technology and process.

Furthermore, the authors hypothesize that this is exactly why policy development is listed in Phase 6 of “PCI DSS Prioritized Approach” [6]. Security professionals say “policy comes first,” but policy needs to be actually implemented have ANY impact on data security and risk.

The above lists only a few of the reasons for why the above pictured environment might or might not be part of a PCI DSS compliant organization.

Thus, “no way to tell” is the only genuine response. There is no way to judge PCI compliance on just isolated servers, pictured on a diagram.

NOTE

Your networks, applications, or tools are not PCI compliant, even though the payment application can be PA-DSS compliant. PCI DSS compliance, however, applies to the entire organization. PCI DSS also does not have any “partial compliance” – there is only “compliant with all relevant requirements” (was clear evidence and league are so their relevance) or “not compliant.” In some cases, payment brands or acquiring banks may allow for you to complete certain phases of the Prioritized Approach, but those are either associated with technology advancements and implementations or special one-off cases.

Answering the question is only possible “in bulk,” considering all the conditions, criteria, and requirements, and the only way to achieve this is to follow your own path through the PCI DSS requirements. The “PCI DSS Prioritized Approach” (updated to PCI DSS 3.0 recently) lists the recommended, risk-derived order for following the requirements that can be helpful on your journey. The document can be found on the PCI Council Web site [5]. [Figure 19.2](#) shows an example from the PCI Council document called “Prioritized Approach to PCI” [6].

As we mentioned, PCI DSS combines technology, process, policy, awareness, and practices as well. For example, Requirement 12 covers security policy, incident

Milestone	Goals
1	Remove sensitive authentication data and limit data retention. This milestone targets a key area of risk for entities that have been compromised. Remember – if sensitive authentication data and other cardholder data are not stored, the effects of a compromise will be greatly reduced. If you don't need it, don't store it.
2	Protect the perimeter, internal, and wireless networks. This milestone targets controls for points of access to most compromises – the network or a wireless access point.
3	Secure payment card applications. This milestone targets controls for applications, application processes, and application servers. Weaknesses in these areas offer easy prey for compromising systems and obtaining access to cardholder data.
4	Monitor and control access to your systems. Controls for this milestone allow you to detect the who, what, when, and how concerning who is accessing your network and cardholder data environment.
5	Protect stored cardholder data. For those organizations that have analyzed their business processes and determined that they must store Primary Account Numbers, Milestone Five targets key protections mechanisms for that stored data.
6	Finalize remaining compliance efforts, and ensure all controls are in place. The intent of Milestone Six is to complete PCI DSS requirements and finalize all remaining related policies, procedures, and processes needed to protect the cardholder data environment.

FIGURE 19.2

PCI DSS Prioritized Approach

response practices, security awareness, and other nontechnical safeguards and controls. Don't focus on isolated pieces; rather, build the big picture, track a path, and move ahead toward PCI DCC compliance and data security.

MYTH #8 PCI IS TOOTHLESS

Myth #8 is simply a view that “PCI DSS Is Toothless.” This myth shows a completely wrong worldview of PCI DSS and security; a dangerous delusion that is also wrong on several levels.

First, it embodies the view that data security measures are only deployed due to regulatory pressure, such as from PCI DSS, and not from genuine need to reduce risk to information and transactions. This myth is often used to justify not doing anything about data security with merchants believing – erroneously that if even they are breached and then also found noncompliant, their business will not suffer. Similarly, people read the popular media and then ignorantly claim that companies are breached and then continue being profitable so they should not care about PCI and security. Just ask that question of a recently fired CSO and see what he says about

“no impact on an organization.” Finally, they claim that compliance costs more than noncompliance without doing any research into this subject, which then leads to faulty decision making.

Honestly, such views are often held by merchants who are used to just accepting the risk of card data theft – until their businesses are no more (in some cases). In fact, in the pre-PCI era it made good sense to accept such risk because it was not the merchant’s loss, but the issuing banks that would need to reissue the stolen cards. Here is a commonly utilized conceptual risk formula¹:

If we apply it to this situation, we’d realize that before PCI the merchant’s risk was indeed exactly 0 and absolutely didn’t depend on threats that the merchant faced, as well as vulnerabilities his environment had. PCI DSS transferred the risk back to where it should belong – wherever the data is lost or stolen. Thus, one can say that PCI DSS created the risk for merchants to motivate them to protect the data as well as educate them about Data Security.

Second, in addition to being a wrong mindset, it is also simply wrong. PCI DSS noncompliance and resulting loss of cardholder data packs a lot of bite which includes fines, possible lawsuits, mandatory breach disclosure costs, investigation costs, possible card processing rate increases, cost of additional security measures, and cost of victim credit monitoring. It is really not only about the fines; among the above “teeth” the following were actually frequently observed:

- Breach disclosure costs: Although it is not the direct consequence of PCI DSS noncompliance, these costs always result from the breaches, frequently caused by blatant disregard for both data security and PCI compliance.
- Consulting fees: Companies breached have to pay guys like us to whip them into shape quickly. We’re not cheap.
- Legal costs: In case of suffering a breach and then being found noncompliant, the chance of a lawsuit mentioning negligence is higher.
- Fines: The card brands are not very vocal about fines, but public reporting of recent large breaches did mention fines ranging all the way up to \$10 m.
- Publicity: The experts can debate how the negative publicity maps into actually dollar losses, there is no denying that the name “Heartland Payment Systems” was not known outside of a narrow segment of payment professionals and now the query on Google for its name finds more than 240,000 pages related to the massive breach. What about all the jokes about “not becoming the next Target?” with puns about the Target 40 million card breach of 2013? Do you really want your company name to become a synonym for “data breaches?” Exactly, neither did they!

To top it off, a victim merchant can be labeled “Level 1” and thus subjected to an annual QSA assessment – at their own expense. Admittedly, not every breach will incur all of the above, but some are simply unavoidable.

¹As this formula is conceptual and not strictly mathematical, it is often written with many different variations, many including the concept of “Probability” or “Likelihood” of an event occurring.

NOTE

If breached and then found noncompliant, your business will suffer. The exact amount of loss that you will take cannot really be predicted upfront, but it will be there. In the very worst case, your business will simply be gone, as happened with some merchants over the years. Or, its name will become synonymous with “data breach,” just as Heartland Payment Systems and Target did. If compliance costs look to be too much for you to handle, your only course of action should be to completely outsource your payment processing to someone else. Besides, in what right mind would a retailer want to be a payment processor?

Overall, it is much more useful to think of customer and cardholder data protection as your “social responsibility” and not as something you do because of some scary “PCI teeth” somewhere! The companies talk about corporate social responsibility (CSR),² but often forget that caring for the private data of their customers that was entrusted it to them due to the current predominant paradigm – use of a payment card with a magnetic stripe – is the simplest form of corporate social responsibility. What more fits the definition of “embracing responsibility for the impact of their activities on the environment, consumers, employees, communities, stakeholders, and all other members of the public sphere,” than caring for those precious bits and bytes from the magnetic stripe?

CASE STUDY

Next, we present a case study that illustrates what is covered in this chapter.

THE CASE OF THE CARDLESS MERCHANT

Sometimes, merchants unknowingly accept PCI-related risk and can be left facing substantial fines.

Payton’s “P-Funk All Stars” Party Palace is a recent startup company attacking the ever-popular children’s birthday party celebration location market. Payton recently left her position at a retailer to start her company, and her previous experience included a basic understanding of PCI and that compliance was important and mandatory. When Payton was looking for the ability to accept major credit and debit cards, she made sure to find an Independent Sales Organization (ISO) that offered management of her POS devices.

Upon receiving the contract from her ISO, she noticed that she would still be filing for her own Merchant ID, but one of the ISO’s divisions, PayProcess Express,

²Sometimes corporate social responsibility (CSR) is defined by stating that the business must accept all responsibility for the results of its activities on the nature, environment, customers, employees, and even public at large. Moreover, commercial organizations must actually work in the public interest by encouraging community and choosing to discontinue practices that harm the public sphere or can be seen as unethical. CSR concept is surely controversial, but evidence suggests that businesses are more often pushed to focus on things other than profit, whether it is good or bad.

would be leasing and managing her POS equipment. For an extra fee, they also agreed to perform settlement services and reconciliation reports. Payton's funding was not substantial, and preferred the transactional-based fees instead of hiring someone part time to perform this function.

Six months into her venture, her business was booming. She received a peculiar call from someone in the fraud department of her ISO asking very odd questions about her setup. She learned that her business was identified as the possible source of a cardholder data breach. After explaining that she outsourced all of her maintenance and upkeep to PayProcess Express, she was informed the business unit was sold to another company, and that the merchant ID was still issued to her directly, therefore she was responsible for paying for fines and the forensic investigation.

Payton was crushed. How did she end up in this situation? She was now facing significant fines that could affect her ability to meet her creditors and payroll.

Payton made one critical error. While her knowledge of PCI was critical to how she set up her payment processing environment, she mistakenly thought that having a third-party manage her systems would cover her in the case a breach occurred. In reality, Payton was responsible for keeping up with her compliance, and she failed to ensure that Requirement 12.8 was met with respect to her outsourcer.

Had Payton processed under PayProcess Express's merchant ID, she may only be facing lost business due to consumer confidence versus facing fines and fees associated with the breach.

SUMMARY

Here are all the myths again:

- PCI just doesn't apply to us, because...we are special.
- PCI is confusing and ambiguous.
- PCI is too hard.
- Recent breaches prove PCI irrelevant.
- PCI is easy: we just have to "say Yes" on SAQ and "get scanned."
- My network, application, tool is PCI compliant.
- PCI is all we need to do for security!
- Even if breached and then found noncompliant, our business will not suffer.

Now that you know what the myths are and what the reality is, you are one step closer to painless, effective PCI DSS program as well as to secure and compliant organization that cares about its customers by protecting their data.

Remember that PCI is basic security; stop complaining about it – start doing it! By focusing on immediately useful parts of PCI DSS, you can start toward a full-scale risk management program, backed up by implemented and maintained controls. Then, after validating that you are compliant, don't stop: continuous compliance and security is your goal, not "passing an assessment." For example, remember that

compensating controls are usually temporary controls, not excuses to never do the right thing.

Overall, it is much more useful to develop “security and risk” mindset, not “compliance and assessment” mindset. Just as there is no true guaranteed “job security” today, there is no “guaranteed information security”; there is only one thing: doing the best you can do and being above average, so that attackers leave for greener pastures.

REFERENCES

- [1] “PCI-DSS: Not on health care provider’s radar” in “SC Magazine” online. <www.scmagazineus.com/PCI-DSS-Not-on-health-care-providers-radar/article/138783/>; 2011 [accessed 12.07.11].
- [2] PCI DSS SAQ. <www.pcisecuritystandards.org/saq/index.shtml>; 2011 [accessed 12.07.11].
- [3] ISO/IEC 27001:2005.
- [4] NIST 800-53. Recommended Security Controls for Federal Information Systems and Organizations. <<http://csrc.nist.gov/publications/drafts/800-53/800-53-rev3-FPD-clean.pdf>>; 2011 [accessed 12.07.11].
- [5] PCI Council. Web site <www.pcisecuritystandards.org>; 2011 [accessed 12.07.11].
- [6] The Prioritized Approach to Pursue PCI DSS Compliance. <www.pcisecuritystandards.org/education/docs/Prioritized_Approach_PCI_DSS_1_2.pdf>; 2011 [accessed 12.07.11].

Index

A

- Acceptable use policy, 131
- Access control lists (ACLs), 93
- Access controls. *See also* Network security
 - authentication, requirements
 - basics of, 84–85
 - identification and, 85
 - case study, 109–111
 - of loose permissions, 110–111
 - of stolen database, 109–110
 - common mistakes and pitfalls, 108–109
 - legacy systems, 108–109
 - physical access monitoring, 109
 - poor documentation, 108
 - configuring in windows, 94
 - educating users, 91–93
 - measures implementation, 316
 - multifactor authentication, 86
 - other ways of, 105–106
 - passwords
 - complexity requirements, 88, 98–99
 - SUID and SGID, 98
 - design, requirements, 88–89
 - encrypt CISCO, 99
 - physical security, 100–105
 - anti-skimming requirements, 104–105
 - handling visitors requirements, 101–102
 - media and physical data entry points
 - requirements, 103–104
 - Posix (Unix/Linux systems) access control, 97–99
 - principles of, 82–83
 - availability, 83
 - confidentiality, 82
 - integrity, 82
 - rendering passwords unreadable in transit and storage, 87
 - requirements
 - databases and, 90–93
 - Linux enforce password complexity, 98
 - locking users out, 86
 - PCI DSS in domain, 81–105
 - two-factor authentication, 89–90
 - user's access, 83–84
 - setting up SSH in cisco environment, 100
 - tools and best practices, 107–108
 - random password for users, 108
 - windows and PCI compliance, 93–97
 - enabling password-protected screen savers, 96
- inactive accounts in active directory, 95
- password requirements enforcement, 95
- setting file permissions, 97
- windows file access control, 93–94
- Access controls systems, 81, 84, 98
- Access lists (ACLs), 145, 301
- Access_log*, 202
- Access point (AP), 142
- Account lockout policy, 96
- ACLs. *See* Access control lists (ACLs)
- Acquiring bank, 4
 - in ASV picking, 180
 - in PCI payment ecosystem, 4
- Active directory. *See* Windows active directory
- Advanced Encryption Standard (AES), 148
 - cryptographic algorithm, 125
- AES. *See* Advanced Encryption Standard (AES)
- Air-gapping networks, 317
- Albrecht discount, 10
- Amazon cloud, 236
- American Express, 2, 16, 17, 21, 22, 26
- Android-based devices, 168
- Antimalware agent, 273
- Anti-spoofing technology, 64
- Antivirus software, 168, 169, 212, 273
 - for malicious software, 169
 - PCI DSS requirements, 210
 - in PCI requirements, 162
 - and Symantec antivirus, 212
- AoC. *See* Attestation of compliance (AoC)
- AP. *See* Access point (AP)
- Apache web server, 212
- Applicability, of PCI DSS, 19–20
- Application IDs, 90
- Application-layer vulnerabilities, 192
- Application vulnerability assessment, 161
- Approved scanning vendors (ASVs), 23, 185, 188, 277, 284, 344. *See also* Qualified security assessor (QSA); Vulnerability management
 - expectations from, 187–188
 - network scanning, 175
- Payment Card Industry Data Security Standard (PCI DSS), 32
- picking considerations, 180
- scanning tool, 287
- services, pricing models for, 181
- vulnerability management, requirements
 - walk-through from

- Approved scanning vendors (ASVs) (*cont.*)
 expectations, 187–188
 operationalizing, 186–187
 working of, 185–186
- Assessment process, 284
- Assessors, 283–293, 313
 advantages, 284–286
 balancing remediation needs, 285–286
 common vulnerability scoring system (CVSS)
 use, 289
 dealing with mistakes, 286–288
 reassessment planning, 292
 remediation planning, 288–292
 remediation process, 288, 290
 roots in security, 286
 use of failed assessment, 286
- ASVs. *See* Approved scanning vendors (ASVs)
- ATM machine, 141
 on-site, 256
- Attestation of compliance (AoC), 309
- Audio/visual (A/V) equipment, 158
- Auditors. *See* Assessors
- Authentication, access controls, 84–85
 basics of, 84–93
 educating users, 91–93
 rendering passwords unreadable in transit
 and storage, 87
 multifactor authentication, 86
 requirements, basics of
 databases and, 90–91
 identification, authentication and, 85
 locking users out, 86
 password design for PCI DSS, 88–89
 two-factor authentication, 89–90
 two-factor authentication, 89–90
- Automated fuel dispensers, 323
- Automated processing, 9
- Availability, 83, 291
 in PCI, 83
- AxCrypt, 122
- B**
- Back-office network, 157
- Bank identification number (BIN), 120
- Banned data, 119
- Base scoring metrics, 291
- BIN. *See* Bank identification number (BIN)
- BitLocker drive encryption, 123
- Bluetooth, 152
- Botnets, 10
- Brand security programs, 27
- Bring your own device (BYOD), 247
- Business
 continuity and risk assessments, 39
 outsource, 255
- Business as usual (BAU) activities, 242
- BYOD. *See* Bring your own device (BYOD)
- C**
- CAP. *See* Compliance acceleration program (CAP)
- Capital expenditure (CapEx), 304
- Card acceptance methods
 SAQ validation types based on, 25
- Card brands, 17, 22, 24, 26, 27, 36, 265, 298, 325
- Cardholder data, 45, 61, 116, 132
 appendix A of PCI DSS, 133–134
 availability, 115
 case study, 138–140
 of leaky data, 138–139
 of satellite location, 139–140
- CIA triad, 114–115
 PCI DSS data protection mapped to, 115
- common mistakes and pitfalls, 136–138
- compliant and security, 134–135
 determine what to do about your data, 135
 determine who needs access, 135
 develop and document policies, 135
 focus on shrinking scope, 135
 identify business processes with card data, 134
 identify where the data is stored, 135
- confidentiality, 114
 and integrity of, 62
- data protection, need of, 113–114
- done for security, 127
- integrity, 114
- PCI and key management, 126
- protection of, 113–140, 313
 stored data, 116–117
- requirements walk through from, 117–126,
 130–133
 addressed in, 116
 encryption methods for data at rest, 121–125
 in PCI, 127–130
 IPSEC virtual private networks, 128–129
 miscellaneous card transmission rules, 130
 secure sockets layer, 127
 transport layer security, 127
 wireless transmission, 129
- Cardholder database, 64
- Cardholder data environment (CDE), 163
- Cardholder data packs, loss of, 352
- Cardholder information security program (CISP),
 26, 116
- security audit procedures, 26

- Card scheme. *See* Card brand
- CardSystems, 29
- Card transmission rules. *See* Miscellaneous card transmission rules
- Cashless payments, 51
- CCO. *See* Chief compliance officer (CCO)
- CDE. *See* Cardholder data environment (CDE)
- Cellular data networks, 141, 142
- Central Office (CO), 139
- Certificate Authority (CA), 128
- CGI. *See* Common gateway interface (CGI)
- Change-detection mechanism, 230
- Checkpoint, 74
 - firewall, 200
- Chief compliance officer (CCO), 1
- Chief information officer (CIO), 51, 191
- Chip & PIN technology, 329
- ChoicePoint breach, 10
- CIO. *See* Chief information officer (CIO)
- Cisco devices, 99
 - ASA appliance record, 211
 - and PCI requirements, 99–100
 - enforce session timeout, 99
 - PIX firewall logs, 232
 - routers, 100
- CISPs. *See* Cardholder information security program (CISP)
- Citrix, 52, 85, 109
- Cleanup rule, 62
- Client application vulnerability, half-life of, 193
- Cloud computing
 - basics of, 235–240
 - cloud badness, 237
 - cloud challenges, 238–240
 - cloud changes everything, 237–238
 - what is cloud, 236–237
 - for better security and compliance, 242–243
 - cloud resources use in, 241–242
 - cost savings, advantage of, 243
 - definition of, 236
 - infrastructure as a service (IaaS), 236
 - PCI cloud examples, 240–241
 - and PCI DSS, 235–245
 - in depth, 244
 - enter the matrix, 244
 - in maintaining and assessing, 243–243
 - platform as a service (PaaS), 236
 - role in, 238
 - software as a service (SaaS), 236
- Cloud security alliance (CSA), 237
- CoBIT, 131
- Column-level encryption, 123–125. *See also* File/folder-level encryption; Full-disk encryption
 - advantages, 124
 - disadvantages, 124–125
- Command-based machines, 85
- Commercial off-the-shelf software, 298
- Common gateway interface (CGI), 133
- Common vulnerability and exposures (CVEs), 289
- Common vulnerability scoring system (CVSS), 165, 289
 - scoring, 188, 291
- Compensating control, 295–307
 - case studies, 305–307
 - of concierge travel agency, 306–307
 - of newborn concierge, 305–306
 - creation, 301–305
 - flat store network, 302
 - funny controls, 299–300
 - if not, 297–299
 - lifespan, 298
 - in PCI DSS, 297
 - store network segmentation, 303
- Compliance, 242
 - benefits of, 38
 - efforts, 1
 - type of, 20
- Compliance acceleration program (CAP), 266.
 - See also* Visa compliance acceleration program (CAP) fines
- Compliance achievement, PCI DSS, 261–282
 - bringing key players, 267–269
 - compliance team formation, 268
 - corporate sponsorship, 267–268
 - fast getting results, 269
 - front line notes, 269
 - roles and responsibilities of team, 268
 - budgeting time and resources, 270–272
 - goals and milestones establishment, 270–271
 - management's expectations, 270
 - setting expectations, 270
 - status meetings, 271–272
 - justifying business case, 262–266
 - compliance overlap, 263
 - leveraging, 263
 - cost for noncompliance, 265–266
 - need identification, 262
 - penalties for noncompliance, 265–266
 - validation level, 264–265
 - prioritized approach, 278–279
 - project quickstart guide. *See* Project quickstart guide

- Compliance achievement, PCI DSS (*cont.*)
 staff education, 272–274
 company on compliance training, 273
 compliance team training, 272–273
 corporate compliance training program setting, 273–274
 VISA technology innovation program (TIP), 279–280
- Compliance plan
 in corporate compliance training program, 273
 creation, 278
 justifying business case, 262
 and key players, 267
- Compliance planning project, 281
- Compliance team
 formation, 268
 training, 272–273
- Compliance training program, 274
- Compliant service provider list, 77
- Confidentiality, 82, 291
- Confidentiality, integrity, and availability (CIA) triad, 114–115, 207
 PCI DSS data protection mapped to, 115
- Configuration standards, 79
- Confusing validation requirements case, 39–40
- Continuing professional education (CPE), 319
- Core competency concept, 51
- Core operating system vulnerability, half-life of, 193
- Corporate compliance training program, 273
- Corporate leaders, 235
- Corporate networks, 151
- Corporate policy, 145
- Corporate social responsibility (CSR), 353
- Corporate sponsorship, obtaining of, 267–268
- Cost, for noncompliance, 265–266
- Couch-hacking, 9
- Country code (CC), 119
- CPE. *See* Continuing professional education (CPE)
- CPU cycles, 77
- Credit card
 acceptance, risks of, 254–255
 brands, 5
 fraud, 9. *See also* Identity theft; Personal data theft; Payment Card Industry Data Security Standard (PCI DSS)
 cybercriminals, 9
 security breaches, 10
 information, 5
 related crimes, 37
- Criminals, 9
 cyber. *See* Cybercriminals
- Critical Infrastructure Protection Standards, 2
- Critical key management, 126
- Crossborder prosecution, 9
- Cross-site request forgery (CSRF), 174
- Cross-site scripting (XSS), 174, 311
- Cryptanalysis, 326
- Cryptography, 136, 138
- CSA. *See* Cloud security alliance (CSA)
- CSRF. *See* Cross-site request forgery (CSRF)
- Current generation protection mechanisms, 237
- CVEs. *See* Common vulnerability and exposures (CVEs)
- CVSS. *See* Common vulnerability scoring system (CVSS)
- Cybercriminals, 9, 10
- Cyber Monday, 238
- Cybersecurity laws, 9

D

- DACs. *See* Discretionary access control lists (DACs)
- DAP. *See* Database audit and protection (DAP) tools
- DAST. *See* Dynamic application security testing (DAST)
- Data acquisition and vulnerability management, 165
- Database activity monitoring, 202
- Database Administrator (DBA), 90
- Database audit and protection (DAP) tools, 199
- Database encryption. *See* Column-level encryption
- Databases, 90, 135
 structure, 74
- Data breaches, 67
 in business processes, 135
 at TJX, 37
- Data encryption, 137
 column-level encryption. *See* Column-level encryption
 for data at rest, 121–125
 disk-only, 300
 file/folder-level encryption. *See* File/folder-level encryption
 full-disk encryption. *See* Full-disk encryption
 key management, 72
 mistakes, 136
 requirements, 298
 whole disk for laptop, 300
- Data loss prevention (DLP) technology, 45, 306
- Data protection methods, 113
- Data Security Standards, 27
- Day care management software, 109
- DBA. *See* Database Administrator (DBA)

Default password, 33, 68, 75
 Defense-in-depth concepts, 59
 De-militarized zone (DMZ), 62, 195
 functionality, 77
 Denial of service (DoS)
 protection, 59
 vulnerabilities, 187
 Desktop applications, 179
 Developing security program case, 38–39
 Digital subscriber line (DSL), 151
Discover, 22, 26
 Discretionary access control lists (DACLs), 93
 Documents library, 248
 Domain name system (DNS), 70, 203
 blackhole, 258
 Drive-by downloads, 10
 DSL. *See* Digital subscriber line (DSL)
 Dynamic application security testing (DAST), 174

E

ECC. *See* Elliptical curve cryptography (ECC)
 E-commerce, 138
 implementation, 233
 transactions, 335
 web sites, 136
 E-discovery process, 231
 EFS. *See* Encrypted file system (EFS)
 eGRC tools, 5
 Egress filtering, 75
 Electronic card payment ecosystem, 17–20
 PCI DSS
 applicability of, 19–20
 goal of, 18–19
 Elliptical curve cryptography (ECC), 148
 E-mail, 76
 logs, 202
 plain card numbers, 130
 scams, 9
 EMV technology. *See* Europay, MasterCard, and Visa (EMV)
 Encrypted file system (EFS), 122
 End sentinel (ES), 119
Error_log, 202
 Ethernet port, 158
 Europay, MasterCard, and Visa (EMV), 262, 280,
 323–324, 329
 friendly language, 118
 terminals, 253
 use and deployment of, 279
 Europe *vs.* US, payment schemes, 329–330
 Evil Twin. *See also* Wi-Fi® cards
 Expiration date (ED), 82, 119
 External vulnerability scans, 32

F

Facility entry controls, requirements, 76
 FDE. *See* Full-disk encryption (FDE)
 Fear, uncertainty, and doubt (FUD), 262
 Federal Energy Regulatory Commission, 2
 Federal Information Security Management Act, 5
 FedEx package, 103
 Field separation (FS), 119
 File/folder-level encryption, 121–122. *See also*
 Column-level encryption; Full-disk
 encryption (FDE)
 advantages, 121
 disadvantages, 121–122
 vs. full-disk encryption, 124
 File-integrity monitoring (FIM) systems, 67
 File permissions
 chmod command, 97
 Linux system, 97
 on standalone windows computers, 97
 FIM. *See* File-integrity monitoring (FIM) systems
 Firewall, 74
 policies, 75
 rules, 75
 Firewall Configuration Standards, establishment
 of, 61–73
 default passwords, 68
 delete unnecessary accounts, 69
 denying traffic from untrusted networks and
 hosts, 62–63
 develop configuration standards, 69
 encrypt nonconsole administrative access, 72
 hosting providers must protect shared hosted
 environment, 73
 implement single purpose servers, 70
 keep the inventory!, 73
 personal firewalls, 65
 requirements
 considerations for, 65
 defaults and other security parameters,
 67–73
 oddball, 65–67
 restricting connections, 63–64
 Simple Network Management Protocol
 (SNMP), 68
 system security parameters configuration, 71–72
 First-time passwords, 89, 108
 Flat networks, 77
 Flat store network, 302
 VLAN with ACLs segmenting, 303
 Flexibility, type of, 238
 Format code (FC), 119
 Fraud, data theft, and regulatory mandates, 9–14
 Free open-source database MySQL, 125

- Front-end servers, 64
- FUD. *See* Fear, uncertainty, and doubt (FUD)
- Full-disk encryption (FDE), 122–123. *See also* Column-level encryption; File/folder-level encryption
- advantages, 122
 - disadvantages, 122–123
 - vs.* file-based encryption, 124
- Fully patched system, 163
- G**
- Gantt chart, 271, 272
 - Gantter, planning software, 271
 - GanttProject, planning software, 271
 - Gap analysis, 278, 288
 - Gap assessment, 292
 - General mobile radio service (GMRS), 152
 - General packet radio service (GPRS) network, 127
 - Global Point of Sale Counterfeit Liability Shift, 337
 - Global System for mobile communications (GSM), 127
 - GMRS. *See* General mobile radio service (GMRS)
 - GNU Grep, 45
 - GNU privacy guard, 122
 - Google checkout, 327–328
 - disadvantages, 327
 - online retailers, 327
 - GPO. *See* Group policy objects (GPOs)
 - GPRS. *See* General packet radio service (GPRS) network
 - GPS data, 153
 - Gramm–Leach–Bliley Act of 1999, 2, 5
 - Group policy objects (GPOs), 93
 - GSM. *See* Global system for mobile communications (GSM)
 - Guest operating systems, 70
- H**
- Hacking, 9
 - Hactivism, 310
 - “Hard-coding” secrets, 137
 - HD-DVD encryption breaches, 137
 - Health Insurance Portability and Accountability Act (HIPAA) of 1996, 1, 5, 263
 - compliance, 306
 - for PCI DSS compliance, 263
 - Heartland payment systems, 5
 - HIDS. *See* Host-based intrusion detection system (HIDS)
 - High availability networks, 66
 - High-power cards, 153
 - High-tech crime, 9
 - HIPAA. *See* Health Insurance Portability and Accountability Act (HIPAA) of 1996
 - HIPS. *See* Host-based intrusion prevention system (HIPS)
 - Host-based intrusion detection system (HIDS), 66
 - Host-based intrusion prevention system (HIPS), 66
 - Host-based security, 59
 - HTTP. *See* Hypertext Transfer Protocol (HTTP)
 - Hypertext Transfer Protocol (HTTP), 161
- I**
- IBM DB2 database, 125
 - IBMs, 180
 - iCloud, 249
 - ICMP. *See* Internet Control Message Protocol (ICMP)
 - Identity theft, 9
 - IDS. *See* Intrusion detection systems (IDSs)
 - Id theft. *See* Identity theft
 - IEC. *See* International Electrotechnical Commission (IEC)
 - Inbound traffic, 258
 - Incident response policy, 319
 - Independent Sales Organizations (ISOs), 259
 - ISO17799 standard, 39
 - ISO27002 standard, 12
 - program, 320
 - security framework, 39
 - Industry-accepted system, 69
 - Information security
 - policy maintainance, 318–319
 - program, 38
 - regulation, 6
 - risk management, 338
 - Information technologists, 5
 - Information technology (IT)
 - concepts, 44
 - department, 261
 - governance frameworks, 173
 - COBIT, 173
 - COBIT/ISO27001, 205
 - ITIL, 173
 - platform, logging mechanisms in, 205
 - staff training on security, 311
 - secure coding practices, 311
 - systems training, 312
 - systems, 305
 - Infrastructure, 208
 - of log management, 223
 - of network, 201, 208
 - and security, 199

In-scope systems, 87, 190, 194, 221, 318
 Integrity, 82
 Intelligence, 66
 Internal network addresses, 64
 Internal security assessor (ISA), 22, 295
 training, 277
 Internal vulnerability scanning, 189–191. *See also*
 Approved scanning vendors (ASVs);
 Vulnerability management
 data, 72
 PCI compliance, 189
 PCI DSS scan issue tracking process, 190
 penetration testing, 190–191
 remediation, 189
 in servers, 189
 system change issue, 190
 International Electrotechnical Commission (IEC), 2
 International Organization for Standardization
 (IOS), 2
 Internet
 facing systems, 195
 IP addresses, 347
 ready appliances, 79
 Internet Control Message Protocol (ICMP), 59
 Internet Explorer, 192
 Internet Protocol (IP)
 addresses, 188, 277
 based point of sale (POS) terminals, 76
 based restriction, 90
 masquerading, 63
 Internet Protocol security (IPSec), 127
 Internet service provider (ISP), 59
 Intrusion detection systems (IDSs), 65, 146, 198,
 226, 228
 signature-based, 229
 Intrusion prevention systems (IPSs), 59, 67, 167,
 198, 228
 solution, 149
 IOS. *See* International Organization for
 Standardization (IOS)
 IP. *See* Internet Protocol (IP)
 iPhone app, 305
 IPSs. *See* Intrusion prevention systems (IPSs)
 Iron Mountain, 49, 103
 ISA. *See* Internal security assessor (ISA)
 ISO. *See* Independent Sales Organizations (ISOs)
 ISP. *See* Internet service provider (ISP)
 IT. *See* Information technology (IT)

J

JCB card, 23
 Juniper firewall log message, 201

K

Key management
 in data encryption, 72
 and PCI, 126
 recommendations for, 126
 KISP. *See* Cardholder information security program
 (CISP)
 Knee-jerk reactions, 253
 Knowledge of encryption, 300
 “known bad” messages, 217

L

LDAP. *See* Lightweight Directory Access Protocol
 (LDAP) server
 Least privilege, principle of, 83
 Legacy systems, 77
 Legitimate constraint, 295
 Leveraging technologies, 151
 Liability shift, 337
 Lightweight Directory Access Protocol (LDAP)
 server, 87
 Linux, 97
 change file permissions in, 97
 password complexity requirements, 98
 server, 230
 Log analysis, large-scale server operating system,
 202
 Logging and monitoring cardholder data
 environment, 197–233
 across PCI DSS requirements, 206
 case study, 232–233
 of risky risk-based approach, 232
 of tweaking to comply, 233
 common mistakes and pitfalls, 231
 in depth, 199–202
 finding exceptions, 215
 high level investigative process, 218
 integrity monitoring, 230–231
 intrusion detection and prevention, 226–229
 monitoring data and log for security issues,
 207–209
 monitoring tools, 225
 Logging and monitoring in PCI DSS, 198–199
 daily tasks, 221
 daily workflows, 218
 event details, 206
 log flow, 214
 logging policies and procedures, 213–221
 building an initial baseline manually,
 216–217
 compliance evidence package, 220
 daily tasks, 220–221

- Logging and monitoring in PCI DSS (*cont.*)
 exception investigation and analysis, 217–219
 guidance for identifying “known bad”
 messages, 217
 periodic operational task summary, 220
 validation of log review, 219–220
 relevance of logs, 203
 requirements covered, 198
 tools for, 221–225
- Logging in PCI requirements, 203–207, 209–213
 logging requirements and address, 205
 logging tools useful for PCI DSS, 223
 log-producing technologies, 201
- Logging policy, 220
- Log investigative checklist, 219
- LogLogic, 223
- Log management
 problem, 200
 project, 233
 tool, 216
- Log review, 204, 214
 validation of, 219–220
- Longitudinal redundancy check (LRC), 119
- LRC. *See* Longitudinal redundancy check (LRC)
- M**
- MAC. *See* Mandatory access control (MAC); *See also* Media access control (MAC)
- Malicious software, 10, 168, 249
- Management sponsorship, 267
- Mandatory access control (MAC), 304
 appropriate systems to, 304
- MasterCard, 22, 25, 26, 265, 335
 fines, 266
 level 1 merchants, 266
 level 2 merchants, 266
 level 3 merchants, 266
 new validation requirements, 39
 presentation, 16
 rules for vulnerability scanning, 26
- MDM. *See* Mobile device manager (MDM)
- Media access control (MAC), 143
- Merchants, 16, 125, 149, 164, 280
 account and ID, 46
 levels of, 19, 20
 original compliance dates for, 21
 PCI, 276
 official definition of, 16
 taking advantage of TIP, 280
- Merchant service provider (MSP), 17
- Message Digest 5 (MD5), 99
- Microsoft Excel spreadsheet, 279
- Microsoft IIS, 128
- Microsoft MS SQL server, 125
- Microsoft operating systems, 122, 123
- Microsoft project, 271
- Microsoft windows server, 157
- Miscellaneous card transmission rules, 130
- Mitigation, vulnerability management and, 167–168
- Mixed mode, 48
- Mobile device manager (MDM), 249
- Mobile scheme, 247–251
 available guidance, 248
 case study, 250
 of summer festival, 250
 deploying technology safely, 249–250
 mobility addressed in PCI DSS 3.0, 247
 PA-DSS 3.0 fit, 248
- Monitor and test networks, 316–318
- Monitoring
 database activity, 202
 file integrity, 67
 and logging, 197–233
 in PCI DSS, 198–199
 physical access, 109
- MSP. *See* Merchant service provider (MSP)
- Multifactor authentication. *See* Two-factor authentication, requirements
- MySQL database, 125
- N**
- National Security Agency (NSA), 98
- National vulnerability database (NVD), 166, 167, 289
- Near-field communication (NFC)
 scheme, 324
 technology, 325–326
- “Need-to-know” for access, 83, 92
- NERC. *See* North American Electric Reliability Corporation (NERC)
- NetStumbler, 153
- Network-based intrusion prevention system (NIPS), 65, 66, 227
- Network connection, 143
- Network intrusion detection system (NIDS), 65, 202, 226
- Network security. *See also* Access controls; Payment Card Industry Data Security Standard (PCI DSS)
 building and maintaining, 59
 case study, 76–79
 of do over, 78–79
 of large, flat corporate network, 77–78
 of small, flat store network, 76–77

- firewall configuration standards establishment, 61–73
 considerations for requirements, 65
 default passwords, 68
 defaults and other security parameters requirements, 67–73
 delete unnecessary accounts, 69
 denying traffic from untrusted networks and hosts, 62–63
 develop configuration standards, 69
 encrypt nonconsole administrative access, 72
 hosting providers must protect shared hosted environment, 73
 implement single purpose servers, 70
 keep the inventory!, 73
 oddball requirements, 65–67
 personal firewalls, 65
 restricting connections, 63–64
 Simple Network Management Protocol (SNMP), 68
 system security parameters configuration, 71–72
 layers of, 60
 mistakes and pitfalls, 75
 documentation, 75
 egress filtering, 75
 system defaults, 75
 PCI DSS requirements in domain, 60
 tools and best practices, 74
 Network segmentation, 297
 Network test access port (TAP), 66
 Network Time Protocol (NTP), 203
 servers, 207
 Network vulnerability
 assessment. *See* Vulnerability assessment
 scanners, 164
 scanning. *See* Vulnerability assessment
 testing. *See* Vulnerability assessment
 New technology file system (NTFS), 94
 Next-gen payments. *See* Europay, MasterCard, and Visa (EMV); *See also* Payment schemes, new
 NFC. *See* Near-field communication (NFC)
 NIDS. *See* Network intrusion detection system (NIDS)
 Nigerian e-mail scams, 9
 NIPS. *See* Network intrusion prevention system (NIPS)
 NIST 800-53 document, 339
 Noncompliance
 cost for, 265–266
 penalties for, 265–266
 North American Electric Reliability Corporation (NERC), 2
 NSA. *See* National Security Agency (NSA)
 NTFS. *See* New technology file system (NTFS)
 NTP. *See* Network Time Protocol (NTP)
 NVD. *See* National vulnerability database (NVD)
- O**
- OmniPlan, planning software, 271
 On-site assessors. *See* Approved scanning vendors (ASVs)
 OpenProj, planning software, 271
 Open-source Apache web server, 202
 Open-source equivalents
 GanttProject, 271
 OmniPlan, 271
 OpenProj, 271
 OpenWorkbench, 271
 Open-source tool, 285
 Open source vulnerability database (OSVDB), 289
 OpenWorkbench, planning software, 271
 Oracle database, 125
 OSVDB. *See* Open source vulnerability database (OSVDB)
 Outbound traffic, 63, 75, 258
 OWASP, 174
- P**
- PABP. *See* Payment application best practices program (PABP)
 Packet-filtering router, 59
 PA-DSS. *See* Payment Application Data Security Standard (PA-DSS)
 PAMs. *See* Pluggable authentication modules (PAMs)
 PAN. *See* Primary account numbers (PANs)
 Passwords, 90
 default. *See* Default passwords
 design for PCI DSS, requirements, 88–89
 and Linux distribution, 98
 policy for, 95, 267
 enforcement of, 107
 random passwords. *See* Random password for users
 Patch-management program, 170
 Payment application best practices program (PABP), 31
 Payment Application Data Security Standard (PA-DSS)
 compliance, 21
 program, 31

- Payment brands. *See* Card brands
- Payment card breach, 254
- Payment card industry (PCI), 43, 60, 161
- applicability, 19
 - breach, 259
 - compliance, 17, 50
 - creators, 168
 - derived logging policy, 214
 - ecosystem, 28
 - environment, 43
 - logging
 - guidance, 223
 - requirements, implementation of, 232
 - payment ecosystem, 18
 - quick reference guide, 28
 - related security breaches, 5
 - requirements, 203–207
 - in cardholder data, 127–130
 - logging requirements and address, 205
 - logging tools useful for PCI DSS, 223
 - log-producing technologies, 201
 - self-assessment, 319
 - standards, 35
 - 3.0 standard, predecessor versions of, 22
- Payment Card Industry (PCI) Council, 27–29, 48, 70, 75, 129, 239
- administers, 29
 - glossary, 16
 - web site, 23, 24
- Payment Card Industry Data Security Standard (PCI DSS), 1–5, 7, 15, 43, 59, 81, 113, 141, 253
- in access controls, requirements in domain, 81–105
 - applicability of, 5
 - ASVs, 32
 - case study, 38–40
 - of confusing validation requirements, 39–40
 - of developing security program, 38–39
 - changes to, 35
 - common myths of, 19
 - compliance and, 1, 15, 38, 47, 55, 309
 - deadlines, 21–23
 - validation, 23–26
 - in depth, 21
 - history of, 26–27
 - implications of, 3
 - key monitoring technology for, 229
 - knowledge and experience in, 54
 - last version of, 65
 - myths/misconceptions
 - breaches, irrelevant, 341
 - cardless merchant, 353
 - case study, 353
 - confusing/ambiguous, 337
 - data security, requirements, 343
 - internet-facing IP addresses, 347
 - network compliant, 348, 349
 - on-site QSA assessment, 346
 - prioritized approach, 350, 351
 - requirements, 333
 - SAQ Type A, 336, 337
 - security professional, 343
 - self-assessment questionnaire (SAQ), 336
 - toothless, 351
 - unreasonable and too much security, 339
 - use of, 348
 - vulnerabilities, 347
 - work example, 334
- organization of, 4
- PA-QSAs, 31
- password design for, 86
- payment card industry forensic investigator (PFI), 31
- PCI DSS 3.0, 104, 247
- PCI professional (PCIP), 32
- perimeter, 52
- periodic log review, principle of, 215
- prioritized approach for, 54, 278
- QIR program, 32
- qualified security assessors (QSAs), 29–30, 31
- requirements, 4, 6, 116
- logging, keys of, 220
 - quick overview of, 32
- risk and, 36–37
- scope of, 34, 225
- standard, 21
- use in daily job, 4
- validation requirements, 24
- version 2.0, 32
- violation of, 37
- Payment Card Industry Data Security Standard (PCI DSS) compliance, 1, 15, 38, 47, 55, 309
- data protection, 136
- deadlines, 21–23
- FAQ 1233, 50
- guidance, 168
- prioritized approach, 118
- security assessment procedures, 297
- steps to, 276
- supplemental guidance, on virtualization, 239
- validation, 23–26
- Payment Card Industry Data Security Standard (PCI DSS), logging and monitoring in, 198–199

- daily tasks, 221
- daily workflows, 218
- event details, 206
- log flow, 214
- logging policies and procedures, 213–221
 - building an initial baseline manually, 216–217
 - compliance evidence package, 220
 - daily tasks, 220–221
 - exception investigation and analysis, 217–219
 - guidance for identifying “known bad” messages, 217
 - periodic operational task summary, 220
 - validation of log review, 219–220
- relevance of logs, 203
- requirements covered, 198
- tools for, 221–225
- Payment Card Industry Data Security Standard (PCI DSS), periodic maintenance requirements for, 312–319
 - building and maintaining, 313
 - implementing strong access control measures, 316
 - maintaining information security policy, 318–319
 - maintaining vulnerability management program, 314–316
 - monitor and test networks, 316–318
 - protect cardholder data, 313
- Payment card industry forensic investigator (PFI) program, 27, 31
- Payment card industry qualified security assessor (PCI-QSA), 2
- Payment card industry (PCI), scope of basics of, 43–47
 - case study, 55–57
 - of entrenched enterprise, 56–57
 - of leaky data, 55–56
 - definition of, 44
 - determining and reducing, 43
 - “gotchas” of, 47–50
 - guidance, 44
 - planning your project, 53–55
 - scope reduction tips, 50–53
- Payment Card Industry Security Standards Council (PCI SSC), 1, 2, 15, 298, 319
 - prioritized approach, 281
- Payment card industry vulnerability management, case study in
- PCI at an e-commerce site, 187
- PCI at retail chain, 194–195
- Payment card processing networks, 200, 258
- Payment card processor, 2
- Payment card transactions, fraud risk of, 18
- Payment schemes, new, 323–328
 - case study, 330–331
 - cashless cover charge, 331
 - customer experience, 330–330
 - EMV, 323–324, 329
 - Europe *vs.* US *vs.* the rest of world, 329–330
 - google checkout, 327–328
 - mobile, 324–325
 - NFC. *See* Near-field communication (NFC) technology
 - paypal. *See* Paypal
 - predictions, 328–329
 - square scheme, 326–327
 - stripe. *See* Stripe
 - taxonomy and tidbits, 329–330
- Payment security conference, 16
- Paypal, 327–328
 - disadvantages, 327
 - online retailers, 327
- PayPass (MasterCard), 325
- Payton, 354
- PCI DSS. *See* Payment Card Industry Data Security Standard (PCI DSS)
- PCI-QSA. *See* Payment card industry qualified security assessor (PCI-QSA)
- PCMCIA slot, 142
- PDF document, 279
- Penetration testing, 87, 145, 154, 162, 317
 - elements in, 317
 - PCI at e-commerce site, 195
 - requirements, 190–191
- Personal data theft, 9
- Personal identification number transaction security (PTS), 27
- Phishing attacks, 10
- Pluggable authentication modules (PAMs), 98
- POI. *See* Point of interaction (POI)
- Point of interaction (POI), 329
- Point of sale (POS), 249
 - network, 101, 141, 301
 - software, 257
 - systems, 49, 221
 - technicians, 45
 - terminals, 105
- Point to point encryption (P2PE), 28
 - guidance, 43
- Policy, definition of, 165
- POS. *See* Point of sale (POS)
- POSIX
 - access control, 97–99
 - style systems, 98

PostgreSQL database, 90, 125
 P2PE. *See* Point to point encryption (P2PE)
 Prescriptive Technical Standards, 35
 Price-sensitive organizations, 164
 Primary account numbers (PANs), 43, 52, 82, 117, 119, 173
 Primary assessment group, 90
 Prioritization, 165–166
 Private network, 63
 Process cardholder data, 239
 Profit & loss (P&L) accountability, 56
 Project quickstart guide, 275–278
 annual assessment preparation, 278
 corporate sponsorship, 275
 gap analysis, 278
 PCI DSS compliance plan creation, 278
 PCI DSS SAQ completion, 277
 PCI level determination, 276
 setting up quarterly external network scans, 277
 steps of, 276
 team identification and establishment, 275
 validation by QSA, 277
 Protect devices, 104
 Public networks, types of, 129

Q

QR codes, 237
 QSA. *See* Qualified security assessors (QSAs)
 Qualified security assessors (QSAs), 22, 29, 30, 48, 49, 61, 86, 102, 203, 243, 265, 288, 295
 assessment, 162
 employee lookup tools, 30
 in PCI DSS, 29–30
 project quickstart guide, validation by, 277
 Quality assurance (QA) environment, 178
 Qualys vulnerability research, 175
 Quick scan-then-fix approach, 164

R

RACL. *See* Reflexive access lists (RACL)
 Radio frequency (RF) communication, 141
 Radio-frequency identification chips, 325
 RADIUS server, 157
 RAID cards, 299
 Random number generator (RNG), 108, 300
 Random password for users, 108
 RC4 algorithm, 143
 RDBMS. *See* Relational database management systems (RDBMS)
 RDP, 306

Recommended Security Controls for Federal Information Systems, 339
 Reflexive access lists (RACL), 145
 Relational database management systems (RDBMS), 202
 Remediation
 planning for, 288–292
 process, 288, 290
 Report on compliance (ROC), 23, 265, 309
 executive summary section of, 53
 reporting instructions, 2
 Request for comment (RFC), 63
 RFC 1918, 63, 64
 RFC. *See* Request for comment (RFC)
 RNG. *See* Random number generator (RNG)
 ROC. *See* Report on compliance (ROC)
 Routers, 74
 Ruby script, 108
 Rule of thumb, 288

S

Safe payment processing, 260
 SAQ. *See* Self-assessment questionnaire (SAQ)
 Sarbanes–Oxley (SOX) Act, 5, 33, 205
 compliance plans, 263
 review, 316
 Scanning vendor, 195
 SCAP. *See* Security Content Automation Protocol (SCAP)
 Scope reduction technique, 51
 Scoping errors, 194
 Secure server operating systems, 197
 Secure shell (SSH), 87, 127
 Secure socket layer (SSL), 87, 127, 202
 certificates, 73
 connection, 147
 Security, 309–310
 case study, 319–320
 of compliant company, 320
 guidance, self-respecting, 198
 motivated regulatory guidance, 34
 patches, 315
 PCI DSS requirements with periodic maintenance, 312–319
 building and maintaining, 313
 implementing strong access control measures, 316
 maintaining information security policy, 318–319
 maintaining vulnerability management program, 314–316

- monitor and test networks, 316–318
protect cardholder data, 313
- PCI self-assessment, 319
periodic review and training, 310–312
pillars of, 81
 availability, 83
 confidentiality, 82
 integrity, 82
professionals, organization, 198
software-defined, 240
standards, 134
templates, 94
- Security assessment procedures, 296
assessors. *See* Assessors
in PCI DSS, 297
reassessment planning, 292
remediation process, 292
- Security Content Automation Protocol (SCAP), 165
- Security focus Bugtraq, 289
- Security policy considerations. *See also* Cardholder data; protection of
 maintaining information, requirements for, 318–319
- Self-assessment questionnaire (SAQ), 23, 54, 60, 280, 281
- Self-assessment questionnaire D (SAQ D), 264, 319
- SE Linux, 98
- Servers, 189
- Service code (SC), 119
- Service providers, 17, 86
 levels, 21
- Service set identifier (SSID) broadcast, 143
- Set group ID (SGID), 98
- Set user ID (SUID), 98
- SIG. *See* Special interest group (SIG)
- SIM-based payments, 324, 325, 331
- Simple Network Management Protocol (SNMP), 68
 community, 147
- Slow-moving compliance, 236
- Small business, PCI for, 253–260
 basic outsourcing cost analysis, 259
 case study, 259
 of cashless cover charge, 259
 credit card acceptance, risks of, 254–255
 e-commerce, 257
 knee-jerk reactions, 253
 new business considerations, 255–257
 point-of-sale (POS) software, 257
 SMB hardening, scheme for, 258
 traffic analysis, 258
- Small companies. *See* Small business
- SNMP. *See* Simple Network Management Protocol (SNMP)
- Social security numbers, 87
- Socratic method, 44
- Software-as-a-service (SaaS), 161
- Software-development processes, 174
- Solid state disks, 104
- Sound information security program, 114
- SOX. *See* Sarbanes-Oxley (SOX) Act
- Special interest group (SIG), 43
- Spyware, 10
- SQL. *See* Structured query language (SQL)
- Square scheme, 327
- SSH. *See* Secure shell (SSH)
- SSID, 143. *See also* Service set identifier (SSID)
 broadcast
- SSL. *See* Secure socket layer (SSL)
- Stale authentication information, 85
- Standard vulnerability scanning technology, 185
- Start sentinel (SS), 119
- Stealth rule, 62
- Stripe, 327–328
- Structured query language (SQL), 166
- Switching network, 78
- Symantec antivirus, 212
- System password management, 123
- T**
- Tablet-based POS system, 250
- Taser®, 300
- Technology improvement program (TIP), 325
- Technology innovation program (TIP), 279
- Telnet command, 72, 87, 100
- Temporal score metrics, 292
- Time-synchronization technology, 207
- TLS. *See* Transport layer security (TLS)
- Tools, for logging and monitoring in PCI DSS, 221–225
- Tracey's systems, 56
- Transaction theory, 51
- Transport layer security (TLS), 127
- Tripwire*, 230
- TrueCrypt, 122
- Twitter, 171
- Two-factor authentication, requirements, 89–90
- U**
- UCF. *See* Unified compliance framework (UCF)
- UK lottery scams, 9
- Unauthorized wireless devices
 quarterly sweeps/wireless IDS/IPS, 150–151
 requirements, 148–151

- Unified compliance framework (UCF), 5
- Unified threat management (UTM), 226
devices, 195
- Unique ID, 84
- UNIX-based systems
Linux, 97
POSIX-style ACLs, 97
- Unix servers, 222
- Un-trusted networks, 258
- USB Flash media, 104
- U.S. Department of Veteran's Affairs, 5
- UTM. *See* Unified threat management (UTM)
- V**
- Validation
level in justifying business case, 264–265
of log review, logging policies and procedures, 219–220
mechanisms, 24
procedures, 214
requirements
MasterCard, 39
submission, acquiring bank in, 277
- SAQ, on card acceptance methods, 25
- Vendor
supplied security patches, 171
wireless network, 145
- Vigilant approach, 242
- Virtual assets, 239
- Virtualization, 48
- Virtual local area network (VLAN), 66, 301
- Virtual private network (VPN), 128, 146
client, 65
connection, 47
option for, 129
software, 78
- Visa, 22, 265
outcomes, 280
card, 335
- Visa compliance acceleration program (CAP) fines, 266
level 1 merchants, 266
level 2 merchants, 266
- VLAN. *See* Virtual local area network (VLAN)
- Voice over Internet Protocol (VoIP)
gateways, 199
technologies, 106
- VoIP. *See* Voice over Internet Protocol (VoIP)
gateways
- VPN. *See* Virtual private network (VPN)
- Vulnerability
databases, NVD, 289
laws of, 192
type of, 288
- Vulnerability assessment
definition of, 161
tools, 75, 164
- Vulnerability management, 161–196
activities in PCI DSS, 196
antivirus log setting, 170
case study, 194–195
PCI at an e-commerce site, 187
PCI at retail chain, 194–195
- internal vulnerability scanning, 189–191
penetration testing, 190–191
- national vulnerability database, 167
in PCI, 163–164
common mistakes, 192–194
- processing stages of, 165–168
data acquisition, 165
mitigation, 167–168
policy definition, 165
prioritization, 165–166
- program, 33
maintainance, 314–316
- requirements walk-through from, 168–170, 170–178, 179–188
- ASV scanning
expectations, 187–188
external vulnerability scanning, 180
operationalizing, 186–187
picking considerations, 180–185
working of, 185–186
- PCI DSS
controls in, 162, 182
covered in, 162–163
secure and compliant, 169–170, 178
web-application firewalls, 177–178
web-application scanning (WAS), 175–177
web-application security and web
vulnerabilities, 174–178
- user privilege violation vulnerability in NVD, 176
- W**
- WAFs. *See* Web-application firewalls (WAFs)
- WAS. *See* Web-application scanning (WAS)
- Web
based equivalent, 271
Gantt, 271
- based map service, 79
- based training class, 274
- portals, 177
- server logs, 202
- uniform resource locator, 207

- Web-application firewalls (WAFs), 59, 174, 316
 Web-application scanning (WAS), 161
 Web site, 77
 Whiz-bang technology, 151
 Whole-disk encryption methods, 122
 Wi-Fi®, 142, 249
 - antenna, 156
 - cards, 153, 156
 - key, 156
 - POS system, 150
 - protected access (WPA), 147
 - signals, 151
 - technology, 144, 150
 WiMAX. *See* Worldwide interoperability for microwave access (WiMAX)
 Windows access control lists (ACLs), 93
 Windows active directory, 49, 70, 95
 Windows and PCI compliance, 93–97
 - enabling password-protected screen savers, 96
 - inactive accounts in active directory, 95
 - password requirements enforcement, 95
 - setting file permissions, 97
 - windows file access control, 93–94
 Windows encrypted file system (EFS), 122
 Windows Explorer, 97
 Windows firewall, 65
 Windows systems, 95
 - Windows 7*, 95
 - Windows 8*, 95
 - Windows Server 2003/2008*, 95
 - Windows Server 2012*, 95
 - Windows Vista*, 95
 Wired equivalent privacy (WEP) key, 142, 147, 155
 Wireless analyzer, 149
 Wireless encryption technologies, 147
 Wireless IDS/IPS technology, 317
 Wireless networks, 78, 142
 - testing, 179
 Wireless network security, 142–143
 - case study, 155–159
 - detached POS, 158–159
 - of double secret wireless network, 158
 - of expansion plan, 157
 - of untethered laptop, 155–156
 - common mistakes and pitfalls, 154–155
 - WEP drawbacks, 155
 - need of, 151–152
 - in PCI DSS, 144–151
 - quarterly sweeps/wireless IDS/IPS, 150–151
 - requirements, 146–148
 - documentation, 145–146
 - logging and, 148
 - testing for unauthorized wireless, 148–151
 - tools and best practices, 153–154
 - use of, 141–159
 - wireless technologies. *See* Wireless technologies
 Wireless technologies, 141, 152
 Worldwide interoperability for microwave access (WiMAX), 152
 WPA2, 802.11 network, 147

Y

“Y,” clicking on computer, 330

Z

“Zero-day” attacks, 168
 Zigbee, 152