Verizon 2010 Payment Card Industry Compliance Report

AUTHORS AND CONTRIBUTORS:

Wade Baker
Michael Dahn
Tabitha Greiner
Alex Hutton
C. David Hylender
Peter Lindstrom
Jennifer Mack
Christopher Porter
Denson Todd
Other members of the PCI and RISK teams

TABLE OF CONTENTS

Verizon 2010 Payment Card Industry Compliance Report

A study conducted by the Verizon PCI and RISK Intelligence teams.

Quick Summary

This report analyzes findings from actual Payment Card Industry Data Security Standard (PCI DSS) assessments conducted by Verizon's team of Qualified Security Assessors (QSAs). The report examines the progress of organizations toward the goal of compliance and includes topics such as how and why some seem to struggle more than others. Also presented are statistics around which PCI DSS requirements and sub-requirements are most and least often in place (or compensated for) during the assessment process. Finally, the report overlays PCI assessment data with findings from Verizon's Investigative Response services to provide a unique risk-centric slant on the compliance process. Key findings include:

- 22% of organizations were validated compliant at the time of their Initial Report on Compliance (IROC). These tended to be year after year repeat clients.
- On average, organizations met 81% of all test procedures defined within PCI DSS at the IROC stage. Naturally, there was some variation around this number but not many (11% of clients) passed less than 50% of tests.
- Organizations struggled most with requirements 10 (track and monitor access), 11 (regularly test systems and processes), and 3 (protect stored cardholder data).
- Requirements 9 (restrict physical access), 7 (restrict access to need-to-know), and 5 (use and update anti-virus) showed the highest implementation levels.
- Sub-requirement 3.4 (render the Primary Account Number (PAN) unreadable) was met through compensating controls far more often than any other in the standard.
- Organizations do not appear to be prioritizing their compliance efforts based on the PCI DSS Prioritized Approach published by the PCI Security Standards Council.
- Overall, organizations that suffered a data breach were 50% less likely to be compliant than a normal population of PCI clients.
- All of the top 10 threat actions leading to the compromise of payment card data are well within scope of the PCI DSS. For most of them, multiple layers of relevant controls exist across the standard that mitigate risk posed by these threat actions."

Introduction

The histories of regulatory compliance and automobile traffic patterns have more in common than may be initially evident. Traffic, like business, operates most optimally in a free-flow pattern, but any number of events can disrupt the flow of automobiles to create a traffic jam. Such events may be major accidents, slow drivers in the passing lane, or changes in the driving environment like merging lanes and intersections. Thus, rules of the road are necessary to maximize flow while protecting the safety of all travelers, especially as these roads are linked together.

These complex transportation networks are similar to the interconnectedness of electronic systems designed to communicate with each other over information networks. We are all too familiar with the numerous events that can disrupt and/or compromise these networks. As a result, "rules of the road" for interconnected systems have manifested themselves in various ways, often in the form of compliance guidelines that regulate how we operate our businesses in a safe and secure manner. For organizations that process, store, or transmit payment card information, that role is filled by the Payment Card Industry Data Security Standard (PCI DSS).

The need for an industry standard to protect payment card data started in 2001 with the creation of payment brand specific compliance programs such as Visa's Cardholder Information Security Program (CISP) or Account Information Security (AIS) and MasterCard's Site Data Protection (SDP). In 2003, the CISP scope was expanded from processors and issuers to high volume merchants and service providers. Then, in 2006, the five industry payment brands came together to create the PCI DSS, which unified the prior compliance standards under one initiative while maintaining enforcement of the program with the individual payment brands.

Since then, the PCI DSS has received much attention and discussion. While many believe it to be a large step forward for the industry in terms of protecting cardholder data, others remain skeptical. These skeptics ask questions such as, "How do we know it's working?", "Is it the best mix of controls?", "Which controls are more effective than others?", "Is the bar set high enough?", "Is the bar set too low?", "Is it worth the investment?", "Does it adequately account for the differences between organizations?", and "Can it deal with changes in the threat environment?". These are all good questions and ones that should be asked about any prescriptive code of practice.

While this report cannot justify proponents of the PCI DSS or validate its skeptics (nor could any other single piece of research), it does lay some necessary groundwork in that direction by providing something that both camps need: data. To that end, this report analyzes findings from actual PCI DSS assessments conducted by Verizon's team of Qualified Security Assessors (QSAs). The report examines the progress of organizations toward the goal of compliance and includes topics such as how and why some seem to struggle more than others. Also presented are statistics around which PCI DSS requirements and sub-requirements are most and least often in place (or compensated for) during the assessment process. Finally, the report overlays PCI assessment data with findings from Verizon's Investigative Response services—the source for The Data Breach Investigations Reports (DBIR). This allows for a unique risk-centric analysis of the PCI DSS as well as one of the first-ever published comparisons between the practices of a "normal" population of organizations (clients assessed by our PCI services) and those that have suffered known security incidents (investigated by our Investigative Response (IR) services).

Though preliminary, we hope the findings and related discussion in this report help organizations approach PCI compliance in a more informed (and more prepared) manner. Whether starting, improving, or continuing a program, a better understanding of the state and struggles of peers should tune compliance efforts for a more successful experience during future assessments. Most importantly, it is our desire that this report—and other research that will follow—will ultimately lead to fewer payment card losses and help to measurably improve the security of financial transactions that are so critical to our economy.

Methodology

The bulk of this report is based on PCI DSS assessments performed by Verizon Qualified Security Assessors (QSAs); the remainder draws directly from payment card breach cases worked by Verizon's Investigative Response team[1]. During these on-site engagements, QSAs interview staff, review policies, verify documentation, and evaluate controls in order to validate the client's adherence to the PCI DSS. Within six weeks of the on-site assessment, Verizon issues an Initial Report on Compliance (IROC) to the client. The IROC provides the client (and Verizon) with a list of action items towards a clean and completed Final Report on Compliance (ROC or FROC). All assessment findings used for this report were culled directly from these IROCs and FROCs given to clients as part of paid engagements.

Verizon QSAs perform hundreds of PCI assessments each year. For several reasons, this report does not include them all. Rather, a simple selection process was used to create a sample of these reports for inclusion in the study. This process ensured a well-rounded sample that included a mix of QSAs, organizations of various types and sizes, IROCs and FROCs, and so forth. This resulted in roughly 200 assessments comprising the final dataset used in this report. The majority of these were conducted in the United States from 2008 to 2009.

While this report cannot justify proponents of the PCI DSS or validate its skeptics, it does lay some groundwork by providing something that both camps need: data.

The dataset includes IROCs/FROCs from PCI DSS 1.1 and 1.2. Differences in the versions were accounted for (as much as possible) during the analysis process and any references to specific requirements, sub-requirements, or testing procedures in this report use the v1.2 designation unless otherwise stated. Since data and observations are taken from both IROCs and FROCs, it is potentially difficult to discern from which set various statistics are drawn. For the sake of clarification, we note the source in all figures. As to how the determination was made on whether IROCs or FROCs should be used, we simply chose which source was most suited to the research question at hand. Nearly all assessment findings are drawn from IROCs since they are most informative when analyzing aspects of the DSS that posed more or less difficulty for organizations in the study (an FROC is not given unless all requirements are met). On the other hand, findings pertaining to compensating controls are taken from FROCs since they represent the official end-result of the assessment. General observations and commentary around these primary statistics are based upon the entire assessment process.

Finally, it is important to note that Verizon is committed to maintaining the privacy and anonymity of our clients. Client names and any other identifiers were removed from the assessments prior to analysis of the aggregated dataset. Data collection and analysis were conducted in cooperation with Verizon's RISK Intelligence team, which has years of experience handling sensitive anonymous information for the DBIR series. Furthermore, the findings in this report are presented in aggregate; results pertaining to particular organizations are never called out.

1 Verizon's IR team is the source for the well-known Data Breach Investigations Report series.

PCI DSS Assessment Results

The concepts of regulatory compliance, validation, and security are interrelated and therefore are often times confused by organizations. This confusion stems largely from two basic misconceptions. The first is simply that many believe that compliance is equal to "secure." This logic is flawed because security is not a binary state but, rather, a range in a spectrum from "absolutely secure" to "absolutely vulnerable" (though neither extreme exists in practice). So, to say that an organization is compliant against a standard only implies that an organization has a similar "degree of security" to that which the standard is designed to deliver.

Second, regulatory compliance is an attempt to address the polycentric nature of information security. As such, the degree of security that the standard is designed to deliver becomes a baseline that an organization should adhere to; one that assures it is doing its part to address the industry-wide risks present to all participants in the economic system. In the case of the payments industry it is the PCI DSS standard that sets the baseline and the QSAs that measure the level of adherence of an organization to the standard.

In order to better understand how QSAs measure this level of adherence, we must further draw a distinction between the terms "compliance" and "validation." Compliance is a continuous process of adhering to the regulatory standard. In the case of the PCI DSS there are daily (log review), weekly (file integrity monitoring), quarterly (vulnerability scanning), and annual (penetration testing) requirements that an organization must perform in order to maintain this continuous state called "compliant".

The degree of security that the standard is designed to deliver becomes a baseline that an organization should adhere to; one that assures it is doing its part to address the industry-wide risks present to all participants in the economic system.

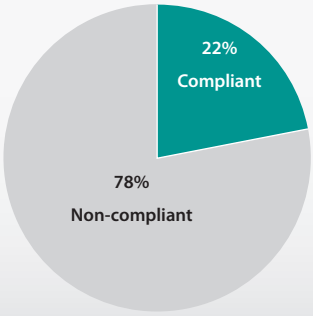
Validation, on the other hand, is a point-in-time event. It is a state of nature analysis that attempts to measure and describe the level of adherence to the standard. An organization may be able to pass validation in order to "achieve compliance" but then—once the QSA leaves—become lax about maintaining the degree of security the standard is designed to provide over time. As such, the goal of any organization should be to maintain its state of security in adherence with the minimum baseline compliance requirements set by the standard.

That's not to say that organizations only perform the minimum requirements of the PCI DSS; certainly, many organizations go above and beyond the standard. But the focus of this paper is on how well organizations adhere to the PCI DSS itself and is, furthermore, an analysis of how QSAs measured that level of adherence during the point-in-time validation of their respective environments. In discussing these findings, we take a top-down approach starting with overall compliance results before moving on to specific requirements and sub-requirements.

Overall Assessment Results

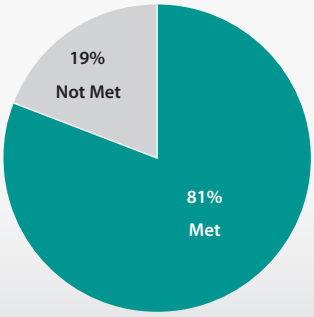
Of organizations assessed by Verizon QSAs within the scope of this study, 22% were validated compliant with the PCI DSS at the time of their IROC. Undoubtedly, this finding will elicit mixed reactions among readers. Some will find the number surprisingly low while others will believe it suspiciously high based on their own experiences or data. Whatever position one takes, the results are worthy of some discussion.

Figure 1. Percent of organizations found compliant at IROC



In examining the 22% of organizations found compliant, the majority of them shared three basic characteristics. Many were able to successfully remediate issues identified during the initial onsite assessment prior to completion of the IROC. Others were either veterans of the validation process (most of these passing IROCs were from 2009) and/or there were a significant number of sub-requirements within the PCI DSS deemed "Not Applicable" to them. It makes sense that organizations having more experience with the PCI DSS would fare better than those newer to it (though it should be remembered that we are discussing a minority (22%) of all organizations sampled; many others that achieved a passing ROC in 2008 were not compliant at their 2009 IROC). Similarly, it's not difficult to see why a smaller set of control requirements under evaluation would be easier to maintain and validate successfully.

Figure 2. Percent of testing procedures met at IROC



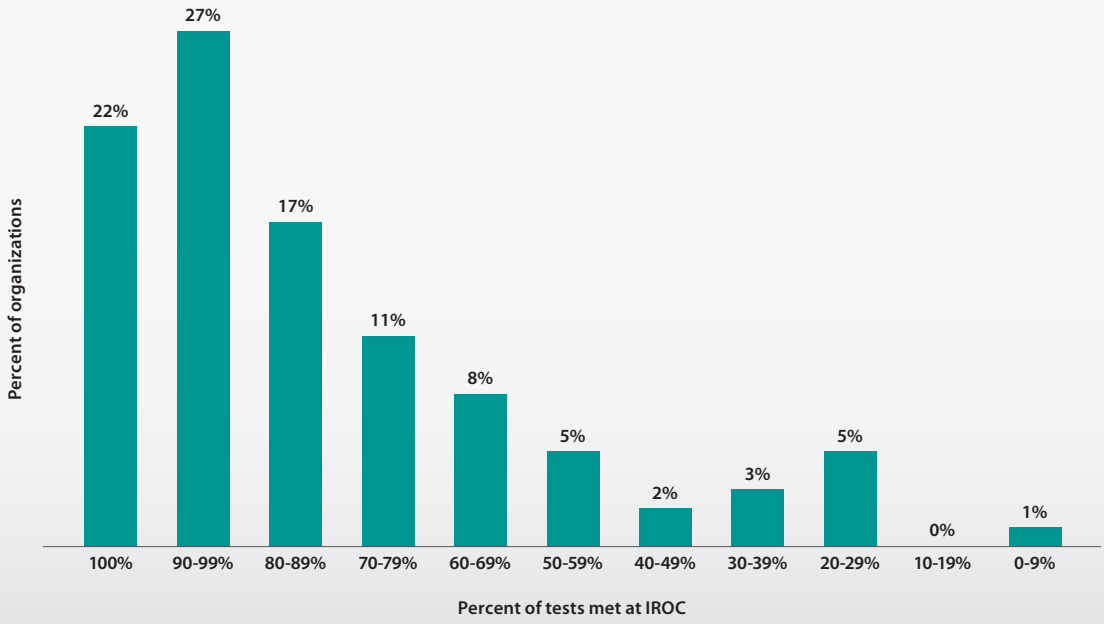
Perhaps most interesting about this result is that these organizations had at least some expectation going into the validation process that they would be found compliant and yet over three quarters of them were not. This is especially important to consider since most of them were not newcomers to the compliance process and many had successfully validated their adherence to the PCI DSS during a previous assessment. What are we to make of the fact that so many organizations struggle to maintain compliance from year to year? Is the standard too demanding or ambiguous? Is the validation process too difficult or inconsistent? Were these organizations simply unprepared or perhaps ill informed? These questions are, of course, without a single or simple answer. Any, all, or none of these could be valid reasons behind these findings. Lacking (and not expecting) a definitive answer but still desiring greater understanding, it helps to examine how non-compliant these organizations were. In other words, did they miss the mark by a little or by a lot?

What we found was that, on average, clients met 81% of all test procedures specified within the PCI DSS at the time of their IROC[2]. There are two ways to look at this finding. As former (or current) students, we may recognize this as an above average passing grade that we might be very proud of—especially if it was attained after a last-minute cramming session. On the other hand, organizations know that a passing grade for the DSS is 100% adherence to the standard. Despite their expectation of passing, those beginning the validation process failed one of every five testing procedures. Considering that the PCI DSS Requirements and Security Assessment Procedures have approximately 250 testing procedures (depending on how you count), this equates to about 50 unmet items per organization. That is not a very small number.

2 For those unfamiliar with the structure of the PCI DSS, each of the 12 main requirements has multiple sub-requirements. For each of these, the SAP specifies one or more testing procedures (or "tests" as we sometimes refer to them) that specify what the QSA is to validate. The tests are the unit of measurement actually marked "In Place" or "Not in Place" during the assessment.

Because single numbers cannot illustrate variability around the mean, Figure 3 shows the distribution of scores for organizations in our sample. As discussed above, 22% achieved the score of 100% they anticipated (teacher's pets), another 27% scored 90% or above, and so on. The fact that the distribution is skewed to the right is a positive sign. Three-quarters of organizations met at least 70% of the testing procedures, meaning they had a fighting chance at timely validation given a little more diligence. Only about 10% met less than half of the tests. All of this seems to suggest a "last mile" kind of problem with respect to the PCI DSS. Whether this is because the last 20% represents the most difficult controls to meet or because organizations concentrate on the majority they feel are most beneficial is unknown.

Figure 3. Distribution of testing procedures met at IROC



Several inferences can be drawn from these statistics around organizational compliance. First of all, it cannot be said that the PCI DSS is a group of simple/easy controls that are already inherent to the typical security program. Most organizations must do things they were not previously doing (or maintaining) in order to be validated compliant. Whether or not these are the right things to be doing is a separate question and one that we touch upon later in the Analysis of Investigative Response Data section of this report.

Furthermore, these findings demonstrate the importance of external validation against the standard. Most organizations appear overconfident when assessing the state of their security practices. The data also suggests that a significant proportion of these practices tend to erode over time, and that maintaining an ongoing approach to compliance is critical.

Next, since organizations are not meeting the requirements of the PCI DSS on their own, or are not maintaining them, we can deduce that the risk tolerance of the organization as implied by the maturity of the security program is above that of what is required as a baseline (the risk tolerance implied by nature of the DSS). This variation in tolerance for risk requires communication about the need for changes, both to senior management for sponsorship, and to the rest of the employee base as an expression of policy.

Finally, concerning the compliance programs of the organizations under study, the fact that 80% of them consistently missed 20% of items required suggests one of several things:

- They were unaware that the missing items were required;
- They were aware that the missing items were required but believed that they had them in place;
- They were aware that the missing items were required but believed that they had compensating controls in place;
- They were aware that the missing items were required, but believed that the validation process would not discover it;
- They were aware of the missing items, and as such, didn't really expect to pass upon issuance of IROC.

All of the above hint at very different problems within these organizations. They also hint at very different solutions necessary to address them. Understanding one's own particular stance and challenges with respect to compliance (or validation) is an important task. To offer additional help in that regard, we dig further into the state of compliance among the 12 requirements specified within the PCI DSS.

Assessment Results by PCI DSS Requirement

Thus far, we have shown that the majority of organizations do not meet their goal of 100% compliance upon initial assessment. We have also discussed how far off that mark they were. This information is interesting enough at a high level but becomes both more valuable and more actionable with increasing detail. Specifically, which requirements are readily achieved by organizations and which prove more difficult? This question can be approached in two ways—by identifying the percentage of organizations validated compliant with each of the 12 requirements or by identifying the average percentage of testing procedures (tests) within each requirement that were met at the time of the IROC. Table 1 depicts both of these statistics[3].

From Table 1, it is fairly straightforward to deduce that Requirements 4 (encrypt transmissions), 5 (AV software), 7 (logical access), and 9 (physical access) are less difficult to meet for most organizations. This should not be surprising, particularly if one keeps in mind that Requirement 7 (logical access) allows some room for interpretation and persuasion around exactly what is "need-to-know." Organizations have experience with physical security controls (R9) that stretches back much farther than the need to protect cardholder data. Encrypting traffic (R4) became prominent with the rise of the Internet and maintaining up-to-date AV software (R5) has become a well-familiar routine of paying subscription fees and letting auto-update do its work.

On the more difficult side of the spectrum, Requirements 3 (stored data), 10 (track and monitor), and 11 (regular tests) show lower levels of compliance. Again, this is unlikely to shock many security professionals. Protecting stored data (R3) can be tricky in execution given legacy systems, data tracking challenges, key management issues, etc. Regular testing (R11) and monitoring (R10) may be the most crucial but underrated and least appreciated aspects of security. Furthermore, the overhead and effort involved in these activities do not add to the ease of compliance.

One may argue that these findings have less to do with the difficulty of what is required than with the amount of what is required. In other words, is the variation in compliance scores among the twelve requirements simply a factor of the number of testing procedures defined under each? To assess simple volume-oriented relationships, we calculated the correlation between the total number of test procedures for each requirement and the two columns of data shown in Table 1. While there was little relationship between the number of tests and the percentage of tests passed ($r = -.025$), there was a reasonably strong negative correlation between the number of tests and the percentage of organizations meeting each requirement ($r = -0.61$). This means that the likelihood of successfully meeting a requirement decreases as the number of testing procedures

3 Note: Table 1 includes data from the 22% of organizations validated compliant at their IROC.

Table 1. Assessment findings at IROC by PCI DSS Requirement.

PCI DSS Requirement	% of Organizations*	% of Tests**
1: Install and maintain a firewall configuration to protect data	46%	82%
2: Do not use vendor-supplied defaults for system passwords and other security parameters	48%	77%
3: Protect Stored Data	43%	75%
4: Encrypt transmission of cardholder data and sensitive information across public networks	**63%**	83%
5: Use and regularly update anti-virus software	**70%**	**86%**
6: Develop and maintain secure systems and applications	48%	83%
7: Restrict access to data by business need-to-know	**69%**	**87%**
8: Assign a unique ID to each person with computer access	44%	82%
9: Restrict physical access to cardholder data	59%	**91**%
10: Track and monitor all access to network resources and cardholder data	39%	75%
11: Regularly test security systems and processes	38%	70%
12: Maintain a policy that addresses information security	44%	83%

Lowest three values denoted in red and highest three are in bold.

* In other words, 46% of organizations fully met Requirement 1 at time of IROC.
**In other words, organizations met an average of 82% of tests under Requirement 1 at time of IROC.

specified for that requirement increases. No surprises here. It is worth mentioning, however, that Requirement 11 bucks this trend, which further highlights the difficulty involved in meeting it (it has relative few tests and low scores).

Sticking with the correlation theme, comparing the columns in Table 1 reveals some other interesting findings. For the most part, the relationship is what one would expect; there is a strong correlation (r= .742) between the percentage of organizations meeting a requirement and the percentage of tests passed for that requirement. But the most intriguing data in this set are those items that don't correlate. Requirements 12 (security policy) and 8 (unique IDs) have the largest difference, though several others show similar disparity. This suggests there are a few tests within these requirements that hinder many organizations from meeting the overall requirement.

Before analyzing each requirement in detail, there is one final high-level observation critical to putting these findings in perspective. Similar to most business processes, security can be viewed as a recurring plan-do-check-act (PDCA) cycle. The planning phase consists of assessing risk and establishing risk tolerance; creating and updating policies, procedures, and programs that reflect this tolerance; and otherwise identifying what the organization wants to do with respect to security. With this done, the organization will begin to do (implement) the things that turn the plan into practice. Next,

smart organizations will check (validate) to make sure these practices are done well, according to plan, and functioning properly. If not (which is often the case), various actions will be required to remediate the discrepancy and maintain proper implementation of the plan. In relation to the PCI DSS, these phases can be viewed as follows:

- **Plan:** Requirement 12
- **Do:** Requirements 1, 2, 3, 4, 5, 6, 7, 8, 9
- **Check:** Requirements 10, 11 (though 1-9 contain "checks" too)
- **Act:** All requirements as needed, particularly those listed in the "Do" phase

Considering this breakout in light of Table 1 yields a key discovery: organizations are better at planning and doing than they are at checking. This is important to understand because checking is a prerequisite to acting. If the check phase is broken, organizations cannot react to events, remediate flaws, or maintain the state of security practices over time. As we have shown year after year in the DBIR, the overwhelming majority of data breaches (especially of cardholder data) come down to a failure to do what was planned. In nearly all cases, these failures should have been found and remediated (without great expense) if adequate checks had been in place.

Organizations are better at planning and doing than checking. If the check phase is broken, they cannot act to maintain the state of security over time.

Detailed Assessment Results by PCI DSS Requirement

With these big picture takeaways in mind, we will continue the analysis of initial assessment findings by examining the PCI DSS requirements in more detail. Under each requirement, the standard specifies various sub-requirements and testing procedures that must be in place to validate compliance. What follows is an evaluation of the twelve requirements with particular attention given to specific items that significantly helped or hurt the ability of organizations to achieve a passing score. Also identified are sub-requirements that are often not applicable or that are satisfied through compensating controls.

Requirement 1 (firewall configuration)

Approximately 46% of organizations initially satisfied Requirement 1, and on average they met 82% of the related testing procedures. Firewalls are standard fare for security professionals, but they can multiply like rabbits if not kept in check and it is fairly common to have runaway rule sets.

The most difficult test in this area involves verifying that firewall rule sets are reviewed at least every six months (1.1.6). More specifically, there was an inconsistency between the practice of reviewing firewall and router rule sets and the ability to provide documentation that the review took place. Along the same line is control 1.1.5b, which concerns documentation and business justification of insecure services, protocols, and ports such as FTP or TELNET. It would appear that the paperwork drill wasn't working well in many instances but the actual traffic, at least, was being sufficiently restricted. Part of the reason for this, we believe, stems from the tools used to administer firewall rule sets. The tools that present a graphical user interface typically have more (and friendlier) capabilities to document rules and changes. Fewer text-based administrative interfaces allow for in-depth comments, creating the need to maintain documentation external to the firewall.

Additionally, we found a divergence in how organizations scored on tests around inbound and outbound traffic. Results show that organizations perform reasonably well on restricting inbound traffic but are much more permissive when it comes to outbound rules (i.e. allowing all desktops SSH, FTP, Telnet to ANY). Egress filtering has been around for some time, but it appears organizations struggle to strike a balance between potential business productivity gains and losses when restricting outbound access.

Requirement 2 (vendor defaults)

Default passwords, settings, and configurations are common attack points for hackers because they are such easy fare. As evidenced by the 48% that initially passed Requirement 2, many organizations have difficulty eliminating them. There were three big reasons clients didn't have more success with this requirement: they didn't sufficiently harden systems by turning off extra services (2.2.2) and functionality (2.2.4), they didn't document why certain services and functions could not be removed due to business reasons (as required by 2.2.2 and 2.2.4), and they didn't encrypt all non-console admin traffic (2.3).

With regard to 2.3, many of the issues relate to legacy systems and equipment still in production. For instance, many companies continue to use TELNET for older network equipment (even if some of the newer devices support SSH). This is because most find it easier operationally to use the same method for administration across the organization. Additionally, some legacy systems use terminal emulation sessions for administration, which are not encrypted. Sub-requirement 2.3 is among those most often met through compensating controls due to the expense of replacing such systems and methods.

There are a couple of sub-requirements within Requirement 2 that are often cited as not applicable to the organizations within our sample. The first is 2.1.1, which covers the configuration of wireless networks. If wireless networks do not exist within your environment, you need not concern yourself with this requirement. Similarly, if you are not a shared hosting provider, then you need not worry about 2.4.

Requirement 3 (stored data)

Many organizations that process and transmit payment card data will also store it. These data stores are prime targets for an attacker and thus require specific protections defined under Requirement 3. Seasoned security professionals know, however, that creating a strong encryption program for data-at-rest isn't easy and our assessment results bear this out. Only 43% of organizations had their ducks in a row for this requirement (bottom-tier among the 12 requirements) and they passed 75% of the tests defined within it (also bottom tier).

On the bright side, more than 90% of organizations succeed in one of the more important sub-requirements, 3.2, by not storing authentication and track data or card verification codes. On the not-so-bright side, it's hard to single out a few things that were substandard for Requirement 3 because so many deserve that distinction (over one-third of the 30+ tests scored below 70%). Most of the difficulty surrounds two major aspects: the use of strong encryption (3.4) and key management (3.6).

Sub-requirement 3.4 mandates that, at a minimum, the Primary Account Number (PAN) be rendered unreadable, by hashing, truncation, index tokens, or strong cryptography. Since most of the trouble centered on the latter, we'll turn our attention there. The exact means used to encrypt PANs (there are multiple options) do not matter as much as the end (encrypted data). One of the keys to understanding the difficulty inherent to this is to recall the various forms in which payment card data can be stored. Database entries, digital images, flat files, audio recordings, and batch settlement files are just a few. While an organization may choose to leverage native database encryption or a third party tool to encrypt its entries, the problem grows when attempting to encrypt actively accessed flat files stored in a common access folder on a file server. Some may instead attempt to leverage file-level encryption but this is not always an option on midrange and mainframe systems. The complexities and cost unfold further when custom-built payment applications (each with its own individual data store) are in play. For all these reasons and more, 3.4.a and .b are compensated for more than any other tests in the DSS.

Another big hurdle within Requirement 3 involves key management and rotation. As Adi Shamir's (the "S" in RSA) Third Law tells us, encryption is more often bypassed than it is penetrated. Poor key management is often the guilty party when it is. All tests under 3.6 exhibit low pass rates, especially 3.6.3, 3.6.4, and 3.6.5, which fell below 70%. 3.6.4 (the least-implemented of the bunch) mandates that encryption keys be rotated at least annually. The challenges here are similar to those for strong

encryption. For example, a hardware security module (HSM) can relatively easily encrypt database columns and rotate the keys on a per record basis. Porting this same functionality to flat files or transaction logs is much harder, since key rotation may actually require a complete extraction, decryption, and re-encryption with a new key. Additionally, documentation proves problematic in 3.6.8 where formal key custodian forms and sign-off procedures are not common practices.

There were two areas within Requirement 3 commonly rated not applicable among our sample. The first, 3.6b (key management processes), is easy to explain as it is directed only at service providers. The other is 3.4.1, which tests the implementation of whole disk encryption. Due largely to a business model dependent upon retrieving credit card numbers and performing real time lookups, speedier options are often preferred over whole disk encryption.

Requirement 4 (encrypted transmissions)

Many end-users believe this is the most important security element of sensitive transactions. The practice of encrypting transmissions was brought about largely by IT professionals to encourage use while assuaging fears about security. While the public Internet need for encrypted transmissions is suspect, things can get dangerous at either end, as shown by major breaches in the past. Almost two-thirds of organizations get this right the first time, and the correspondingly high 83% test-passing rate falls in line.

Since not all organizations use wireless networks, sub-requirement 4.1.1 is a regular on the Not Applicable list. For those that do, we are curious to see if compliance to 4.1.1 will drop now with the passing of the deadline set for eliminating Wired Equivalent Privacy (WEP) usage (June of 2010).

In comparison to the others, sub-requirement 4.2 is fairly often overlooked. It prohibits the sending of unencrypted Payment Account Numbers (PANs) through end-user messaging technologies. Organizations often send credit cards via email without really addressing security concerns (e-fax, customer service reps, accounting). Email (especially if internal-to-internal) is often perceived as private and escapes the examination of information security teams with respect to sensitive data protection.

Requirement 5 (AV software)

At 70%, using and updating AV software enjoys the highest initial compliance rate of the 12 requirements. It is also near the top with regard to the percentage of test procedures passed (86%). Even so, these numbers fall below the high 90 percent levels often reported in industry surveys on AV implementation. Previous research has shown, however, that analyzing the quality of specific practices around AV software (rather than, for instance, just asking if AV is used) reveals lower rates of adoption similar to what is shown here[4]. Either way, there is little doubt that AV software is one of the most commonly employed security solutions in industry today.

Broad adoption of AV software (and other AV controls) is at least partially due to the series of massive and widely publicized malware outbreaks in the early 2000s (ILOVEYOU, Code Red, Nimda, SQL Slammer, Blaster, etc.). Additionally, maintaining modern enterprise-level AV solutions is a fairly turnkey operation. That's not to say these solutions are without flaw; recent questions about the adequacy of signature-based tools continue to heat up as malware variants, customization, and packing tools rise dramatically.

Requirement 6 (development and maintenance)

When one considers everything under the domain of Requirement 6, the fact that almost half of organizations satisfied it at the IROC stage is quite surprising. The equally (if not more) surprising 83% pass rate for all test procedures suggests the presence of a few pesky sub-requirements causing most of the trouble. Organizations appear relatively successful at identifying vulnerabilities (6.2), traditional development (6.3), and change control (6.4). They falter in patching (6.1) and web application development (6.5).

4 Baker, W. and Wallace, L. "Is Information Security Under Control? Investigating Quality in Information Security Management." IEEE Security and Privacy, Vol. 5, No.1, pp 36-44, 2007.

With regard to 6.1, we see that patching on a monthly basis is not a common practice. Many companies maintain a quarterly schedule for various reasons. From a risk perspective, the benefit of monthly over quarterly patch deployment is negligible in most situations, yet the additional cost can be significant. The regularity of self-inflicted denial of service incidents tied to untested and/or hastily deployed patches is a deterrent too. Organizations sometimes shy away from "bleeding edge" and patching is one of those areas where many feel the "status quo" is just fine. Finally, applications and network equipment are often only patched in break/fix situations or for major upgrades. In light of this, it is understandable why 6.1 is among those most often submitted for compensating controls.

Another trouble spot for organizations in our sample concerns secure web application development (6.5). The complexity, exposure, nature, and role of web applications make this an inherently difficult task. However challenging, the ample data on related attacks should be disturbing enough to motivate action. Due in part to these trends, some will elect to outsource development instead (resulting in a larger number of "NAs" for 6.5).

Requirement 7 (logical access)

Speaking frankly, that Requirement 7, which restricts access to need-to-know, boasted the second highest scores (69% of organizations, 87% of tests) among the 12 surprised those of us more familiar with breach investigations than PCI DSS assessments. However, a little conferring among ourselves brought up one of life's most time-tested lessons—appearances can be deceiving.

Remember that the goal for many organizations is to achieve compliance against the standard and thus many choose the shortest path towards that goal.

Remember that the goal for many organizations is to achieve compliance against the standard and thus many choose the shortest path towards that goal. The concept of need-to-know—and more importantly, the determination of what constitutes it—is not a binary issue. When QSAs assess adherence to Requirement 7, they attempt to confirm that privileges are indeed restricted to those having a legitimate need for them in order to perform business responsibilities. Because it would be cost prohibitive for all involved to analyze every user's permissions to every system in detail, the aim is to ascertain if, in general, the organization follows the principle and to collect some evidence of that. Since the client alone has all the knowledge about what is necessary based on business need, it is often difficult for the QSA to debate the matter (within the bounds of the standard) except in the more flagrant violations.

In other words, it seems that enough ambiguity exists in the concept, the standard, and the validation process to give these scores a boost. The important thing here—the one that is hard to audit—is whether organizations are truly asking themselves the hard question of "Is this really necessary?" and doing something about it when the answer is "No."

Requirement 8 (unique IDs)

The title of this requirement is somewhat misleading since it covers many aspects of user account management from unique IDs to authentication, passwords, timeouts, lockouts, and decommissioning. While Requirement 8 lies in the middle of the pack in terms of compliance scores, it can be a formidable challenge depending on the environment and circumstances.

Of the sub-requirements, those defined in 8.5 (user authentication and password management) showed comparatively lower scores. It is interesting that password length (8.5.10 at 68%) and change (8.5.9 at 68%) requirements were the lowest of all. These might be aggravating for users but they are normally not terribly difficult to implement and enforce. Perhaps organizations are skeptical as to whether seven character passwords and 90-day rotation provide sufficient risk reduction to offset the extra administrative overhead and impact on usability.

Several of the testing procedures under Requirement 8 are among those in the DSS most often met through compensating controls. Again, 8.5 is the main culprit. Legacy systems (or others) that lack the necessary features or centralized management and authentication mechanisms are the primary reasons for this. If not managed centrally, maintaining hundreds of local individual accounts on certain network equipment, UNIX, and POS systems, can become intractable.

Requirement 9 (physical access)

Test procedures defined by Requirement 9 held the pole position among the 12 with the highest pass rate of all. Over half of the "top 10 list" across the DSS can be found within this requirement. These tests focus primarily on verifying physical controls around payment infrastructure, including data centers, wireless devices, and backup tapes.

For the most part, these high marks match expectations. Many find physical security easier to conceptualize than electronic data security; it is familiar and tangible. Physical controls are traditionally strong around data centers, relatively easy to apply to centralized physical devices and media, and are frequently assessed due to numerous regulations and standards that require them. Furthermore, physical controls typically do not require the level of daily care and feeding that more complex and dynamic genres (i.e., log monitoring and review) need to remain effective and in proper order.

The one item in Requirement 9 that does trip up a number of organizations is 9.1.1, which mandates video cameras or other mechanisms to monitor physical access to sensitive areas. This probably has something to do with the cost of installation, setup, and storage capacity for three months worth of data. Once deployed and coverage is directed at cardholder environments, organizations have no real difficulty maintaining compliance.

Requirement 10 (tracking and monitoring)

In our discussion of the high compliance to Requirement 9, we attributed some of that success to the more static nature of physical controls. In many ways, Requirement 10 is the antithesis of Requirement 9 because it involves constant attention to a complex and ever changing set of circumstances across potentially thousands of disparate devices. Furthermore, audit logging and monitoring is often a thankless task since its intended purpose is to (1) alert on suspicious activity and (2) facilitate a forensic investigation. Few consider it fun to learn of or look into "bad things," and this is probably another factor in the low percentage of organizations able to initially pass Requirement 10 (39%) and the low percentage of test procedures they are able to meet (75%).

Understanding the structure of Requirement 10 is helpful in comprehending the challenge it presents. The sub-sections flow in logical order, each one dependent upon the next. 10.1 and 10.2 necessitate that audit logs be enabled to track user activity on all systems in scope while 10.3 specifies what those logs must contain. 10.4 requires time-synchronization of logs for event reconstruction and 10.5 covers various aspects of securing logs from unauthorized alteration. A review process for anomalies must be in place to satisfy 10.6, and finally, 10.7 sets parameters around log retention timeframes. Nothing to it, right? Wrong. Organizations tend to struggle in all of these areas, most notably with generating (10.1 and 10.2), protecting (10.5), reviewing (10.6), and, to a lesser extent, archiving (10.7) logs.

Meeting 10.1 and 10.2 requires logging the entire trail of a user across the network to accessing cardholder data. The main difficulty here seems to be the sheer breadth and number of information assets involved. Organizations typically have no serious problems implementing logging on network devices and operating systems but fail to do so for applications. There are many custom and legacy payment applications in use, and they often lack the logging capabilities necessary to meet the standard. These situations may require additional logging at either the operating system or database layer in order to compensate for the lack of application layer audit logs. Of course, even when all systems within an organization support logging, it does not mean the feature will be enabled.

With regard to securing the logs from alteration, there are two factors at work: 1) most organizations' standard operating procedure is to allow whatever logs are enabled by default to write locally and get overwritten according to default/disk space, and 2) file-integrity monitoring (FIM) is not implemented. If logs are constantly overwriting themselves, or "self-altering," then the organization can't hope to meet sub-requirement 10.7 (maintaining an audit history). This issue crops up when organizations, because of storage constraints, do not have a centralized logging solution to off-load logs prior to their being overwritten. 10.5.5, or the requirement for FIM, is quite complex, difficult, and expensive to implement. The difficultly for most organizations lies in hashing dynamic log files, which causes them to seek compensating controls for 10.5.5.

The dilemma faced in trying to monitor and review logs effectively may be unique to the security industry but the root issue is one that affects the entire modern world: the amount of information available far exceeds our ability to extract meaning from it. Anyone even remotely familiar with log monitoring understands this concept. Because it would be an impossible feat to accomplish with the naked eye alone, analytical tools are used to automate the parsing of logs. This is certainly of huge benefit, but like any other tool, they must be tuned to extract the needles from the haystacks (or at least find the haystacks as we recommend in the 2010 DBIR). As if the problem weren't difficult enough, many organizations make it even harder on themselves by logging everything rather than taking the time to identify, prioritize, and enable only what is required and/or useful in the context of their payment environment.

Just in case this wasn't overwhelming enough, the heaps of log files generated must be stored somewhere. 10.7 requires organizations to archive logs for one year to support a forensic investigation with at least the last three months ready for immediate analysis. Storage is cheaper by the day, but the number and size of log files and the operational headache of maintaining them offset these savings.

Requirement 11 (regular testing)

It would appear that many security professionals haven't read "The Blind Side," else they might have a better appreciation for the more mundane controls within our vocation. Among various other side stories, the book explains how the position of left tackle in American football (a rather mundane "control" to protect the quarterback), came to be considered one of the most important and highest-paid roles on a field usually dominated by flashy, high-scoring types. Perhaps it should be required reading for all new hires. At any rate, an organization that does not conduct regular tests of its systems and security practices is bound to give up some bone-crushing sacks. With Requirement 11 at the very bottom in terms of the initial compliance rate (38% of organizations) and the typical share of test procedures met (70%), it looks like an injury-prone field.

Though everything under Requirement 7 shows dismal results, the dubious distinction of the least-implemented across the entire DSS at 49% went to 11.5—file-integrity monitoring. Without doubt, this sub-requirement presents a formidable challenge. In fact, when paired with its corollary in Requirement 10 pertaining to FIM of audit logs (10.5.5), it is among the most difficult practices to achieve. At one time, this requirement was only considered necessary for protecting the "crown jewels" or high-risk assets, but the PCI DSS now makes it applicable to all systems in the cardholder data environment. This brings a number of legacy and mainframe systems in scope of this sub-requirement that may not natively support FIM such as Tandem, AS400, MS DOS, and so on. As such, 11.5 is sometimes satisfied through compensating controls.

We cannot touch on all troublesome areas under Requirement 11, but it's worth discussing 11.3 and 11.2 to round out the bottom 3. 11.3 specifies that an annual penetration test be performed by a qualified internal or external party, the idea being to check whether existing controls are able to repel a skilled and determined attacker. For example, an organization may configure the system settings (2.2), apply the proper security patches (6.1), and test the web application for vulnerabilities (6.5), but a penetration test will tie all of these point controls together to determine if they work in concert to protect the payment infrastructure. Organizations typically have a list of findings to remediate after a penetration test before it can be

re-run. Whereas some of the issues uncovered by the test may be addressed through simple configuration (i.e., turning on NTP or enabling SSL), others will be time-consuming to investigate and remediate. For instance, many testing tools perform application banner grabbing and make "best guess" determinations if a system is susceptible to an attack. Due to tool limitations, it becomes extremely difficult for a penetration tester to fully validate a finding, yet further probing could possibly affect the security attributes of the assets getting tested. This can lead to many false positives and/or unanswered questions that require additional time to remediate and follow-up. For this reason, penetration tests (and others that may prompt remedial work) should be performed earlier in the process.

Sub-requirement 11.2 entails network vulnerability scans and faces similar issues to penetration testing with respect to validating false positives. The main challenge for organizations here, however, is that the scans must be run quarterly. Maintaining a recurring process—along with evidence that it was followed—proved difficult to many in our sample. Furthermore, organizations that fail to perform quarterly scans (or provide evidence that they did) may find it even more difficult in the future because the assessor will often require additional evidence that a more rigid process is in place going forward.

All in all, many struggle with Requirement 11 because it is seen as mundane and it requires a routine. It goes against the natural tendency of many organizations, which is to simply to fire and forget when deploying systems, practices, and procedures. Unfortunately, successful security doesn't work that way.

Requirement 12 (security policies)

A rather low 44% of organizations met this requirement during their initial assessment but (on average) they passed a reasonable 83% of the test procedures within it. Security policies usually are not difficult to create or validate but when it comes to their true effectiveness, you tend to get what you pay for. What really matters is the quality of their content, how well they are known, and how well they are followed. As mentioned already, there is often a big discrepancy between codified policy and actual practice. This causes some to believe policies have little value, but that sounds suspiciously like the baby and bathwater idiom to us.

Although most tests within Requirement 12 show decent marks, there are some opportunities for improvement. Having an incident response plan, addressed in 12.9, is one such opportunity and the two items under that needing the most work are 12.9.2 and 12.9.4. The first (12.9.2) stipulates documentation and—big surprise—it seems while organizations may create response plans, rarely do they update or review them. Neglecting to adequately train responders on their roles and responsibilities in the event of an incident (12.9.4) is the second stumbling block. We could talk about why both of these spell trouble but we did that recently (with pretty pictures) in the "Discovery to Containment" section (p. 47) of the 2010 DBIR. Don't forget those five P's!

PCI DSS Milestone Analysis

Another way to analyze adherence to the DSS is through the lens of the Prioritized Approach, which was created by the PCI Security Standards Council to "help stakeholders understand where they can act to reduce risk earlier in the compliance process.[5]" The approach defines a roadmap of activities around six ordered milestones, the idea being that early milestones address the most critical risks. These milestones, which are listed in Table 2, are not organized around the twelve requirements; rather, sub-requirements are broken out and organized around the six milestones. Because the Prioritized Approach is relatively new (2009), one may wonder if organizations have begun to order their compliance efforts around these milestones. Table 2 provides data on this topic.

5 From the "The PCI DSS Prioritized Approach" available at https://www.pcisecuritystandards.org/education/docs/Prioritized_Approach_PCI_DSS_1_2.pdf

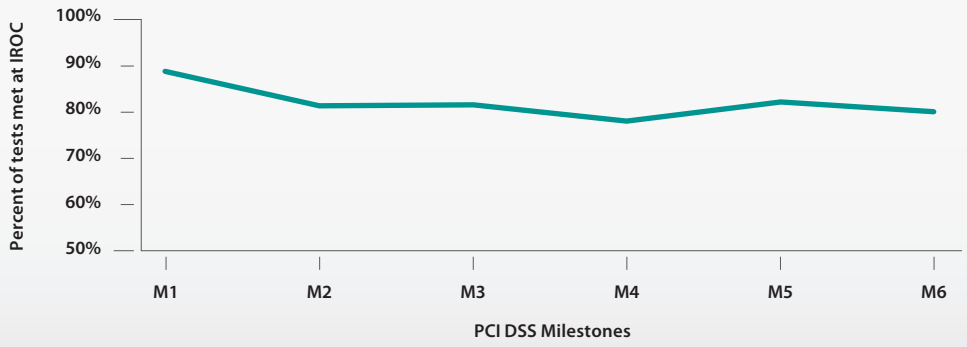
Table 2. Percent of test procedures met at IROC by priority milestone*

Milestone	Goal	In Place
1	Remove sensitive authentication data and limit data retention.	88%
2	Protect the perimeter, internal, and wireless networks.	81%
3	Secure payment card applications.	81%
4	Monitor and control access to your systems.	79%
5	Protect stored cardholder data.	83%
6	Finalize remaining compliance efforts, and ensure all controls are in place.	80%

* Based on the PCI DSS Prioritized Approach from the PCI Security Standards Council.

It should be remembered that in order to validate compliance to the PCI DSS, all test procedures must be met regardless of their milestone classification. That said, one would expect an IROC (which, as we know, is often imperfect) to show higher scores for the early milestones if organizations are utilizing the Prioritized Approach. Assuming that Milestone 1 meets its goal of addressing the most critical risks first, it is good to see that it indeed shows the highest implementation. Beyond that, though, the remaining milestones show roughly the same percentage of controls in place. The relatively flat line in Figure 4 depicts this trend well.

Figure 4. Percent of testing procedures met at IROC by priority milestone*



* Based on the PCI DSS Prioritized Approach from the PCI Security Standards Council.

Based on this, it does not appear that the Prioritized Approach has significant adoption among organizations in our sample. However, it must be noted that our sample is comprised of 2008 and 2009 assessments. Since the Prioritized Approach is a 2009 publication, it is entirely possible that organizations have not yet had sufficient time to implement it. Furthermore, we expect that it will take some time before evidence of this transition will be seen in the assessment findings.

Analysis of Investigative Response Data

For the past several years, Verizon has published a series of reports covering forensic engagements worked by its Investigative Response (IR) team. The Data Breach Investigations Reports dig into the who, what, when, where, how, and why of organizational data breaches and pass these findings on to the public. The series spans six years of data, over 900 breaches, and more than 900 million compromised data records. Since many of these losses involve payment card information, this extensive dataset offers two very interesting and unique lines of analysis with respect to PCI DSS. The first compares organizations assessed by our QSAs to payment card breach victims who engaged our IR team. The second lists the top threat actions used to compromise cardholder data in IR cases worked over the past two years.

Table 3. Percent of organizations meeting PCI DSS requirements. IR data based on post-breach reviews; PCI data based on QSA assessments at IROC.

PCI DSS Requirement	PCI Data	IR Data
1: Install and maintain a firewall configuration to protect data	46%	32%
2: Do not use vendor-supplied defaults for system passwords and other security parameters	48%	41%
3: Protect Stored Data	43%	19%
4: Encrypt transmission of cardholder data and sensitive information across public networks	**63%**	**77%**
5: Use and regularly update anti-virus software	**70%**	**58%**
6: Develop and maintain secure systems and applications	48%	12%
7: Restrict access to data by business need-to-know	**69%**	27%
8: Assign a unique ID to each person with computer access	44%	26%
9: Restrict physical access to cardholder data	59%	**49%**
10: Track and monitor all access to network resources and cardholder data	39%	15%
11: Regularly test security systems and processes	38%	19%
12: Maintain a policy that addresses information security	44%	25%

Lowest three values denoted in red and highest three are in bold.

One of the common arguments made by skeptics of the PCI DSS is that there is relatively little evidence supporting its effectiveness.

Comparison to IR Assessments

One of the common arguments made by skeptics of PCI DSS is that there is relatively little evidence supporting its effectiveness. Since results-based studies around PCI are scarce, the objection is at least understandable (whether or not the premise is true). An exhaustive and controlled study comparing the security failures/losses of highly compliant organizations to those of non-compliant organizations would go a long way toward settling the dispute (or at least giving it some empirical fodder).

Though neither exhaustive nor controlled, assessments conducted by Verizon's PCI and IR teams do allow comparison between organizations for which compliance and security results are at least partially known. At the culmination of a forensic engagement, the lead investigator performs a review of PCI DSS requirements and conveys these findings to the relevant payment card brands. This exercise is not an official assessment, but it does provide insight into which requirements tend to be deficient among breach victims.

Table 3 shows PCI DSS compliance results for two groups of organizations. The first includes the same sample of PCI clients discussed throughout this report (Results mimic Table 1). The second group consists of organizations suffering a confirmed data breach investigated by our IR team between 2008 and 2009.

Figure 5 presents the same data as Table 3, but the message is much more apparent: IR clients exhibit a lower likelihood of meeting PCI DSS requirements than do PCI clients. Said differently, breach victims are less compliant than a *normal*[6] population of organizations. This is true across all requirements except 4 (encrypted transmissions). As will be discussed in the next section, techniques attempting to compromise data traversing public networks were not a common threat action across our caseload. Though the disparity between the groups fluctuates per requirement, on average, PCI clients scored better than breach victims by a 50 percent margin[7]. So while "prove" is too strong a word to use in this case, these results do suggest that an organization wishing to avoid breaches is better off pursuing PCI DSS than shunning it altogether.

Figure 5. Percent of organizations meeting PCI DSS requirements. IR data based on post-breach reviews; PCI data based on QSA assessments at IROC.

6 "Normal" is not used here in the statistical sense. It simply refers to the fact that the group of PCI clients represents a set of organizations with no known atypical characteristics other than they all utilized our PCI assessment services.

7 It should also be considered that these results represent an IROC. Most of the organizations in the PCI dataset addressed the deficiencies shown here and were eventually validated fully compliant. In this respect, the difference between the IR and PCI datasets could be said to be even greater.

Top Threat Actions

All analysis prior to this point has been vulnerability-centric (or control-centric). The ultimate purpose of the PCI DSS is not just to establish a set of requirements but to reduce losses of cardholder data, therefore, a risk-centric perspective is relevant and useful to this study. For such a perspective, we draw from breach investigations worked by Verizon's IR team. Table 4 lists the Top 10 threat actions leading to the compromise of payment card data from 2008 to 2009. Threat actions describe what the threat agent did to cause or contribute to the incident. Though VERIS recognizes seven categories of threat actions[8], only three are present in the top 10 list (and one of those is mentioned only once). The percentages in Table 4 add up to more than 100% because most breaches involve more than one action. The remainder of this section discusses these threat actions as well as the PCI DSS requirements that can deter, prevent, and detect them.

A BRIEF PRIMER ON VERIS

The threat actions in Table 4 are based upon the Verizon Enterprise Risk and Incident Sharing (VERIS) framework. VERIS is designed to provide a common language for describing security incidents in a structured and repeatable manner. It takes the narrative of "who did what to what or whom with what result" and translates it into the kind of data you see presented in the DBIR. The framework is available for free public use and can be accessed from the VERIS community wiki[9].

Table 4. Top threat actions based on 2008-2009 payment card breaches investigated by Verizon IR team

Category	Threat Actions	% of Breaches
Malware	Backdoor	25%
Hacking	SQL Injection	24%
Hacking	Exploitation of backdoor or command/control channel	21%
Hacking	Exploitation of default or guessable credentials	21%
Misuse	Abuse of system access/privileges	17%
Hacking	Use of stolen login credentials	14%
Malware	RAM scraper	13%
Hacking	Exploitation of insufficient authorization	13%
Malware	Packet sniffer	13%
Malware	Keylogger / Spyware	13%

8 The seven threat action categories defined by VERIS are Malware, Hacking, Social, Misuse, Physical, Error, and Environmental.
9 https://verisframework.wiki.zoho.com

Backdoors

In examining the threat actions from Table 4 it is convenient to organize the discussion around several logical groupings. Backdoors (Malware) and the exploitation of them (Hacking) are both in the top three and represent a good starting point[10]. Backdoors are tools that provide remote access to infected systems by bypassing normal authentication mechanisms and other security controls. While they can infect systems via any number of vectors, the most common across Verizon's cases was direct installation or injection by remote attackers. With the backdoor established, an attacker can exploit it to access the system at will and engage in all manner of nefarious activities. Backdoors are popular tools because they facilitate the goals of concealment and persistence that cybercriminals crave. They are also the predominant means of exfiltrating payment card data from the victim's environment.

In terms of PCI DSS requirements relevant to the installation and exploitation of backdoors, Requirement 5 (AV software) is an obvious one. It seems counterintuitive that one of the most implemented control areas relates to the most frequent threat actions. But as discussed in the DBIR, modern malware is highly customized (more than half according to the DBIR) and installed through vectors that evade AV software. That doesn't mean AV is useless, it just means attackers are adapting and other controls are needed as well. Because it is not signature based, file integrity monitoring (Requirement 11) holds promise as an effective measure against backdoors and other malware ("holds promise" is the operative word here; organizations struggle to get this right). Several other areas of the PCI DSS can help as well, but Requirements 1 (firewall configuration) and 10 (tracking and monitoring) are particularly relevant. Unfortunately, they are among the least implemented. Backdoors often operate via odd ports, protocols, and services, so ingress and egress filtering can be very effective in locking them down and aren't forced to rely on a known signature. Monitoring and logging network access (if actually monitored) provides another layer of defense and is useful for detecting suspicious traffic that may signify the presence of a backdoor.

All analysis to this point has been vulnerability-centric but a risk-centric perspective is relevant and useful to this study.

SQL Injection

The next most frequent threat action, at 24% of payment card breaches, is SQL injection. It is a technique that exploits how web pages communicate with back-end databases. At a very high level, the attacker inserts SQL statements into the application through the web server and gets the answer to their query or the execution of other SQL statements. If the application trusts user input and does not validate it at the server, it is likely to be vulnerable to SQL injection, cross-site scripting, or one of the other input-validation vulnerabilities. In data breach scenarios, SQL injection has three main uses: 1) query data from the database, 2) modify data within the database, and 3) deliver malware to the system. The versatility and effectiveness of SQL injection make it a multi-tool of choice among cybercriminals.

Since SQL injection is almost always an input validation failure, Requirement 6 (development and maintenance) is critical to thwarting it. Rather than waiting until the application is complete, considering security issues throughout the development lifecycle is paramount to creating secure applications. Once developed, applications should be tested at regular intervals to verify controls are in place and up to par (Requirement 11). All the while, these applications should have logging enabled and be monitored (Requirement 10) to identify SQL injection attacks as they occur or shortly thereafter. Of course, none of this will help at all if your organization, like so many of those in Table 3, neglects these requirements altogether.

10 It may seem odd to split these two actions, but there is a reason for it. VERIS classifies incidents as a series of discrete but related events. The introduction of malware to a system that opens a backdoor is viewed as a separate event from a threat agent using that backdoor to gain unauthorized access (the second is dependent upon the first but not certain to occur).

Authentication and authorization attacks

The next group of threat actions target authentication (who you are) and authorization (what you can do) mechanisms. Exploitation of default or guessable credentials, the use of stolen login credentials, and the exploitation of insufficient authorization are consistently among the most prevalent and damaging attacks against cardholder data. Many systems and devices come preconfigured with standard usernames and passwords to allow initial setup. Because these are widely known by criminals (and because they're often simple), failure to change them often results in unauthorized access. This is especially prevalent in the hospitality and retail space where point-of-sale (POS) systems are managed by a third party. If an attacker successfully steals valid user credentials, subsequent actions will appear to come from a legitimate user and are much less likely to be tagged as malicious by detection mechanisms. It also makes it easier to cover his tracks as he makes off with the victim's data. Access control lists (ACLs) are designed to specify which entities can access an object and what operations they can perform. If these authorization mechanisms are missing, weak, incorrectly scoped, or misconfigured, attackers can access resources and perform actions not intended by the victim.

Several requirements are designed to mitigate these threat actions. Eliminating default and guessable credentials is one of the main purposes of Requirement 2. Those that slip through the cracks should be found and remediated if an organization is regularly testing security systems and processes in line with Requirement 11. Since malware is commonly used to swipe credentials, Requirement 5 (AV software) can help prevent and detect known password-stealing malware. Restrictive firewall rules and network segmentation (Requirement 1) help shore up insufficient authorization at the network level, while restricting access to need to know (Requirement 7) works well at the application and system level. The large disparity between PCI clients and IR clients on Requirement 7 (see Table 3 or Figure 5) is interesting. Security professionals are intimately familiar with the concept of least-privilege but as business demands and complexity grow, so too do the administrative challenges of adhering to it in practice. Apparently, breach victims struggle with this much more than other organizations, which should grab our attention. While Requirements 8 (unique IDs) and 10 (logging and monitoring) might prevent some authentication and authorization attacks (though most will still look like legitimate activity), they do add accountability since specific actions against specific assets can be tied to specific agents. This, in turn, greatly aids the response, containment, and recovery process.

Data-capturing malware

Since Table 4 depicts the top techniques to compromise payment cards, it should be no surprise that RAM scrapers, packet sniffers, and keyloggers/spyware—all of which are designed to capture data—made the list. RAM scrapers, which have come into vogue in the last few years, are designed to capture data from volatile memory (RAM) within a system. Packet sniffers (aka network sniffer or packet analyzer) monitor and capture data traversing a network and are long-time favorites of cybercriminals. Last, keyloggers and spyware specialize in monitoring and logging the actions of a system user. Keyloggers are typically used to collect usernames and passwords as part of a larger attack scenario.

At first glance, Requirement 4 (encrypted transmissions) may appear to be a widely-implemented and obvious countermeasure against data-capturing malware. However, the "across public networks" is an important distinction; the malware described in the preceding paragraph operate on systems inside the network. Internal networks are less likely to be encrypted (though it is becoming more common) and packet sniffers take advantage of this fact. Even for organizations that do encrypt all traffic internally, keyloggers, spyware, and RAM scrapers exploit soft spots in the armor by enabling criminals to capture data processed within systems. This reminds us of the cat and mouse game between attackers and defenders and that security practices must evolve alongside the threats against them.

This gives rise to the importance of controls within Requirement 11 (Regular testing) such as file integrity monitoring. These applications observe system settings and monitor specific system and applications files. As is the case with IDS/IPS, tuning requires some effort and frequent false positives cause many to dumb down or ignore them altogether. In the event that malware evades the aforementioned defenses to infect systems, strict egress filtering (specified under Requirement 1) can contain it and Requirement 10 (logging and monitoring) stands a chance of detecting attempts to retrieve (i.e., via a backdoor) or send data out of the network.

Abuse of system access/privileges

The final threat action in this section differs in one key respect from those we have covered to this point. All others in the top ten are almost always perpetrated by external threat agents. However, in order to abuse privileges, one must be granted privileges, and that is only done for trusted parties like insiders and business partners. The category of Misuse contains a diverse collection of threat actions that range from "minor" policy violations to outright malicious behavior. This particular one refers to the deliberate abuse of IT system access or privileges. A system administrator gone rogue is a classic example. This is very difficult to prevent as the nature of this threat action is such that it is sufficient to accomplish the goal in and of itself. If one has privileged access already, one does not need methods of elevating privileges or circumventing controls.

From these results, it cannot be said that the DSS fails to address the most prevalent threats to cardholder data. None of the top threat actions listed above falls outside the scope of its 12 requirements.

A good formula for controlling insider abuse is discretion-direction-restriction-supervision. Deterring the abuse of privileges starts with not having known criminals and other shady characters around that will abuse them (showing *discretion*). Those that make the cut must know what is expected of them and what is forbidden (give them *direction*). As such, Requirement 12 (security policies) provides the foundation for controlling malicious insiders. The next line of defense is to limit the number of users who have high-level privileges and restrict whatever privileges are granted them to the minimum required to perform their duties (*restriction*). Strict adherence to the concept of need-to-know under Requirement 7 (a rare thing) should help accomplish this. If, in spite of all this, an insider engages in inappropriate activity (they will), the combination of Requirements 8 (unique IDs) and 10 (logging and monitoring) will help detect it and tie it back to those responsible. Hiring trustworthy people is good, but so is ongoing verification of their trustworthiness—"Trust but verify."

In summary

From these results, it cannot be said that the PCI DSS fails to address the most prevalent threats to cardholder data. None of the top threat actions listed above falls outside the scope of its 12 requirements. For most of them, in fact, multiple layers of relevant controls exist across the standard. There is also not a great deal of excess or waste (at least at a high level) within the standard; with the exception of each of 3 (stored data)[11], 4 (encrypted transmissions), and 9 (physical access), each of the 12 requirements was mentioned in our discussion above. On the ominous side, the requirements exhibiting the worst assessment scores (10, 11) are also those most broadly applicable to the threat actions shown in Table 4. It should not be terribly surprising, then, that organizations suffering known data breaches were not highly compliant with the PCI DSS. As Figure 5 clearly shows, breach victims were less compliant across the board with the exception of Requirement 4 (encrypted transmissions). Does this mean the PCI DSS is a perfect standard and a guarantee against payment card losses? Of course not; but this does offer encouragement for organizations struggling to meet and/or maintain compliance that their efforts are not for naught.

11 Though we didn't mention Requirement 3 specifically, it should be noted that it is generally applicable to all the threat actions listed in Table 3. If data is not present, it cannot be compromised.

Conclusions and Recommendations

We hope the material presented in this report gives you a better picture of the state of compliance in the payment card industry. Even better if it helps you determine where your organization fits within that picture and proves useful in reaching your goals. At the end of the day, there is no magic formula that will guarantee success in all your future PCI DSS assessments and endeavors. There are, however, certain practices shared by highly successful organizations when it comes to attaining and maintaining compliance. Many of these come down to basic common sense, but, for whatever reason, often get lost among the everyday concerns and routine of running a business. We have enumerated these practices below.

Don't drive a wedge between compliance and security.	Whatever your stance on the "compliance vs. security" debate, hopefully we can all agree that intentionally keeping them apart doesn't make sense from either a compliance or a security perspective. Why force a false dichotomy between two concepts that should, in theory, be in alignment? After all, they both have the goal of protecting data. Sure, maybe you'll need to do some things for compliance that you wouldn't do for security (based on risk assessment or tolerance) or vice versa, but it's hardly an either-or situation across the board. The overall direction of managing compliance should be in line with the security strategy. Is your compliance management team the same as your security management team? If not, is there a concerted effort to collaborate when and where possible or do both sides govern their own private islands with no trade routes between them? If the latter situation is truer of your organization, perhaps you should ask why and whether it's best for it to remain that way.
Build security into your processes, not onto them.	By now, most organizations have learned the hard way that security applied as a band-aid is both costly and ineffective. What many do not seem to realize is that such an approach impacts compliance as well. While it is difficult to analyze through raw data, experience tells us that organizations that build security into their core processes generally spend less and achieve more when it comes to validating compliance. This probably has something to do with the previous recommendation; if an organization truly and consistently strives to be secure then it should not require a giant leap to be compliant.
Treat compliance as a continuous process, not an event.	The difference between process-driven and event-driven compliance programs is relatively simple to identify for an experienced assessor. Organizations that enjoy continued success in achieving and maintaining PCI compliance are those that have integrated the DSS activities into their daily operations. They work on an ongoing basis to review and meet requirements along with other external or internal compliance initiatives. They document security processes, maintain records, meet periodic internal checkpoints, and can quickly provide evidence of adhering to the designated controls. They create a roadmap for the next few years and consult it regularly to understand what challenges are on the horizon, what changes are necessary, and how best to integrate efforts into the short- and long-term strategy for protecting payment infrastructure and data. Put another way, achieving and maintaining PCI Compliance should not be considered an annual project but a daily process.

When preparing to validate, don't procrastinate.	This recommendation is related to and flows from the former. When organizations treat compliance like an event with a looming deadline, a great deal of rushing to and fro ensues. A tremendous amount of energy and resources will be spent in frantic preparation for the imminent arrival of the QSA. A year's backlog of changes are hastily made (those will come back to haunt you), the dust is blown off old documents (if they even exist and can be found), while duct tape and a good spit shine make everything else appear sturdy and tidy (move along, nothing to see here). This is the behavior of an organization that is almost certainly doomed to fail its assessment. It is also the behavior of an organization doomed to security failures, since—even if it manages to pass the assessment—will soon revert to normal behavior. The duct tape will wear off, the shiny stuff will tarnish, and the true state of things will be exposed. The organization will pay for compliance failures in wasted time and effort and for security failures in losses and penalties. The worst part, though, is that cardholders will pay for their behavior as well.
Avoid a failure to communicate.	As it relates to compliance initiatives, organizations often do not recognize the importance of communication. Those preparing for or undergoing an assessment should proactively communicate with all parties involved to avoid hindering the process through a lack of coordination. While this is certainly important among internal parties, it is especially critical when it comes to working with external parties. Open lines of communication to vendors that manage systems within the scope of the assessment will help make sure necessary information is available and ready for the QSA. If organizational changes make it impossible to meet a compliance deadline, the acquiring bank should be aware of the delay and the circumstances surrounding it. Keeping everyone in the loop is a rather obvious recommendation but it's also one that will help keep everyone happy—and a little more happiness to go around never hurts.
Understand how your decisions affect compliance.	A leopard can't change its spots and an organization can't (easily) change certain things about itself. Some of these unchangeables impact compliance. For example, the PCI DSS requires service providers to do some things differently than merchants, and there is nothing they can do to change that. This does not mean, of course, that organizations will never enact significant changes. It happens all the time, and it's important to understand how these decisions will affect the organization's ability to attain and/or maintain compliance. For instance, choosing to stick with legacy systems can make it more difficult to pass certain test procedures around access controls, audit logs, and encryption. Electing to outsource a function may result in it being more difficult to obtain information required for validation from the provider. Ultimately, such decisions will be made based on business needs and other related factors, but organizations do well to consider and prepare for the potential ramifications to the compliance process.

Keep it small and simple.	Continuing with the theme of decision-making, organizations often cannot fundamentally alter their IT environment. Business processes may necessitate certain infrastructure, applications, functions, and configurations. At times, however, a choice can be made between a more complex option and a simpler one. All else being equal, the simpler alternative will almost always be easier to manage in terms of both security and compliance. Based on our experience, ease of management correlates highly with successful management. Therefore, embrace this modified KISS rule whenever possible, and keep it small and simple.
Discover and track your data.	Nothing unnecessarily increases the scope of your PCI assessment like losing track of cardholder data. Make no doubt about it—those bits and bytes have an uncanny ability to multiply and migrate, and your IT infrastructure provides ample roosting places. Understanding data flows and stores is essential to establishing the scope of assessment. A poor understanding of this usually results in an overly large scope, which, in turn, usually results in more expense and difficulty. To overcome this, tight control over data is essential. This is a continuous process that begins with discovery. Thankfully, there are in-house tools and third-party services available that can help with this. We find they typically pay for themselves in the long run due to the difficulties and dangers of data run amuck.
Prioritize your approach to compliance.	On the road to "security," organizations may set different destinations based on their individual risk tolerance. The road to "compliance" diverges from "security" in this regard (though they should be headed in the same general direction); the PCI DSS sets the destination (or at least a waypoint) for those governed by it. This does not mean that organizations must take the same route to get there or that all routes are equal. There is an optimal route and there are many sub-optimal routes. The PCI Security Standards Council recognizes this and published the Prioritized Approach as a guide on the road to compliance. Reliable data on which threats are most pertinent to the payment card industry can help adapt the route as well. Used properly, these resources will contribute to a safe, profitable, and successful journey.
Check yourself before you wreck yourself.	The phrase may smack of trite song lyrics, but it's sage advice for security and compliance programs. Take the idea that everything will be alright as long as all the i's are dotted and t's are crossed and toss it out the window. It's dangerous thinking. Healthy thinking understands that underneath it all, things are rarely what they seem. No organization is perfect. No security or compliance program is perfect. There are things right now that need to be acted upon and remediated but you will never know about them unless you check. If the dismal compliance scores in Requirement 10 and 11 aren't enough to convince you of this, perhaps the findings we share about breach victims in this report (and in the DBIR) will. An organization that does not check cannot act. An organization that does not act cannot be successful in the long term. Don't be one of them.

ABOUT VERIZON PCI SERVICES

Avoiding PCI compliance efforts, or not fully understanding how PCI applies to you, can be costly in more ways than you might think. You may face penalties and fines, litigation, and the costs of re-issuing compromised cards. And if a data breach occurs, you could lose money—and your well-earned reputation.

When you need assistance with implementing solutions and compensating controls to comply with the PCI DSS requirements, we can provide the right resources. The Verizon Business PCI Team performs hundreds of assessments each year and works with both local and global Fortune 500 companies. It is composed of QSAs and PA-QSAs in six global regions that support over 20 languages.

This dedicated team focuses on PCI DSS and PA-DSS Assessments as well as PCI readiness, advisory, and remediation services. In addition to professional services, Verizon Business assists PCI customers through a variety of product platforms including our Merchant Compliance Program (MCP), Online Compliance Program (OCP), and Partner Security Program (PSP).