PCI Security Standards Council®

Date: April 2018

Author: Cloud Special Interest Group
PCI Security Standards Council

Information Supplement:

PCI SSC Cloud Computing Guidelines

Document Changes

Date	Version	Description
February 2013	2.0	Initial publication of PCI DSS v2.0 Cloud Computing Guidelines, produced by 2013 Cloud SIG.
April 2018	3.0	Updated PCI SSC Guidelines for Secure Cloud Computing, produced by 2017 Cloud SIG. Changes include: • Restructure of the document for better flow (e.g., consolidation of Sections 6.3 and 6.4, and moving Section 6.5 to Appendix E). • Updated guidance on roles and responsibilities, scoping cloud environments, and PCI DSS compliance challenges. • Expanded guidance on incident response and forensic investigation. • New guidance on vulnerability management, as well as additional technical security considerations on topics such as Software Defined Networks (SDN), Containers, Fog Computing and Internet of Things (IoT). • Standardized terminology throughout the document. • Updated references to PCI SSC and external resources. • Minor grammatical updates.

Table of Contents

1 Introduction

Cloud computing is a form of distributed computing[1] that enables access to a scalable and elastic pool of shareable resources with on-demand provisioning and administration. There are a number of factors to be considered when planning to use cloud services, and organizations need to clearly understand their needs before they can determine whether and how they will be met by a particular solution or provider.

This guidance is intended for organizations using, or thinking of using, providing or assessing cloud technologies. It provides guidance on the use of cloud technologies and considerations for maintaining security controls in cloud environments. This document is intended to provide an initial point of discussion for Providers and Customers and is not intended to address specific technical configurations or compliant scenarios.

The guidance in this document is structured as follows:

- **Cloud Overview –** Describes the cloud deployment models and service models discussed throughout this document.
- **Cloud Provider/Customer Relationships –** Discusses how roles and responsibilities may differ across different cloud service and cloud deployment models.
- **PCI DSS Considerations –** Provides guidance and examples to help determine responsibilities for individual PCI DSS requirements, and includes segmentation and scoping considerations.
- **PCI DSS Compliance Challenges –** Describes some of the challenges associated with validating PCI DSS compliance in a cloud environment.
- **Security Considerations –** Explores a number of business and technical security considerations for the use of cloud technologies.

The following appendices are included to provide additional guidance:

- **Appendix A: PCI DSS Responsibilities for Different Service Models –** Presents additional considerations to help determine PCI DSS responsibilities across different cloud service models.
- **Appendix B: Sample Inventory –** Presents a sample system inventory for cloud computing environments.
- **Appendix C: PCI DSS Responsibility Management Matrix –** Presents a sample matrix for documenting how PCI DSS responsibilities are assigned between Provider and Customer.
- **Appendix D: PCI DSS Implementation Considerations –** Suggests a starting set of questions that may help in determining how PCI DSS requirements can be met in a particular cloud environment.
- **Appendix E: Technical Security Considerations –** Provides guidance on various cloud-based technologies.

[1] Wayne Jansen and Timothy Grance, *NIST Guidelines on Security and Privacy in Public Cloud Computing,* NIST Special Publication 800-144, (Gaithersburg: National Institute of Standards and Technology, December 2011). https://doi.org/10.6028/NIST.SP.800-144.

The information in this document provides supplemental guidance and does not supersede, replace or extend the requirements in any PCI SSC standard, nor does it endorse the use of any specific technologies, products or services. For the purposes of this document, all references made to PCI DSS are to version 3.2; however, the general principles and practices offered here may be applied beyond the context of PCI DSS.

1.1 Intended Audience

The information in this document is intended for merchants, service providers, assessors and other entities looking for guidance on how the use of cloud computing may affect PCI DSS compliance. For example:

- **Merchants –** Guidance on the security and PCI DSS considerations that are applicable to cloud environments and may be useful to merchants managing their own cloud infrastructure as well as those looking to engage with a third party. Guidance for working with third-party Providers and PCI DSS compliance challenges may also be useful.
- **Service Providers –** Guidance on the security and PCI DSS considerations that may provide useful information for Providers to assist their understanding of the PCI DSS requirements, and may also help Providers to better understand Customers' PCI DSS needs. The guidance on Provider/Customer relationships and PCI DSS compliance challenges herein may also be useful for Providers.
- **Assessors –** Guidance on the security and PCI DSS considerations that may help assessors to understand what they need to know about an environment in order to be able to determine whether a PCI DSS requirement has been met.

1.2 Terminology

In addition to terms defined in the PCI DSS Glossary of Terms, Abbreviations and Acronyms, the following terms are used throughout this document:

- **Cloud Service Provider ("Provider"):** It is the entity providing the cloud service. It acquires and manages the infrastructure required for providing the services, runs the cloud software that provides the services and delivers the cloud services through network access.[2]
- **Cloud Service Customer ("Customer"):** The entity subscribing to a service provided by a Provider. May include merchants, service providers, payment processors and other entities utilizing cloud services.
- **Cloud Service User:** Person, or entity acting on his or her behalf, associated with a Customer that uses cloud services. ***Note:*** *Examples of such entities include devices and applications.*[3]
- **Multi-tenancy:** Allocation of physical or virtual resources such that multiple cloud tenants and their computations and data are isolated from and inaccessible to one another.[4]
- **Cloud Tenant:** One or more Customers sharing access to a set of physical and virtual resources.

[2] Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger, and Dawn Leaf, *Cloud Computing Reference Architecture*, NIST Special Publication 500-92 (Gaithersburg: National Institute of Standards and Technology, September 2011). https://dx.doi.org/10.6028/NIST.SP.500-292.

[3] Joint Technical Committee ISO/IEC JTC 1, *Information technology – Cloud computing – Overview and vocabulary ISO/IEC 17788*. http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html.

[4] Ibid

1.3 Summary of Recommendations

In addition to the business and risk considerations, the implementation of security controls in a cloud environment requires specialized technical knowledge and skills. It is therefore crucial that, prior to migrating payment card operations into a cloud environment, the Customer engage its technical, legal, due diligence, information security and compliance teams to work together to define its needs and evaluate potential cloud service offerings against those needs.

Ensuring that cloud services are designed, maintained and used securely is a shared responsibility between the Provider and the Customer. It is important to note that all cloud services are not created equal. Clear policies and procedures should be agreed upon between the Customer and the Provider for all security requirements. Responsibilities for operation, management and reporting should be clearly defined and understood for each requirement and acknowledged, in writing, in contractual agreements.

Regarding third-party or public clouds, Customers should consider that while they can outsource the day-to-day operational management of the data environment, they retain responsibility for the data they put in the cloud. Customers are encouraged to "shop around" until they find a Provider who can provide the level of security and assurance they require.

The following steps should be followed by any organization looking to migrate to or evaluate cloud services:

- UNDERSTAND your risk and security requirements first.
- CHOOSE a deployment model that aligns with your and your industry's security and risk requirements.
- EVALUATE different service options.
- KNOW what you want from your Provider.
- COMPARE Providers and service offerings.
- ASK questions of the Provider and verify the responses; for example:
  - What does each service consist of exactly, and how is the service delivered?
  - What does the service provide with respect to security maintenance, PCI DSS compliance, segmentation and assurance, and for what is the Customer responsible?
  - How will the Provider provide ongoing evidence that security controls continue to be in place and are kept up to date?
  - What will the Provider commit to in writing?
  - Are other parties involved in the service delivery, security or support?
- DOCUMENT everything with your Provider in written agreements - for example, Service Level Agreements (SLAs)/Terms of Service contracts, etc.
- REQUEST written assurances that security controls will be in place, and periodic verification (e.g., compliance reports) that controls continue to be maintained.
- REVIEW the service and written agreements periodically to identify whether anything has changed.

Providers are encouraged to work with their Customers to understand their security and compliance needs. Both parties should be willing to maintain open communication and monitoring to avoid any misunderstandings or gaps in security responsibilities.

If account data is stored, processed or transmitted in a cloud environment, PCI DSS will apply to that environment, and compliance will typically involve validation of both the Provider’s environment and the Customer’s usage of that environment.

Customers have final responsibility for security of their cardholder data.

Even though a Provider may claim to be PCI DSS compliant, the Customer should confirm that all the consumed services and locations were included in the Provider PCI DSS compliance validation, and that the services are used in a compliant manner.

Moreover, the allocation of responsibility between Customer and Provider for managing security controls does not exempt a Customer from the responsibility of ensuring that their cardholder data (CHD) is properly secured according to applicable PCI DSS requirements. Customers should define what PCI DSS requirements are shared among the Customer, Provider and any intermediaries (e.g., a payment gateway) and confirm their compliance.

2 Cloud Overview

Cloud computing provides a model for enabling on-demand network access to a shared pool of computing resources (for example: networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or Provider interaction.[5]

Cloud computing can be used to provide Customers with access to the latest technologies without a costly investment in hardware and software. Due to the economies of scale associated with the delivery of cloud services, Providers can often provide access to a greater range of technologies and security resources than that to which the Customer might otherwise have access. Organizations without a depth of technically skilled personnel may also wish to leverage the skills and knowledge provided by Provider personnel to securely manage their cloud operations.

Cloud computing therefore holds significant potential to help organizations reduce IT complexity and costs, while increasing agility. Cloud computing is also seen as a means to accommodate business requirements for high availability and redundancy, including business continuity and disaster recovery.

2.1 Cloud Deployment Models

Deployment models are defined to distinguish between different models of ownership and distribution of the resources used to deliver cloud services. Cloud environments may be deployed over a private infrastructure, public infrastructure or a combination of both. The most common deployment models include:[6]

- **Public cloud:** Cloud deployment model where cloud services are potentially available to any Customer and resources are controlled by the Provider. A public cloud may be owned, managed and operated by a business, academic or government organization, or some combination of them. It exists on the premises of the Provider. Actual availability for specific Customers may be subject to jurisdictional regulations. Public clouds have very broad boundaries, where Customer access to public cloud services has few, if any, restrictions.
- **Private cloud:** Cloud deployment model where cloud services are used exclusively by a single Customer and resources are controlled by that Customer. A private cloud may be owned, managed and operated by the organization itself or a third party and may exist on premises or off premises. Private clouds seek to set a narrowly controlled boundary around the private cloud based on limiting the customers to a single organization.
- **Community cloud:** Cloud deployment model where cloud services exclusively support and are shared by a specific collection of Customers who have shared requirements and a relationship with one another, and where resources are controlled by at least one member of this collection. A community cloud may be owned, managed and operated by one or more of the organizations in the community, a third party or some combination of them, and it may exist on or off premises. Community clouds limit participation to a group of Customers who have a shared set of objectives, in contrast to the openness of public clouds,

[5] Peter Mell and Timothy Grance, *The NIST Definition of Cloud Computing Special Publication 800-145,* NIST Special Publication 800-145 (Gaithersburg: National Institute of Standards and Technology, December 2011). https://doi.org/10.6028/NIST.SP.800-145.

[6] Joint Technical Committee ISO/IEC JTC 1.

while community clouds have broader participation than private clouds. These shared concerns include, but are not limited to, mission, information security requirements, policy and compliance considerations.

- **Hybrid cloud –** The cloud infrastructure is a composition of two or more clouds (private, community or public) that remain unique entities but are bound together by technology to enable portability. Hybrid clouds are often used for redundancy or load-balancing purposes—for example, applications within a private cloud could be configured to utilize computing resources from a public cloud as needed during peak capacity times (sometimes called "cloud-bursting").

2.2 Cloud Capabilities Types and Cloud Service Categories

Cloud service categories ("<Something>-as-a-Service") identify different control options for the Customer and Provider. For example, SaaS Customers simply use the applications and services provided by the Provider, where IaaS Customers maintain control of their own environments hosted on the Provider's underlying infrastructure. The National Institute of Standards and Technology (NIST) defines three types of cloud service categories.[7]

Software as a Service (SaaS) – Capability for Customers to use the Provider's applications running on a cloud infrastructure. The applications are accessible from various devices through either a thin client interface, such as a web browser, or a program interface.

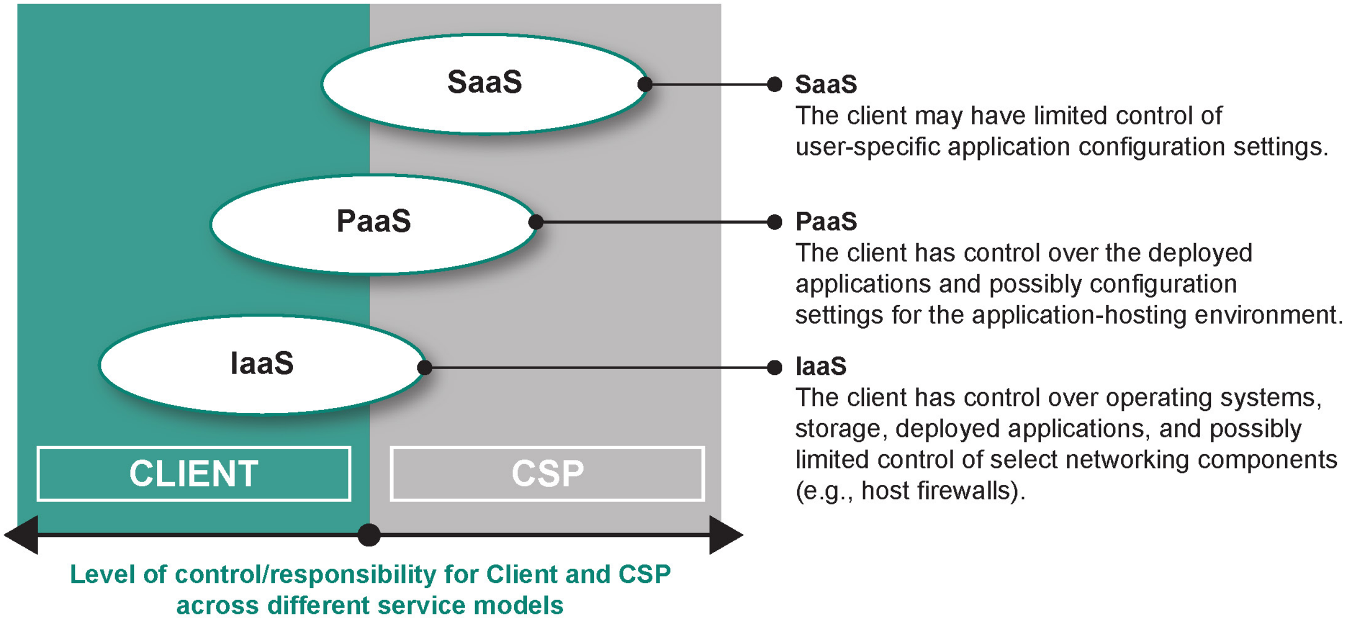
Platform as a Service (PaaS) – Capability for Customers to deploy their applications (created or acquired) onto the cloud infrastructure, using programming languages, libraries, services and tools supported by the Provider.

Infrastructure as a Service (IaaS) – Capability for Customers to utilize the Provider's processing, storage, networks and other fundamental computing resources to deploy and run operating systems, applications and other software on a cloud infrastructure.

There are other standards and frameworks that define different vocabulary and reference architecture (e.g., ISO/IEC 17788:2014) which are mostly comparable with the terms and categories defined by NIST. For consistency, this guidance document uses NIST terminology (i.e., SaaS, PaaS and IaaS) to describe the service categories of Provider services.

The main differences between cloud service categories relate to how control is shared between Customer and Provider, which in turn affects the level of responsibility for both parties. It should be noted that, other than in a self-managed private cloud scenario, the Customer rarely has any control over hardware, and it is the degree to which virtual components, applications and software are managed by the different parties that differentiates the cloud service categories. As a general rule, SaaS provides Customers with the least amount of control, whereas IaaS offers the most control for the Customer. Figure 1 shows how control is typically shared between the Provider and Customer across different service categories.

[7] Wayne Jansen and Timothy Grance, *NIST Guidelines on Security and Privacy in Public Cloud Computing,* NIST Special Publication 800-144, (Gaithersburg: National Institute of Standards and Technology, December 2011) . https://doi.org/10.6028/NIST.SP.800-144.


SaaS
PaaS
IaaS
CLIENT
CSP
Level of control/responsibility for Client and CSP across different service models

SaaS
The client may have limited control of user-specific application configuration settings.

PaaS
The client has control over the deployed applications and possibly configuration settings for the application-hosting environment.

IaaS
The client has control over operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).

Figure 1: Level of control/responsibility for Customer and Provider across different service categories

The level of security responsibility across the cloud service categories generally migrates towards the Customer as the Customer moves from a SaaS category (least Customer responsibility) to an IaaS category (most Customer responsibility). The greatest level of responsibility for the Provider to maintain security and operational controls is present in the SaaS service category.

While Customers may be attracted to the SaaS and PaaS categories due to the resource savings and reduced responsibility for administering the cloud environment, they should be aware that these categories also correspond to a greater loss of control of the environment housing their sensitive data.

Contractual agreements and ongoing due diligence become especially critical where control is outsourced, to ensure that the required security measures are being met and maintained by the Provider for the duration of the agreement.

It is important to note that these descriptions for cloud deployment models and cloud service categories, although now standardized and widely accepted by the industry, may not be universally followed by Providers or reflect actual cloud environments. For example, a Provider might be selling a "private cloud" service that does not meet the intent of "private" as it is described above. Similarly, the details of what is and what is not included in a particular service will probably vary between Providers, even if they identify their services by the same term (IaaS, PaaS, or SaaS).

It is expected that as the industry and the Customer requirements evolve as cloud offerings mature, there will be additional cloud service categories besides those listed above. However, this guidance document focuses on the prevailing three cloud service categories: SaaS, PaaS and IaaS.

3 Cloud Provider/Customer Relationships

3.1 Understanding Roles and Responsibilities

The lines of accountability and responsibility will be different for each cloud service category and deployment model and will be governed by the signed contractual agreements. Customers should not make assumptions about any service—clear policies and procedures should be agreed upon between Customer and Provider for all security requirements, and clear responsibilities for operation, management and reporting need to be defined for each requirement.

The PCI Security Standards Council has published the Information Supplement *Third-Party Security Assurance,* which provides further guidance on implementing third-party assurance program.[8]

3.2 Roles and Responsibilities for Different Cloud Deployment Models

The entity performing the role of Provider will vary according to the type of deployment model. For example, the Provider role may be assigned entirely to an external third party (as in a public cloud), or the role may be undertaken by an internal department or business function (as in an on-premises private cloud). Similarly, the role of Provider may be assigned to more than one entity in a community or hybrid cloud scenario.

To understand how responsibilities are assigned in a particular deployment model, consider the following:

- **Public cloud –** The Provider is a third party that is organizationally separate from its Customers. The cloud is deployed within a Provider's environment and responsibility is delineated according to the particular cloud service category, as defined by the Provider.
- **Private cloud –** Where a private cloud is managed on-premises, the Provider role may be undertaken within the Customer. For example, the IT department could take on the role of Provider with various operational departments as its Customers. In this scenario, the Customer retains full control of its environment and responsibility for its security and compliance.

  Dedicated, private clouds may also be provisioned off-premises by a third-party Provider. In this case, the delineation of responsibility will also depend on the particular cloud service category, as described in Section 3.3, "Responsibilities for Different Cloud Service Categories."
- **Hybrid cloud –** The Provider role may be assigned to both internal and third-party entities for different elements of the overall cloud infrastructure. Responsibility will be assigned based on the combination of deployment models and cloud service categories implemented.
- **Community cloud –** The Provider could be one of the Customers within the community or a separate third party. The delineation of responsibility follows the particular cloud service category implemented. The responsibility for implementation, operation and management of security controls will be shared differently within each of the cloud service categories and needs to be clearly understood by both the Customer and Provider. The Customer also needs to understand the level of oversight or visibility it will have into security functions that are outside its control. If these security responsibilities are not properly

[8] Third-Party Security Assurance and Shared Responsibilities Special Interest Groups and PCI Security Standards Council, *Third-Party Security Assurance* (PCI SSC, March 2016), https://www.pcisecuritystandards.org/documents/ThirdPartySecurityAssurance_March2016_FINAL.pdf.

assigned, communicated and understood, insecure configurations or vulnerabilities could go unnoticed and unaddressed, resulting in potential exploit and data loss or other compromise.

3.3 Responsibilities for Different Cloud Service Categories

In all deployment models, and particularly in public cloud environments, it is important for all parties to understand the specific elements of the cloud service category used and its associated risks. Any cloud deployment model that is not fully self-managed is by nature a shared responsibility model, where a portion of responsibility for the cloud service falls under the realm of the Provider (see Section 3.4, “Nested Service Provider Relationships,” for more information), and a portion of responsibility also falls to each Customer. The level of responsibility that falls to the Provider or the Customer is determined by the cloud service category being utilized—for example, IaaS, PaaS or SaaS. Clear delineation of responsibilities should be established as a prerequisite to any cloud service implementation to provide a baseline for the cloud operation.

Table 1 illustrates how control of the different technical layers is often shared across different cloud service categories. For illustration purposes, different layers of the cloud stack are described as follows:

Layer	Description
Application Program Interface (API) or Graphical User Interface (GUI)	The interface by which cloud service users interact with the application. The current most common API is RESTful HTTP or HTTPS. The current most common GUI is an HTTP- or HTTPS-based website.
Application	The actual application being used by one or more cloud service users
Solution Stack or Technology Stack	This is the programming language used to build and deploy applications. Examples are .NET, Python, Ruby, Perl, etc.
Operating Systems (OS)	In a virtualized environment, the OS runs within each VM. Alternatively, if there is no underlying hypervisor present, the operating system runs directly on the storage hardware.
Virtual Machine (VM)[9]	A virtual container executed on a hypervisor on a host. A set of system isolation technologies that provide various degree of security isolation with the host machine’s OS kernel
Containers	Virtualization technique that allows execution of multiple isolated user-space instances while sharing the same underlying OS kernel
Virtual Network Infrastructure	For communications within and between virtual machines
Hypervisor	When virtualization is used to manage resources, the hypervisor is responsible for allocating resources to each virtual machine. It may also be leveraged for implementing security.

[9] Providers will often distinguish between Paravirtual (PV) and Hardware Virtual Machine (HVM) virtualization. The Customer should be familiar with the difference and consider its impact on PCI DSS or process isolation concerns.

Layer	Description
Processing and Memory	The physical hardware that supplies CPU time and physical memory
Data Storage	The physical hardware used for file storage
Network	This can be a physical or virtual network. It is responsible for carrying communications between systems and possibly the internet.
Physical Facility	The actual physical building where the cloud systems are located

Appendix B *illustrates a sample inventory for cloud computing systems as guidance for the ways in which Providers and Customers can document the different layers of the cloud environment.*

Table 1: Example of how control may be assigned between Provider and Customer across different cloud service categories

■	*Customer*
▤	*Provider*
□	*Shared*

Responsibility	Service Models		
	IaaS	PaaS	SaaS
Security Governance, Risk and Compliance (GRC)	Customer	Customer	Customer
Data Security	Customer	Customer	Customer
Application Security	Customer	Customer	Shared
Platform Security	Customer	Shared	Provider
Infrastructure Security	Shared	Provider	Provider
Physical Security	Provider	Provider	Provider

Note: *This table provides an example of the ways in which responsibilities might be assigned according to common descriptions of the different cloud service categories. However, it is important to note that the technology layers and their corresponding lines of responsibility may be different for each Provider, even if they use the same terminology to describe their services, and the individual service offerings may or may not align with the responsibility assignments indicated above.*

Some Providers offer multiple options for their services—for example, a Provider may have one IaaS offering that includes a Customer-controlled hypervisor and a separate IaaS offering with no Customer access to the hypervisor. It is imperative that Customers and Providers clearly document and understand where the boundaries are in their particular relationships, rather than assuming that any particular responsibility model applies to them.

Even where a Customer does not have control over a particular layer, it may still have some responsibility for the configurations or settings that the Provider maintains on its behalf. For example, a Customer may need to define firewall rules and review firewall rule-sets for those firewalls applicable to the protection of its environment, even though the Provider actually configures and manages the firewalls. Similarly, Customers may be responsible for approving and reviewing user access permissions to its data resources, while the Provider configures the access according to Customer needs.

The allocation of responsibility for managing security controls does not exempt a Customer from the responsibility of ensuring that its cardholder data is properly secured.

3.4 Nested Service Provider Relationships

Nested service provider relationships are common in cloud scenarios, as Providers sometimes rely on internal departments or other third-party companies to deliver aspects of their services. For example, some Providers use third-party storage providers as part of their cloud service offering, and some might partner with other Providers for redundancy or fail-over as part of their cloud-delivery strategy.

An example of nested service provider relationships is illustrated in Figure 2 below:

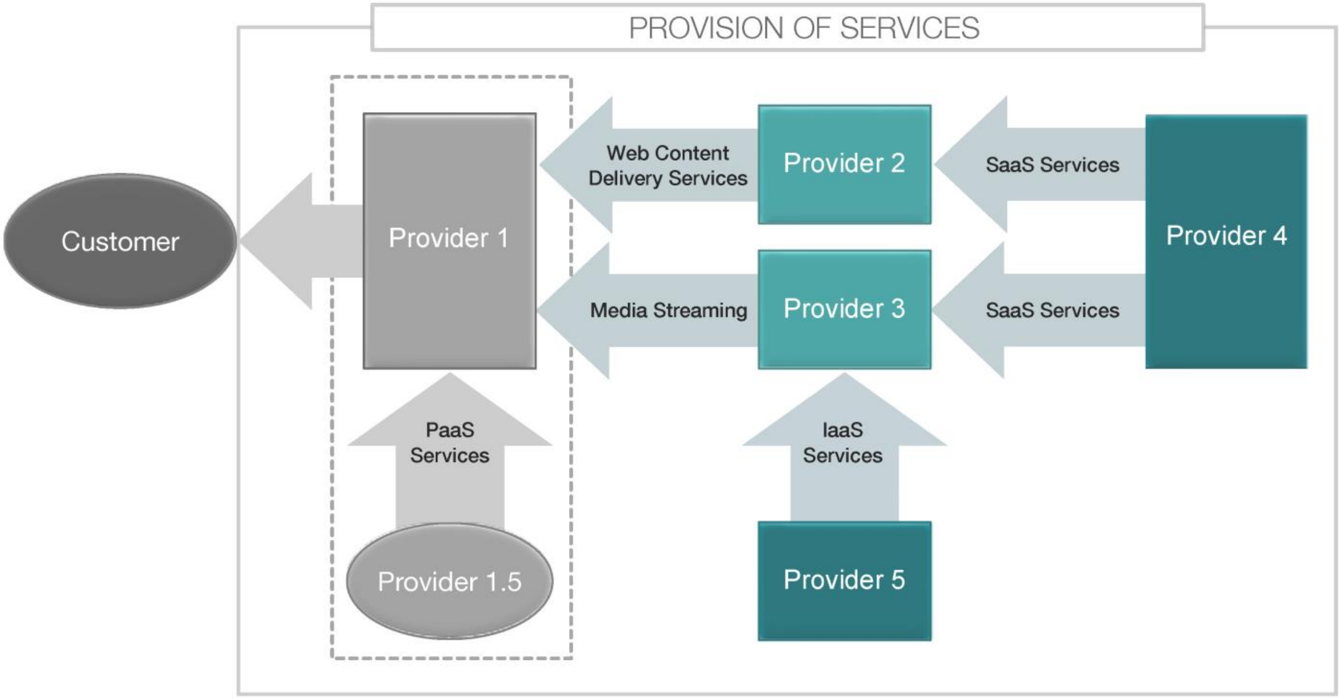


Figure 2: Example of nested service provider relationships

- Provider 1 relies on Provider 2 for web content delivery services and Provider 3 for media streaming, as well as Provider 1.5 (internal/private cloud) for PaaS.
- Provider 4 provides SaaS service to both Providers 2 and 3.
- Provider 5 provides IaaS service to Provider 3.

There may be multiple layers or levels of Provider dependency, which can affect the security of the cardholder environment. Identifying all third-party relationships that the Provider has in place is important in order to understand the potential ramifications for a Customer's environment. The existence of multiple nested relationships—for example, where there is a chain of vendors and other Providers required for delivery of a cloud service—will also add complexity to both the Provider's and the Customer's PCI DSS assessment process.

Where the Customer has a direct contractual relationship with all nested Providers, the Customer will need to understand the impact each Provider has on its CDE, and how PCI DSS responsibilities are managed for each service. Where the Customer does not have contractual relationships with all nested Providers, the Customer would rely on the primary Provider (that is, the Provider with which the Customer has a direct relationship, and which, in turn, manages relationships with the nested Providers) to manage the relationships and PCI DSS responsibilities for all Providers involved in delivery of the service. In both cases, it can be helpful to communicate these relationships to the Customer (consider using the optional PCI DSS Responsibility Matrix), so that it can understand PCI DSS compliance considerations for all components of the service.

Moreover, Customers may also leverage this nested Provider relationship information for pre-engagement due diligence (Requirement 12.8) as well as the following processes:

- Risk profiling and management
- Business continuity and disaster recovery planning
- Threat monitoring
- Supply-chain management

4 PCI DSS Considerations

4.1 Understanding PCI DSS Responsibilities

The responsibilities delineated between the Customer and the Provider for managing PCI DSS controls are influenced by a number of variables, including but not limited to:

- The purpose for which the Customer is using the cloud service
- The scope of PCI DSS requirements that the Customer is outsourcing to the Provider
- The services and system components that the Provider has validated within its own operations
- The service option that the Customer has selected to engage the Provider (e.g., IaaS, PaaS or SaaS)
- The scope of any additional services the Provider is providing to proactively manage the Customer's compliance (for example, additional managed security services)

The Customer needs to clearly understand the scope of responsibility that the Provider is accepting for each PCI DSS requirement, and which services and system components are validated for each requirement. For example, PCI DSS Requirements 6.1 and 6.2 address the need for vulnerabilities to be identified and ranked according to risk, and for missing patches to be deployed in a timely manner. If not properly defined, a Customer could assume that the Provider is managing this process for the entire cloud environment, whereas the Provider could be managing vulnerabilities for its underlying infrastructure only, and assuming that the Customer is managing vulnerabilities for operating systems and applications.

4.2 PCI DSS Responsibilities for Different Cloud Service Categories

As a general rule, the more aspects of a Customer's operations that the Provider manages, the more responsibility the Provider has for maintaining PCI DSS controls. However, outsourcing maintenance of controls is not the same as outsourcing responsibility for the data overall. Customers should not make assumptions about any service, and should clearly spell out in contracts, memorandums of understanding or SLAs exactly which party is responsible for securing which system components and processes.

Table 2 provides an example of how responsibilities for PCI DSS requirements may be shared between Customers and Providers across some of the various cloud service categories. There will of course be exceptions and variations across each individual service, and this table is provided as a guideline for Customers and Providers to help plan discussions and negotiations.

Responsibilities have been identified as follows:

- **Customer –** Generally, each Customer will retain responsibility for maintaining and verifying the requirement.
- **Provider –** Generally, the Provider will maintain and verify the requirement for its Customers.
- **Shared –** Generally, responsibility is shared between the Customer and the Provider. This may be due to the requirement applying to elements present in both the Customer environment and the Provider managed environment, or because both parties need to be involved in the management of a particular control.

***Appendix A** includes additional considerations for determining how PCI DSS responsibilities may be assigned for each cloud service category.*

***Appendix C** illustrates a sample PCI DSS Responsibility Matrix, as guidance for how Providers and Customers can document PCI DSS responsibility assignments.*

The concept of shared or joint responsibility can be particularly tricky to deal with. While some services and functions will be relatively straightforward to scope and establish boundaries, many services and functions will overlap if not clearly demarcated at the outset of the service relationship.

Where the Provider maintains responsibility for PCI DSS controls, the Customer is still responsible for monitoring the Provider's ongoing compliance for all applicable requirements. Providers should be able to provide their Customers with ongoing assurance that requirements are being met, and where the Provider is managing requirements on behalf of the Customer, it should have mechanisms in place to provide the Customer with the applicable records to demonstrate that the requires security controls are in place—for example, audit logs showing all access to Customer data.

Customers are still required to validate their compliance in accordance with payment brand programs.

Table 2: Example of PCI DSS responsibility sharing between Customers and Providers

■	*Customer*
▤	*Provider*
□	*Shared*

PCI DSS Requirement	Example Responsibility Assignment for Management of Controls		
	IaaS	**PaaS**	**SaaS**
1: Install and maintain a firewall configuration to protect cardholder data.	Shared	Shared	Provider
2: Do not use vendor-supplied defaults for system passwords and other security parameters.	Shared	Shared	Provider
3: Protect stored cardholder data.	Shared	Shared	Provider
4: Encrypt transmission of cardholder data across open, public networks.	Customer	Shared	Provider
5: Protect all systems against malware and regularly update anti-virus software or programs.	Customer	Shared	Provider
6: Develop and maintain secure systems and applications.	Shared	Shared	Shared
7: Restrict access to cardholder data by business need to know.	Shared	Shared	Shared
8: Identify and authenticate access to system components.	Shared	Shared	Shared
9: Restrict physical access to cardholder data.	Provider	Provider	Provider
10: Track and monitor all access to network resources and cardholder data.	Shared	Shared	Provider
11: Regularly test security systems and processes.	Shared	Shared	Provider
12: Maintain a policy that addresses information security for all personnel.	Shared	Shared	Shared
PCI DSS Appendix A1: Additional PCI DSS Requirements for Shared Hosting Providers	Provider	Provider	Provider

***Note:** The sample responsibilities illustrated in this table do not include consideration for any activities or operations performed outside a hypothetical cloud service offering. This table provides an example of the ways in which PCI DSS responsibilities might be assigned for different cloud service categories. However, each Provider ultimately defines its own service and particular service offerings may or may not be consistent with those illustrated above. Customers and Providers should clearly document their responsibilities as applicable to their particular agreements.*

4.3 Understanding Responsibilities of Security as a Service (SECaaS)

Security as a Service, or SECaaS, is sometimes used to describe the delivery of security services using a SaaS-based cloud service category. While SECaaS solutions are not directly involved in storing, processing or transmitting CHD, it may still be an integral part of the security of the CDE, and therefore it is important to understand for which PCI DSS controls the SECaaS Provider is responsible. As an example, a SaaS-based anti-malware solution may be used to update anti-malware signatures on the Customer's systems via a cloud-delivery model. In this example, the SECaaS offering is delivering a PCI DSS control to the Customer's environment, and the SECaaS functionality will need to be reviewed to verify that it is meeting the applicable requirements.

SECaaS can vary in complexity from a relatively simple cloud-based application subscription model to complex outsourcing of entire security functions such as a Security Operation Center (SOC) or a Network Operation Center (NOC) using cloud-delivery models. The number of applicable PCI DSS requirements that fall under the responsibility of the SECaaS Provider will increase relative to the complexity of the service being provided. Merchants using SECaaS must include SECaaS service providers in the list of PCI service providers, and ensure compliance with, at the very least, PCI DSS Requirements 12.8.1 – 12.8.5 for each SECaaS service provider used.

4.4 Segmentation Considerations

Outside a cloud environment, individual entity environments would normally be physically, organizationally and administratively separate from each other. Customers utilizing a public or otherwise shared cloud must ensure that their environments are adequately isolated from the other cloud tenants.

In addition to enforcing separation between Customer environments, segmentation may also be desired within a Customer's environment to isolate its CDE components from non-CDE components in order to reduce its own PCI DSS scope.

Segmentation on a cloud-computing infrastructure must provide a level of isolation equivalent to that achievable through physical network separation. Mechanisms to ensure appropriate isolation may be required at the network, operating system and application layers; and most importantly, there should be guaranteed isolation of data that is stored (see Section 4.4.3, "Segmentation Technologies," for more information). Cloud tenant environments must be isolated from each other such that they can be considered separately managed entities with no connectivity between them. Providers should test segmentation (Requirement 11.3.4) between all entities within their control at least semiannually and demonstrate results.

Any systems or components shared by the Customers in multi-tenant environments, including the hypervisor and underlying systems, must not provide an access path between environments. Any shared infrastructure used to host an in-scope Customer environment would be in scope for that Customer's PCI DSS assessment.

In a hybrid environment, the responsibility for confirming segmentation is shared by the Provider and Customer. Confirming that segmentation is effectively isolating the CDE supports the Customer by simplifying and minimizing the CDE (rather than, for example, its entire on-premises network being included in its CDE). Moreover, it is the Customer's responsibility to ensure that the link or connection to the Provider, as well as any devices (e.g., physical router on premises, cloud gateway or peering transit networks, etc.) used to facilitate the connection, is secured and managed.

A segmented cloud environment exists when the Provider enforces isolation between Customers in multi-tenant environments. Examples of how segmentation may be provided in shared cloud environments include, but are not limited to:

- Traditional Application Service Provider (ASP) model, where physically separate servers are provided for each Customer's cardholder data environment (CDE)
- Virtualized servers that are individually dedicated to a particular Customer, including any virtualized storage such as Storage Area Networks (SANs), Network Attached Storage (NAS) or virtual database servers
- Environments where Customers run their applications in separate logical partitions using separate database management system images and do not share disk storage or other resources

The PCI DSS assessor must validate the effectiveness of the segmentation to ensure that it provides adequate isolation. If adequate segmentation is provided between the cloud tenants (in a multi-tenant environment), only the Customer environment and the Provider-managed environment and processes would be in scope for a Customer's PCI DSS assessment. However, if adequate segmentation is not in place or cannot be verified, the entire multi-tenant cloud environment would be in scope for all Customers' assessments hosted in that environment. Examples of non-segmented cloud environments include but are not limited to:

- Environments where organizations use the same application image on the same server and are only separated by the access control system of the operating system or the application
- Environments where organizations use different images of an application on the same server and are only separated by the access control system of the operating system or the application
- Environments where organizations' data is stored in the same instance of the database management system's data store

Without adequate segmentation, all cloud tenants of the shared infrastructure, as well as the Provider, would need to be verified as being PCI DSS compliant in order for any one Customer to be assured of the compliance of the environment. This will likely make compliance validation unachievable for the Provider or any of its Customers.

4.4.1 Segmentation Challenges

Segmentation in traditional hosted environments can be applied via separate physical servers and security measures applied using known methods. The difference in a cloud environment is that there are common shared layers (such as hypervisors and virtual infrastructure layers), which can present a single point of entry (or attack) for all systems above or below those shared layers. The security applied to these layers is therefore critical not only to the security of the individual environments they support, but also to ensure that segmentation is enforced between different cloud tenant environments.

Once any layer of the cloud architecture is shared by CDE and non-CDE environments, segmentation becomes increasingly complex. This complexity is not limited to shared hypervisors; all layers of the infrastructure that could provide an entry point to a CDE must be included when verifying segmentation.

In a private cloud environment, one approach that may help reduce the complexity of segmentation efforts could be to locate all CDE virtual components on a dedicated CDE hypervisor, and ensure that all non-CDE virtual components are located on separate hypervisors, adequately segmented from the CDE hypervisor.

The need for adequate segmentation of Customer environments in a public or shared cloud is underscored by the principle that the other cloud tenant environments running on the same shared infrastructure are to be considered untrusted networks. The Customer has no way of confirming whether other cloud tenant environments are securely configured or patched appropriately to protect against attack, or that they are not already compromised or even designed to be malicious. This is particularly relevant where a Provider offers IaaS and PaaS services, as the individual Customers have greater control and management of their environments.

4.4.2 Segmentation Responsibilities

Ultimately, the Provider needs to take ownership of the segmentation between Customers and verify that it is effective and provides adequate isolation between individual Customer environments, between Customer environments and the Provider's own environment, and between client environments and other untrusted environments (such as the internet). Applicable PCI DSS controls for the segmentation functions would also be the Provider's responsibility (for example, firewall rules, audit logging, documentation, reviews, etc.). The Customer is responsible for the proper configuration of any segmentation controls implemented within its own environment (for example, using virtual firewalls to separate in-scope VMs from out-of-scope VMs), and for ensuring that effective isolation is maintained between in-scope and out-of-scope components. The Provider should test and report security controls that isolate networks from each other according to PCI DSS Requirement 11.3.4.

Customers wishing to implement segmentation within their cloud environments also need to consider how the Provider's environment and processes may affect the effectiveness of the segmentation. For example, Provider systems could be providing connectivity between the Customer's own VMs that is not visible to the Customer. Customers should also consider how the Provider manages offline or dormant VMs, and whether in-scope and out-of-scope VMs could potentially be stored together by the Provider without active segmentation controls.

4.4.3 Segmentation Technologies

Traditional network segmentation technologies consist of hardware devices such as firewalls, switches, routers and so forth. These physical components could be used to separate VMs hosted on the same or multiple hypervisors, similar to the manner in which systems could be segmented in a physical network. This would require hypervisors with multiple network interfaces and with configurations that meet PCI DSS requirements for the various types of network hardware. Additionally, virtual counterparts of firewalls, switches and routers now exist and can be incorporated into a virtual environment.

As mentioned above, a key consideration is how secure the common layers (such as hypervisors, container implementations and shared physical components) are, and to what extent they represent a potential attack surface between zones or Customers.

Examples of controls to be considered when evaluating segmentation options include, but are not limited to:

- Firewalls and network segmentation at the infrastructure level
- Firewalls at the hypervisor and VM level
- VLAN tagging or zoning in addition to firewalls
- Software Defined Networks (see Section E.4, "Software Defined Networking," for additional information)
- Intrusion prevention systems at the hypervisor level, VM level or both, to detect and block unwanted traffic
- Data-loss-prevention tools at the hypervisor level, VM level or both
- Controls to prevent out-of-band communications occurring via the underlying infrastructure
- Isolation of shared processes and resources from cloud tenant environments
- Container-based system isolation based on industry standard, vetted technologies
- Segmented data stores for each Customer
- Strong, two-factor authentication
- Separation of duties and administrative oversight
- Continuous logging and monitoring of perimeter traffic, and real-time response

Segmentation controls should be tested annually (for merchants) or semiannually (for service Providers) to confirm the effectiveness of isolation between Customers in a multi-tenant cloud environment. The Information Supplement *Penetration Testing Guidance* provides further guidance on segmentation controls and testing principles.[10]

4.5 Scoping Considerations

Merchants or other organizations looking to store, process or transmit payment card data in a cloud environment should clearly understand the impact that extending their CDE into the cloud will have on their PCI DSS scope. For example, in a private-cloud deployment, an organization could either implement adequate segmentation to isolate in-scope systems from other systems and services, or it could consider its private cloud to be wholly in scope for PCI DSS. In a public cloud, the Customer and Provider will need to work closely together to define and verify scope boundaries, as both parties will have systems and services in scope.

> ***Appendix D** includes Implementation Considerations for PCI DSS Requirements.*

[10] Penetration Test Guidance Special Interest Group and PCI Security Standards Council, *Penetration Testing Guidance,* (PCI SSC, September 2017), https://www.pcisecuritystandards.org/documents/Penetration-Testing-Guidance-v1_1.pdf.

Recommendations for minimizing and simplifying PCI DSS scope in a cloud environment include:

- Do not store, process or transmit payment card data in the cloud. This is the most effective way to reduce the scope of PCI DSS in a cloud environment.
- Usage of PCI-listed P2PE solutions may help to reduce PCI DSS scope. While a PCI-listed P2PE solution does not completely remove the need for PCI DSS validation of the payment acceptance environment, the Customer back-end environments could potentially be considered out of scope.
- Use other technologies to reduce exposure and devalue the payment card data, such as tokenization. Customers should be aware of and understand the scope impact of various cloud-based encryption and tokenization solutions[11]—for example, those outsourced to the cloud, products developed in-house or off-the-shelf products.
    - Third-party cloud-based solutions could potentially limit the Customer's exposure to a clear-text primary account number (PAN), as payment card data can be stored with the Provider, and not by the Customer itself.
    - In-house hosted solutions, whether custom developed or off-the-shelf encryption or tokenization products, require the entity to protect the stored cardholder data within the solution, and will likely involve cryptography, key management and usage of segmentation techniques.
- Implement a dedicated physical infrastructure that is used only for the in-scope cloud environment. The scoping process will be simplified if all in-scope operations are limited to a known, defined set of physical and virtual system components that are managed independently from other components. Once these are defined, the Customer will be reliant on the Provider's ability to ensure that scope boundaries are maintained—for example, by ensuring that all segmentation controls are operating effectively and that any new components connected to the in-scope environment are immediately brought into scope and protected accordingly. Although service providers are required to perform testing of segmentation controls semiannually, continuous testing in a cloud environment would present validation of controls.
- Minimize reliance on third-party Providers for protecting payment card data. The more security controls for which the Provider is responsible, the greater the scope of the CDE will potentially be, thereby increasing the complexity involved in defining and maintaining CDE boundaries.

Ensuring that clear-text account data is never accessible in the cloud may also help to reduce the number of PCI DSS requirements applicable to the cloud environment. For instance, Customer performs all encryption and decryption operations and all key-management functions[12] in its own data center and uses a third-party cloud only to store or transmit encrypted data. In this scenario, clear-text data would never exist in the cloud environment—not even temporarily or in memory. Additionally, the cloud environment would never have access to cryptographic keys or key-management processes.

[11] Scoping SIG, Tokenization Taskforce, and PCI Security Standards Council, *PCI DSS Tokenization Guidelines*, Section 3.2, "Maximizing PCI DSS Scope Reduction" (PCI SSC, August 2011), https://www.pcisecuritystandards.org/documents/Tokenization_Guidelines_Info_Supplement.pdf.

[12] In accordance with PCI DSS Requirements.

It should be noted that the encrypted data is still in scope for PCI DSS (generally for the entity that controls or manages the encrypted data or the cryptographic keys[13]) to ensure that applicable controls are in place. However, by keeping all encryption/decryption and key-management operations isolated from the cloud, the number of PCI DSS requirements that the Provider is required to maintain may be reduced, as these requirements will instead be applicable to the Customer's own environment and personnel. The Provider will still be in scope for any PCI DSS requirements it manages on behalf of the Customer—for example, access controls managed by the Provider will need to be verified to ensure that only authorized persons (as determined by the Customer) have access to the encrypted data, and that access is not granted to unauthorized persons.

Alternatively, if clear-text account data is present (for example, in memory) in the cloud environment, or the ability to retrieve account data exists (for example, if decryption keys and encrypted data are present), all applicable PCI DSS requirements would apply to that environment.

For more information, refer to the Information Supplement *Guidance for PCI DSS Scoping and Network Segmentation,* which is intended to provide further understanding of scoping and segmentation principles as applicable to the PCI DSS environment.[14]

4.5.1 Security of Cloud Service Customer Systems

Customer systems used to access the cloud environment such as workstation, smartphone and Internet of Things (IoT) device, should not be overlooked, as they could potentially become weak links in a Customer's cloud security strategy. Customers need to ensure that their systems and internal processes do not provide unauthorized access to the cloud environment. For example, if a Customer workstation or other device is compromised, an attacker may be able to use credentials and an authorized channel to gain access to the cloud environment from the compromised Customer system. The Customer will therefore need to ensure that its Customer-side devices are appropriately secured and protected from unauthorized physical and logical access. ***Note:*** *Customer-side systems used to access cardholder data in the cloud would also be in scope for all applicable PCI DSS requirements.*

[13] For additional guidance, refer to FAQ "*Is encrypted cardholder data in scope for PCI DSS?"* on PCI SSC website.

[14] PCI Security Standards Council, *Guidance for PCI DSS Scoping and Network Segmentation* (PCI SSC, December 2016), https://www.pcisecuritystandards.org/documents/Guidance-PCI-DSS-Scoping-and-Segmentation_v1.pdf.

4.5.2 Scoping Examples for Different Cloud Deployment Models

For private cloud environments, segmentation efforts are focused on isolating CDE components from non-CDE components to reduce the number of systems in scope for PCI DSS. In public or shared cloud environments, segmentation between cloud tenants is critical for the security of the entire Customer environment and is additional to any segmentation managed by the Customer within its environment for the purposes of scoping.

A number of simple scoping examples are presented here to provide guidance.

Scenario	Environment Description	PCI DSS Scoping Guidance
Case 1: Private cloud hosted and controlled by entity seeking PCI DSS compliance, with segmentation	• All CDE VMs or containers are hosted on a single, dedicated hypervisor/server; non-CDE VMs are hosted on a separate hypervisor(s) or server(s). • Validated segmentation of CDE systems from non-CDE systems using a combination of physical and logical controls.[15]	The CDE hypervisor, VMs and containers, and all system components that are not segmented are in scope (segmentation must be validated as providing effective isolation).
Case 2: Private cloud hosted and controlled by entity seeking PCI DSS compliance, no segmentation	• All VMs or containers are hosted on one or more hypervisors or servers; some VMs transmit, process or store cardholder data and are considered CDE systems, and some do not process, transmit or store cardholder data or are not required to communicate to the CDE systems. • No segmentation of CDE systems from non-CDE systems.	The entire cloud environment and all connected systems are in scope and considered part of the CDE (similar to a flat network).

[15] Penetration Test Guidance Special Interest Group and PCI Security Standards Council, *Penetration Testing Guidance,* (PCI SSC, September 2017), https://www.pcisecuritystandards.org/documents/Penetration-Testing-Guidance-v1_1.pdf.

Scenario	Environment Description	PCI DSS Scoping Guidance
Case 3: Third-party Provider hosting a PCI DSS compliant public cloud supporting multiple Customers, with validated segmentation for Customer environments	• VMs may be on one or multiple hypervisors; all hypervisors and VMs are configured by Provider to support PCI DSS requirements. • Multiple Customers are hosted on each hypervisor. • One or more containers running within one or more dedicated container orchestration servers • Validated segmentation of Customer environments using a combination of physical and logical controls	The Provider is responsible for compliance of all elements of each underlying PCI DSS compliant cloud service provided to the Customer. Each Customer's scope would include its own environment (e.g., VMs, applications, etc.), the user and system configuration and management of the Provider services (Customers' access controls, etc.), and any other elements not managed by the Provider. Segmentation must be validated as providing effective isolation between Customers as part of the Provider's validation and may require additional validation as part of each Customer's validation.
Case 4: Third-party Provider hosting a PCI DSS compliant public cloud supporting multiple Customers, no Customer segmentation	• VMs may be on one or multiple hypervisors; all hypervisors configured by Provider to support PCI DSS requirements. • Multiple Customers are hosted on each hypervisor, VM configuration managed by each Customer. • Customers may be running isolated containers that may be in orchestration servers. • Segmentation between Customer environments is not verified. All systems are in scope.	This is not an environment likely to be compliant given that the entire cloud service and all Customer environments are in scope. Note that validating PCI DSS compliance may be intractable and infeasible, as every Customer environment would need to be included in the assessment (including Customer applications and software). Penetration testing would need to be conducted for all Customer environments as well. Due to regulatory and confidentiality requirements, individual cloud Customers will rarely be granted contractual audit rights to complete a PCI DSS assessment that includes other Customers.

Scenario	Environment Description	PCI DSS Scoping Guidance
Case 5: Hybrid cloud – Customer is utilizing a third-party Provider that hosts a PCI DSS compliant public cloud supporting multiple Customers, but also has connectivity with a mix of third-party hosted or internally hosted private cloud or on-premises Customer-hosted environment that utilizes services and direct connectivity between these different platforms/environments.	• VMs may be on one or multiple hypervisors. Hypervisors are likely to be configured by Provider to support PCI DSS requirements and may have Customer-managed hypervisors if connectivity exists to Customer on-premises environment. • Multiple Customers hosted on each hypervisor of Provider. Customer Private Cloud or on-premises environment would be dedicated to Customer. • One or more containers running within one or more dedicated container orchestration servers • Providers may provide services utilized for connectivity of the mixed platform environment that should be included as part of their annual validation.	**Scoping Guidance for Provider** The Provider is responsible for compliance of all elements of the cloud service it provides. Each Customer's scope would include its own environment (e.g., VMs, applications, services, etc.) and any other elements not managed by the Provider. If Provider provides the services allowing connectivity between the mixed environments, then this solution and Provider-managed equipment and service should be validated by the Provider. Alternatively, if for example the Provider enables services (either physical or virtual) to be configured and managed by the Customer, enabling connectivity between mixed environments, then the Customer scope would include all Customer-managed configurations, while the underlying infrastructure would be validated by the Provider. Segmentation controls utilized within the Provider-provided cloud must be validated as providing effective isolation between Customers as part of the Provider's validation. **Scoping Guidance for Customer** Customer is responsible for ensuring that solutions and services provided by its Provider for connectivity have been validated as compliant. Customer would be responsible for compliance of all elements it configures and manages within the Provider environment and any on-premises or private cloud connected to the CDE. Customer is responsible for ensuring that any connectivity between the Provider and any private cloud or on-premises environments is properly secured. Customer must validate that all segmentation controls utilized are effective and properly isolate the CDE within all environments of the mixed/hybrid platform.

Scenario	Environment Description	PCI DSS Scoping Guidance
Case 6: Provider is providing security services (authentication, authorization, auditing, etc.) to on-premises or hosted CDE systems that could affect the security of the CDE (for example, authentication and authorization service allowing access to cardholder data).	• IaaS VMs/PaaS configured by cloud Customer with custom-developed security services to support PCI DSS requirements	The Customer is responsible for compliance based on the responsibility matrix agreed upon with the Provider to maintain them in a PCI DSS compliant manner. It should be included as part of the Customer's and the Provider's validation.

5 PCI DSS Compliance Challenges

The distributed architectures of cloud environments add layers of technology and complexity that challenge traditional assessment methods. As a result, it may be particularly challenging to validate PCI DSS compliance in a distributed, dynamic infrastructure such as a public or multi-tenant environment. Examples of compliance challenges include but are not limited to the following:

- Customers may have little or no visibility into the Provider’s underlying infrastructure and the related security controls, which makes it difficult to identify which system components are in scope for a particular service or identify who is responsible for particular PCI DSS controls.
- Customers may have limited or no oversight or control over cardholder data storage. Organizations might not know where cardholder data is physically stored, or the location(s) can regularly change. For redundancy or high-availability reasons, data could be stored in multiple locations at any given time.
- It can be difficult to determine an appropriate sample size for dynamic, rapidly changing cloud environments and processes (for example, cloud-bursting, continual deployment and termination of virtual machines, dynamic IP addressing and so on).
- Some virtual components do not have the same levels of access control, logging and monitoring as their physical counterparts.
- Perimeter boundaries between Customer environments can be fluid.
- Public cloud environments are usually designed to allow access from anywhere on the internet.
- It can be challenging to verify who has access to cardholder data processed, transmitted or stored in the cloud environment.
- It can be challenging to collect, correlate and archive all the logs necessary to meet applicable PCI DSS requirements.
- Organizations using data-discovery tools to identify cardholder data in their environments, and to ensure that such data is not stored in unexpected places, may find that running such tools in a cloud environment can be difficult and result in incomplete results. It can be challenging for organizations to verify that cardholder card data has not "leaked" into the cloud.
- Not all services offered by a Provider may be included in the Provider’s PCI DSS compliance validation. Many Providers might not support the right to audit for their Customers.

These challenges will affect a number of factors related to how PCI DSS compliance is managed, including how segmentation is implemented, how PCI DSS assessments are scoped, how individual PCI DSS requirements are validated and which party will perform particular validation activities.

At a high level, Providers can be identified as those that have been validated as meeting a particular level of PCI DSS compliance and those that have not. The recommended practice for Customers with PCI DSS considerations is to work with Providers whose services have been independently validated as being PCI DSS compliant and have mechanisms available to Customers to attain such evidence.

5.1 What Does It Mean When a Provider States, "I Am PCI DSS Compliant"?

High confidence is placed in the statement "I am PCI DSS compliant," but what does this actually mean for the different parties involved?

Use of a PCI DSS compliant Provider does not automatically result in PCI DSS compliance for the Customers. The Customer should confirm that the Provider is PCI DSS compliant and that the services used by the Customer were included in the Provider's PCI DSS compliance validation (see Section 5.2, “Verifying the Scope of PCI DSS Validated Services and Components”). Moreover, the Customer must still ensure that it is using the service in a compliant manner and is also ultimately responsible for the security of its CHD—outsourcing daily management of a subset of PCI DSS requirements does not remove the Customer’s responsibility to ensure that CHD is properly secured and that PCI DSS controls are met. The Customer therefore must work with the Provider to ensure that evidence is provided to verify that PCI DSS controls are maintained on an ongoing basis. An Attestation of Compliance (AOC) reflects a single point in time only; however, maintaining compliance requires ongoing monitoring and periodic confirmation (e.g., at least once per year) that controls are in place and working effectively.

Even where a cloud service is validated for certain PCI DSS requirements, this validation does not automatically transfer to the Customer environments within that cloud service. For example, a Provider’s validation may have included use of up-to-date anti-virus software on the Provider’s systems; however, this validation might not extend to the individual Customer OS or VMs (such as in an IaaS service). Additionally, the Customer must still maintain compliance for all of its own operations—for example, by ensuring that anti-virus is installed and updated on all Customer-side systems used to connect into the cloud environment. Similarly, a Customer’s PCI DSS compliance does not result in any claim of compliance for the Provider, even if the Customer’s validation included elements of the service managed by the Provider. As a result, a Customer should confirm that services provided by the Provider support its PCI DSS compliance.

Regarding the applicability of one party’s PCI DSS compliance to the other, consider the following:

- If a Provider is compliant, this does not mean that its Customers are.
- If one or more of a Provider’s Customers is compliant, this does not mean that the Provider is compliant.
- If a Provider and the Customer are compliant, this does not mean that any other Customers are.

The Provider should ensure that any service offered as being PCI DSS compliant is accompanied by a clear and unambiguous explanation, supported by appropriate evidence, detailing which aspects of the service have been validated as compliant and which have not.

5.2 Verifying the Scope of PCI DSS Validated Services and Components

Customers will need to obtain details of the Provider’s PCI DSS compliance validation in order to determine whether the service they are using is wholly covered. Providers that have validated PCI DSS compliance may be included on a list published by a payment card brand or they may not; however, all PCI DSS validated Providers should be able to provide an AOC detailing the services and locations included in the PCI DSS compliance validation.

The following questions may be useful for Customers to ask their Providers:

- How long has the Provider been PCI DSS compliant? When was its last validation?
- What specific services were included in the validation?
- For each service used by the Customer, which PCI DSS requirements were included in the validation?
- Was the compliance validation conducted by a PCI-qualified and trained assessor (e.g., QSA or ISA)?
- Has the Provider supplied information (for example, a Responsibility Matrix) to Customers that clearly delineates the PCI DSS requirements met on behalf of the Customer?
- What specific services, facilities and system components were included in the validation?
- Are there any system components that the Provider relies on for delivery of the service that were not included in the PCI DSS validation?
- How does the Provider ensure that Customers using the PCI DSS compliant service cannot introduce non-compliant components to the environment or bypass any PCI DSS controls?

Providers should provide their Customers with evidence that clearly identifies the scope of their PCI DSS assessment, the specific PCI DSS requirements that the environment was assessed against and the date of the assessment. The Customer must have a detailed understanding of any security requirements that are not covered by the Provider and are therefore the Customer's responsibility to implement, manage and validate as part of its own PCI DSS compliance. The Customer should discuss its needs with the Provider to determine how the Provider can provide assurance that required controls are in place.

Providers that have undergone PCI DSS assessment to validate their compliance will have the results summarized in an AOC and detailed in a Report on Compliance (ROC) or Self-Assessment Questionnaire (SAQ) D for Service Providers. The Executive Summary and Scope of Work sections of the ROC should detail the scope of the assessment including the specific components, facilities and services that were assessed. Evaluation and attestation by a PCI-qualified and trained assessor provide higher levels of assurance that PCI DSS requirements were understood and testing procedures were followed. If a Provider's assessment was not performed by a PCI-qualified and trained assessor, the Customer may wish to ask additional questions about the rigor of the assessment, the qualification of the assessor and so on.

5.3 Verifying PCI DSS Controls Managed by the Cloud Service Provider

As with all hosted services in scope for PCI DSS, the Customer should request sufficient evidence and assurance from its Provider that all in-scope processes and components under the Provider's control are PCI DSS compliant. This verification may be completed by the Customer's assessor (such as a QSA or ISA) as part of the Customer's PCI DSS assessment. If the Provider has already undergone a PCI DSS assessment that was performed by another assessor, the Customer's assessor will need to verify that the Provider's validation is current, that the assessment covered all services provided to or used by the Customer and that all applicable requirements were found to be in place for the environments and systems in scope.

Providers that have undergone PCI DSS compliance assessment and validation should be able to provide their Customers with the following:

- Proof of compliance documentation (such as the AOC and applicable sections from the ROC), including date of compliance assessment
- Documented evidence of system components and services that were included in the PCI DSS assessment (as applicable to the service)
- Documented evidence of system components and services that were excluded from the PCI DSS assessment (as applicable to the service)
- Documented evidence (such as the AOC and applicable sections from the ROC) of compensating controls that were used to meet any of the PCI DSS requirements
- Appropriate contract language (per PCI DSS Requirements 12.8.2 and 12.9)

Providers that have not undergone PCI DSS compliance assessment will need to be included in their Customer's assessment. The Provider will need to agree to provide the Customer's assessor (i.e., an ISA or a QSA) with access to its environment or an environment managed by a third-party service provider (see Section 3.4, "Nested Service Provider Relationships,") in order for the Customer to complete its assessment. The Customer's assessor may require onsite access and detailed information from the Provider, including but not limited to:

- Access to systems, facilities and appropriate personnel for onsite reviews, interviews, physical walk-throughs, etc.
- Policies and procedures, process documentation, configuration standards, training records, incident response plans, etc.
- Evidence (such as configurations, screen shots, segmentation testing reports, process reviews, etc.) to show that all applicable PCI DSS requirements are being met for the in-scope system components
- Appropriate contract language (per PCI DSS Requirements 12.8.2 and 12.9)

The Customer and Provider will need to agree upon which assessment activities can be performed by the Customer and which testing is the responsibility of the Provider. For example, in an IaaS/PaaS service, the Customer may wish to test within its own environment and whatever else it can access, such as the boundaries between itself and other Customers, or between itself and the Provider's systems. However, if such testing is not permitted by the Provider, the Customer will have to rely on the Provider to perform and validate these requirements. In SaaS environments, the Customer will have limited or no visibility or permission to perform testing, and will generally be reliant on the Provider for all testing and validation. Defined testing activities and their associated controls and permissions should be detailed in the SLA.

Ideally, the Provider should be able to provide Customers with specific details, as applicable to the ongoing maintenance of PCI DSS compliance. For example, depending on the service provided, the Provider could provide copies of log files, patch update records or firewall rule-sets that specifically apply to an individual Customer's environment.

Providers wishing to provide a PCI DSS compliant service may want to consider isolating the PCI DSS compliant services from their non-PCI DSS compliant services. This may help to simplify the compliance validation process for both the Provider and for its individual Customers. It may also help the Provider to standardize the PCI DSS compliant services being provided to its Customers.

6 Security Considerations

While the use of cloud services can provide an attractive opportunity for organizations of all sizes to outsource and utilize centrally managed security resources, organizations should also be aware of the risks and challenges associated with a particular cloud choice before moving their sensitive data or services into the cloud environment. This section explores some of these additional security considerations.

6.1 Governance, Risk and Compliance

A primary challenge with cloud environments is governance, risk and compliance management, which is typically a shared responsibility between the Customer and Provider. In security, sharing undergoes rigorous scrutiny to bring clarity to both responsibility and accountability for performing certain control activities. The delineation of responsibilities emphasizes the importance of a strong governance, risk-management structure and SLAs. Without a clear governance strategy, the Customer may be unaware of issues arising from use of the cloud service, and the Provider may be unaware of issues within the Customer environment that could affect the provision of its service. During the cloud scoping, it is imperative to include the internal and external interfaces of the security architecture and demarcate the boundaries representing the governance domain of cloud user and Provider.

A responsibility matrix would be an appropriate approach to clearly define the governance strategy in the cloud, particularly when documented in the SLA. This enables clarity of responsibilities between Customer and Provider for operational security and risk management. Reporting and monitoring mechanisms should be made available by the Provider to its Customers to provide assurance that effective governance is applied and maintained by the Provider throughout the service. Examples of reports include, but are not limited to:

- Internal audit reports
- Independent audit reports
- Vulnerability and penetration testing reports
- Risk and vulnerability remediation action plan

6.1.1 Risk Management

Consistent with a risk management approach for in-house services, outsourced cloud services should be assessed against an organization’s risk strategy with the intent of identifying critical assets, analyzing potential risks to those assets and developing an appropriate risk treatment plan.

In traditional environments, the physical location of sensitive data can be restricted to dedicated systems and jurisdictions, facilitating the identification and implementation of effective risk-mitigation controls. However, the advent of new technologies requires a reevaluation of traditional risk strategies. For example, data in cloud environments is no longer tied to a physical system or location, reducing the effectiveness of traditional security mechanisms to protect data from risk. Traditional security approaches that build security controls to protect sensitive data may therefore need to evolve to address this emerging risk environment.

Similarly, traditional forms of risk assessment might not take into consideration particular cloud characteristics, such as a pay-as-you-go model or multi-tenancy (described in Section E2, "Multi-tenancy"), and may therefore require new or modified procedures.

6.1.2 Due Diligence

A Provider that stores, processes or transmits cardholder data, or can otherwise affect the security of the Customer, would be considered a third-party service Provider of the Customer. As with all service Providers, Customers should follow a thorough due-diligence process (see PCI DSS Requirement 12.8) prior to engaging the Provider. The specific due-diligence process and goals will vary for each Customer; however, common objectives typically include:

- Confirming that the Provider has a history of sound work practices and ethical behavior and is legitimately performing the services the Customer believes it to be
- Understanding the Provider's operational responsibilities, such as incident response, encryption and security monitoring
- Verifying that the Provider is compatible with the Customer's business image and risk profile
- Identifying potential risks or circumstances associated with the Provider that may affect the Customer's operations or business
- Identifying elements of the service that need to be clarified, and that need to be included in contracts or SLAs

Effective due diligence is not simply reading the Provider's marketing material or relying on a Provider's claims of PCI compliance; rather, it involves research, review and evidence collection. Customers should be sufficiently assured that they are engaging with a Provider that can meet their security and operational needs before undertaking any such engagements. The scope of the due-diligence exercise should consider, at a minimum, the topics discussed throughout this document, as applicable to a Customer's particular requirements.

Performing a due-diligence exercise prior to engagement with the Provider does not remove the need to perform an ongoing monitoring and review of the services offered by the Provider.

6.1.3 Service Level Agreements

The use of cloud services includes the deployment of a defined service model and should always be underwritten by comprehensive SLAs that conform to the international standards for cloud computing SLAs, including ISO/IEC 19086-1:2016 Information technology -- Cloud computing -- Service level agreement (SLA) framework -- Part 1: Overview and concepts. The secure delivery of any cloud service is dependent on the Provider's personnel, processes and technologies, while the secure usage of cloud services remains the responsibility of the Customer.

Typically, cloud hosting agreements are concerned with "up time" and high availability, with little or no mention or assurance of data integrity and security. However, the Customer is ultimately responsible for ensuring that the service that it is using meets its data integrity and security requirements and compliance obligations.

SLAs and other written agreements between the Provider and Customer should clearly identify the delineation of responsibilities between parties, including responsibilities for implementing and managing different security controls. These SLAs form the fundamental components of the manner in which operations and security will be undertaken and as a result should be established as a prerequisite to any cloud service implementation. PCI DSS compliance validation and testing activities (with the associated controls, permissions and schedules) should also be clearly detailed in the SLA.

Failure to develop and agree upon appropriate SLAs may result in issues for the Customer if the cloud service does not meet the needs and demands of its business. SLAs should be established and agreed upon as part of any contract and service negotiations. Performance, availability, integrity and confidentiality should be considered and SLAs agreed upon for each service managed or operated by the Provider. Written agreements should also cover activities and assurances to be provided by both parties upon termination of the service provision.

6.1.4 Business Continuity Plans and Disaster Recovery

Organizational requirements for business continuity plans (BCP), fault tolerance, high availability and disaster recovery (DR) controls apply to the Customer's outsourced environments as they do for Customer-managed facilities. Customers should consider whether the Provider's continuity and recovery procedures are sufficient to meet the Customer's organizational requirements, and the PCI DSS scope of the cloud service should include any fail-over sites and systems that might be used to store, process or transmit cardholder data in a BCP or DR situation. The ability to perform tests of the BCP and DR capabilities and to observe results of the Provider's testing should also be considered.

6.1.5 Human Resources

Management of the Provider's human resources is largely out of the control of the Customer. The Customer's due-diligence processes should include an understanding of the Provider's human resources and ongoing information security awareness training practices. PCI DSS Requirements 12.6 and12.7 provide a basis for assessing the employment screening process and security awareness training program of the Provider.

6.2 Facilities and Physical Security

Cloud services involve physical resources located within the Provider environment (including DR infrastructure discussed above) that are accessed remotely from the Customer's environment. Similar to other third-party providers, Providers of public and shared clouds provide services to multiple Customers whose data and virtual components co-exist in the same physical location and are managed by the same physical systems as those of other Customers. For a Provider facility, physical security controls need to be implemented that will protect the Provider's infrastructure as well as Customers' data. For PCI DSS validated Providers, the AOC should include a list of all physical locations that were evaluated as part of the PCI DSS compliance validation.

In a private cloud, the physical location of all components is known and can be verified. When using a public cloud, different elements of the environment, such as VMs, hypervisors, virtual network devices, etc., could be frequently relocated according to the Provider's service delivery strategy (e.g., load distribution, fault tolerance, high availability) across multiple physical sites. Verifying that appropriate physical security is in place can be challenging in an environment where data and infrastructure can be in multiple different locations at different times. A Customer should seek assurance that its physical security requirements are consistently applied across all potential locations.

6.3 Data Security Considerations

The following diagram shows a typical representation of the data life cycle:[16]

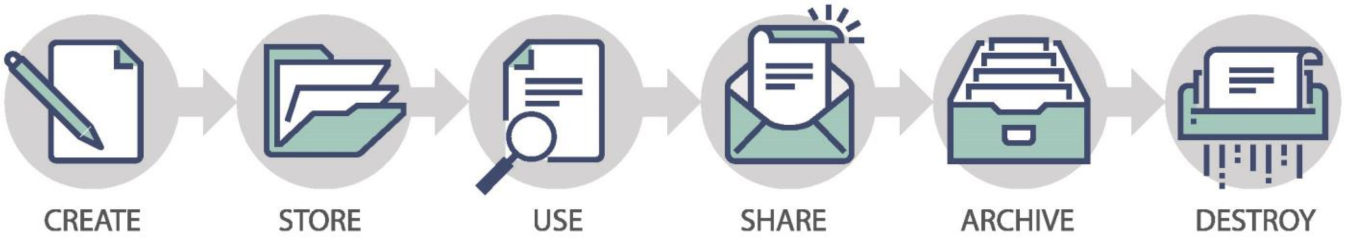


Figure 3. The typical data life cycle

It is important to identify and define the data used and produced in your environment and how the security aspects are managed all along its life cycle. For all cloud service categories, clear requirements for data retention, storage and secure disposal should form an integral part of the engagement process to ensure that sensitive data is:

- Retained for as long as needed
- Not retained any longer than needed
- Stored only in appropriate and secured locations
- Accessible only to those with a business need
- Handled in accordance with the Customer's security policy

6.3.1 Data Acquisition

The Customer will ultimately determine how and when the cardholder data is acquired in the cloud environment. End-to-end processes and data flows must be documented across both Customer and Provider networks, so that it is clearly understood where cardholder data is located and how it is traversing the infrastructure (see PCI DSS Requirement 1.1.2). This will also help the Customer and Provider to identify where each entity acquires and relinquishes cardholder data throughout the process.

[16] Cloud Security Alliance, *Security Guidance for Critical Areas of Focus in Cloud Computing V4.0* (Cloud Security Alliance, 2017). https://cloudsecurityalliance.org/download/security-guidance-v4.

6.3.1.1 Data Classification

Data classification and the management of data according to its classification will vary from organization to organization. A defined data classification system can help organizations identify data that is sensitive or confidential, and data with specific security needs. This in turn allows organizations to assign appropriate protection mechanisms based on the security needs of different data types, and helps to prevent sensitive data from being inadvertently mishandled or treated as non-sensitive.

6.3.1.2 Migration to a Cloud Environment

It is recommended that data security needs be evaluated for all types of information being migrated to a cloud environment, not only cardholder data. For example, operational data, security policies and procedures, system configurations and build standards, log files, audit reports, authentication credentials, cryptographic keys, incident response plans and employee contact details are just some of the types of data with different security requirements that may need to be considered. If data security processes are not clearly defined and documented, the data may be unintentionally exposed or subject to unnecessary risk that could result in loss or inappropriate disclosure.

6.3.1.3 Data Sovereignty and Legal Considerations

Depending on the deployment model and cloud service category adopted, and due to the dynamic nature of cloud operations, it may not be known where particular information actually resides. This may result in concerns over data ownership and potential conflicts between domestic or international legal and regulatory requirements. For example, the Provider’s infrastructure may result in data traversing or being stored in politically or economically unstable countries, or being subject to regional regulations.

Understanding the legal jurisdictions that apply to data in different countries or regions can be a challenge for the Customer. For example, Customers subject to regional laws restricting cross-border flows of data will need to verify all locations and flows of their data to ensure that their cloud services are compliant with their legal obligations.

Other legal considerations include requirements for electronic discovery, evidence preservation and integrity, and data custody. Providers should have documented processes for responding to legal requests for seizure of records, including data/audit logs belonging to the Provider and its Customers. Customers should understand the ramifications of such laws in the countries where their data exists, as well as the processes in which their Provider will engage.

Further to the data sovereignty considerations mentioned above, public Providers often have multiple data storage systems located in multiple data centers, which may often be in multiple countries or regions. Consequently, the Customer may not know the location of its data, or the data may exist in one or more of several locations at any particular time. Additionally, a Customer may have little or no visibility into the controls protecting its stored data. This can make validation of data security and access controls for a specific data set particularly challenging.

6.3.2 Data Storage and Persistence

In addition to the known range of intended storage locations, data may also be present in other Provider systems used for maintenance of the cloud infrastructure, such as VM images, backups, monitoring logs and so on. Cardholder data stored in memory could also be written to disk for recovery or high availability purposes (for example, in the case of virtual machine suspension or snapshot). Such stored data may easily be forgotten and so not protected by data security controls. All potential capture points should be identified and managed as necessary to prevent unintended or unsecured storage or transmission of sensitive data. Specialized tools and processes may be needed to locate and manage data stored on archived, off-line or relocated images.

Potential hypervisor access to data in memory should also be taken into consideration, to ensure that Customer-defined access controls are not unintentionally bypassed by Provider administrator personnel.

Organizations should ensure that their particular data security needs can be met by the cloud service before migrating that data into the cloud environment. Considerations should include how storing data types with different levels of sensitivity in the same virtual environment may affect the protection levels required for each data type. Cardholder data, user credentials and passwords, and cryptographic keys are examples of sensitive data that must be protected according to its individual needs.

6.3.3 Data Usage

Data must be accessible only to those with a business need, and handled in accordance with the established information security policy.

6.3.4 Sharing Data

Because all environments outside the Customer-controlled environment could potentially be untrusted, cloud services should support the secure transmission of cardholder data throughout the cloud infrastructure, between the Customer and cloud environments, between Customer environments and between the cloud infrastructure and other public networks. It is recommended that sensitive data be encrypted for all transmissions through any cloud environment that is not entirely private or controlled by the Customer. Cloud environments outside the Customer-controlled environment should be treated as open or public networks (see PCI DSS Requirement 4.1).

6.3.5 Decommissioning and Disposal

In a distributed cloud environment, verifying that all instances of cardholder data have been securely deleted in accordance with the Customer’s data-retention policy is subject to the same challenges identified above for validating data security and access controls. Disposal of cardholder data must be conducted using secure methods in accordance with PCI DSS requirements, and all locations of cardholder data from within both the Customer and Provider environments need to be included. The disposal method should ensure that data is not recoverable upon completion of the disposal process.

In addition to data disposal, resource-decommissioning requirements should be defined to support Customers’ future decisions to migrate to a new Provider, decommission their cloud resources or move out of a cloud environment altogether. The Provider should provide data-disposal mechanisms that provide assurance to the Customer that all data has been securely removed and deleted from the cloud environment. Procedures for termination of service should be clearly defined and documented, and considered in the context of being subject to regional regulations.

Customers may choose to ensure that all data is encrypted with strong cryptography (see Sections E10, “Data Encryption and Cryptographic Key Management,” and E.11, “Secure Cryptography Devices in the Cloud,” for further information) to reduce the risk to any residual data left behind on Provider systems. However, Customers should be aware that leaving potentially unknown quantities of encrypted data on Provider systems after their agreement has been terminated is likely to be a violation of their data-retention policy.

6.4 Incident Response and Forensic Investigation

Incident response, escalation procedures and forensics investigations, to ensure timely and effective handling of all security incidents, are critical to both Customers' and Providers' operations and essential elements as part of overall PCI DSS compliance. However, there are distinct differences and challenges both in handling of forensics data and how the incident response processes will need to adjust for each cloud service category.

Customers should work with their Providers to document security incident response, forensics and data breach notification-related roles and responsibilities as part of SLAs and contractual agreements, taking into consideration the need to comply with security incident management (i.e., PCI DSS Requirements 12.5.3, 12.8.3 and 12.10) and service provider requirements (i.e., PCI DSS Requirement 12.9).

6.4.1 Incident Response

Customers need to know when an issue, incident, or breach has occurred and the impact to their environment or to their data. Issues, incidents and data breaches should be communicated by the Provider to all affected Customers in a timely manner. Customers should also consider whether their Provider requires all Customers to immediately notify the Provider of potential breaches in their environments, allowing the Provider to respond more quickly to contain the breach and minimize its impact to other Customers. Based on the type of cloud service category used – relating to facilitating the storage, processing or transmitting of cardholder data—each phase of the incident response life cycle (e.g., as per NIST 800-61rev2) is affected at a different level. (Note that other international standard frameworks for incident response are ISO/IEC 27035 and ENISA "Strategies for Incident Response and Cyber Crisis Cooperation.") Definitions of what constitutes a breach or incident requiring notification between Customer and Provider should be agreed upon. Notification processes and timelines should be included in SLAs, and incident response plans should include notification requirements. In some cases, renegotiation of the SLAs may be required if the agreed-upon response time or viability of the information critical to the investigation is not adequate or sufficient.

6.4.2 Forensics Investigation

The potential for Customer data to be captured by third parties during a breach investigation should also be clearly understood. Incident investigation may involve consideration of legal and jurisdiction requirements, and these requirements should be included in SLAs or operational agreements (refer to Section 6.3.1.3, “Data Sovereignty and Legal Considerations”).

Forensic functionality should be specified in service level objectives (SLOs) incorporated into the SLA between the Customer and the Provider. SLOs may include requirements for notification, identification, preservation and access to potential evidence sources.[17]

Customers and law enforcement agencies require, and rely on Providers for, forensics support, and these obligations varies depending upon cloud service category as noted below.[18]

- **SaaS:** The capability for forensics is dependent upon the Provider’s support, as Customers have no control over the Provider’s environment. Forensics examiners may need to rely on high-level application logs available from the SaaS application. SLOs may include evidence sources such as logs from applications, web, database server, guest OS/host, portal, network capture and billing systems.
- **PaaS:** The capability for forensics is shared between Customers and Providers. Customers control the developed and hosted software application, and hence control forensics capability within the application, automatic logging to an external log server can be configured to capture the applicable audit trail. However, since the actual operation of the application is within the Provider’s controlled infrastructure, Customers must clearly identify Providers' responsibilities with respect to forensics investigation. SLOs may include evidence sources such as logs from the application, web, database server, guest OS/host, portal, network capture, billing and management portal.
- **IaaS:** The capability for forensics is shared between Customers and Providers. Customers have greater control over the range of potential evidence sources; however, some essential data only exists with Providers and under their control. Customers must clearly identify Providers' responsibilities with respect to forensics investigation. SLOs may include evidence sources such as logs from the cloud network perimeter, DNS servers, virtual machine monitor, APIs, host OS, network capture, billing and management portal.

Investigating potential breaches in cloud environments brings additional challenges. For example, compromised VM instances may be deactivated before anyone is aware that a breach occurred. It may be nearly impossible to properly investigate a breach when the source of the breach is no longer in use or even in existence.

6.4.3 Breach Notification

Customers should contractually require data breach notification from their Providers in clear and unambiguous language, taking into consideration the need to comply with local and global regulatory/breach laws, data privacy, security incident management and breach notification requirements.

[17] Cloud Security Alliance, *Security Guidance for Critical Areas of Focus in Cloud Computing V4.0* (Cloud Security Alliance, 2017). https://cloudsecurityalliance.org/download/security-guidance-v4.

[18] Cloud Security Alliance, *Mapping the Forensic Standard ISO/IEC 27037 to Cloud Computing* (Cloud Security Alliance, June 2013). https://downloads.cloudsecurityalliance.org/initiatives/imf/Mapping-the-Forensic-Standard-ISO-IEC-27037-to-Cloud-Computing.pdf.

Categories of data and security incidents should be prioritized with expected timelines for notification depending on incident type and requirements by the Customer, and where required, should include:

- Defined breach notification response responsibilities and obligations
- Customers' contact information (e.g., email in lower priority incidents, or 24/7 incident response call number for high priority incidents)
- Rights to involve Customer's or card brand forensic teams in the investigation of an incident or breach

Similarly, Providers may want to contractually require their Customers to inform them of any suspected or actual compromise—for example, breach of authentication credentials or identified service vulnerabilities. When an incident notification is received by the Provider or Customer, it is that party's responsibility to follow the implemented incident response plan (per PCI DSS Requirement 12.10). In addition, Providers are required to perform reviews at least quarterly to confirm that personnel are following established processes to respond to security alerts (PCI DSS Requirement 12.11).

6.5 Vulnerability Management

Proactive testing, identification and mitigation of vulnerabilities are an important part of achieving and maintaining compliance with PCI DSS for environments that utilize cloud services and systems. Within the PCI DSS are six distinct areas of vulnerability management: web application vulnerability testing (PCI DSS Requirement 6.6), internal network vulnerability scanning (PCI DSS Requirement 11.2.1), external network vulnerability scanning (PCI DSS Requirement 11.2.2), external penetration testing (PCI DSS Requirement 11.3.1), internal penetration testing (PCI DSS Requirement 11.3.2) and segmentation testing (PCI DSS Requirement 11.3.4).

Scoping is a critical element of vulnerability management. Customers need to ensure that they have properly identified all in-scope systems and services, including those provided by the Provider, those for which the Customer and Provider have shared responsibility and those that fall uniquely to the Customer (e.g., on-premises, private cloud, hybrid systems, or applications or systems that the Customer maintains). Penetration testing is used to confirm segmentation controls intended to constrain scope, and to proactively identify vulnerabilities that could be exploited to allow an attacker to breach these boundaries.

Testing vulnerabilities in the cloud also requires an in-depth understanding of the cloud deployment model to determine responsibility when it comes to performing the appropriate testing exercise. It is critical to understand the aspects of the environment that will be tested by the Provider and those that will be required to be tested by the Customer. It is not sufficient to identify responsibility by physical system, as each entity may have distinct or shared responsibility for aspects of a physical system (e.g., physical hardware, hypervisor, guest OS, application, configuration). These responsibilities will vary depending on cloud service delivery model (i.e., IaaS, SaaS, PaaS) or other division of control.

Where shared responsibility exists for vulnerability testing activities, the Customer and Provider should cooperate to ensure that these tests are performed and vulnerabilities are resolved. It is ultimately the Customer's responsibility to provide evidence that all required tests have been performed.

The PCI Security Standards Council has published the Information Supplement *Penetration Testing Guidance,* which is intended to provide further understanding of penetration testing requirements, methodologies and principles as applicable to the PCI DSS environment.[19]

6.5.1 *Web Application Vulnerability Testing*

All public-facing web applications must be protected, either by deploying an automated technical solution that detects and prevents web-based attacks or by employing application vulnerability security testing in accordance with PCI DSS Requirement 6.6. If a Provider is providing a web application (e.g., a SaaS application) that stores, processes, transmits or affects the security of cardholder data, the application should be either protected by a web application Firewall (or similar solution) or tested by the Provider, and this should be reflected on the Provider's AOC and responsibility matrix. Providers that expose APIs to their Customers should also perform testing and reporting on those APIs.

For applications that are supplied by the Customer (e.g., a microservice running on a PaaS or IaaS service), it is the Customer's responsibility to perform the web application vulnerability security testing as a part of its PCI compliance program. Providers should recognize this requirement and support these required testing activities (e.g., by supporting the ability to disable controls that would impede controlled testing, by supporting applications that may perform these operations or offering a service to perform these services).

6.5.2 *Network Vulnerability Scanning*

The Customer is responsible for ensuring that network vulnerability scans (PCI DSS Requirement 11.2) are performed in a compliant manner, although this requirement may be met by the Provider or another qualified third-party service. The entity that maintains the physical systems, networking devices, operating systems, and networked applications would generally be responsible for performing the scans necessary to ensure that these services are free from known vulnerabilities. For example, a Provider providing a SaaS service would generally be responsible for performing these scans and addressing the vulnerabilities found in the operating systems and applications it maintains. In PaaS and IaaS delivery models, there may be shared responsibility, as each entity must scan the devices, guest operating systems, hypervisors and applications for which it has responsibility.

6.5.2.1 *Internal Network Vulnerability Scanning*

All connected systems or services within the CDE must be scanned for vulnerabilities from within the internal network on a quarterly basis, or when significant changes have been made to firewall rules or network topology (e.g., virtual networks, security groups, access control lists (ACLs)). This operation may require the initiation of a VM or application within the network to perform the scanning service, or the Provider may wish to offer a qualified scanning service to support these activities.

[19] Penetration Test Guidance Special Interest Group and PCI Security Standards Council, *Penetration Testing Guidance,* (PCI SSC, September 2017), https://www.pcisecuritystandards.org/documents/Penetration-Testing-Guidance-v1_1.pdf.

6.5.2.2 External Network Vulnerability Scanning

All publicly addressable external IPs to the CDE must be scanned for vulnerabilities by an Authorized Scanning Vendor (ASV) on a quarterly basis, or when significant changes have been made to firewall rules or network topology (e.g., virtual networks, security groups, ACLs). Where virtual networks route across public IP space, these IP addresses must be included in this requirement, even if these IPs are considered private due to network controls (e.g., peering).

6.5.3 Penetration Testing

To meet the intent of PCI DSS Requirement 11.3, penetration testing of the CDE should include both the infrastructure (virtual network, security group, ACL, guest OS and above) along with application penetration testing where applicable.

6.5.3.1 External Penetration Test

Penetration testing activities from an external source must be initiated across the public untrusted network. If desired, these can be initiated from virtual instances hosted by the same Provider, but these systems should traverse the publicly routed IP space, and not have direct access to the CDE in order to ensure simulation of an external attack.

6.5.3.2 Internal Penetration Test

Penetration testing from within the internal network should be conducted from network segments that have sufficient access to critical systems. This test should simulate an attack by an entity against internal infrastructure (where permitted by the Provider), where the attacker has already gained access to the internal virtual network. Depending on the architecture of the virtual network, as well as on the security controls implemented, the internal penetration test could be conducted from within the CDE or from other internal virtual network segments (e.g., management VLAN).

6.5.3.3 Segmentation Testing

Testing segmentation (PCI DSS Requirement 11.3.4) in the cloud may be challenging, as numerous segmentation controls may be used to isolate systems, including ACLs, firewalls, Software Defined Networking (SDN) and network routing. In order to limit scope, penetration testing must be performed to test the suitability of these controls to confirm that the controls are operational and effective. Validation testing should include tests between VMs/instances, between applications/microservices hosted in separate virtual networks.

The Provider should perform penetration testing as part of its own PCI DSS assessment to demonstrate separation of client networks, in order to aid Customers in meeting this requirement. These should include tests between Provider management nodes and Customer systems, and between Customers on shared infrastructure. This testing should be used to verify that adequate restrictive network controls are in place to segregate the environments (e.g., assessing the effectiveness of the restricted network controls implemented at the security groups/firewall or ACLs).

6.5.4 *Notification of Testing*

Given the dynamic nature of the cloud, the shared environments in which vulnerability management testing must be conducted and the cooperative relationship of Providers and Customers, it may be necessary to provide notice and obtain prior permission before attempting certain testing activities. Notification may include anticipated testing dates, types of testing and details such as IP address range(s) affected. It is also important for the Customer to understand what testing activities are permitted by the Provider, and ensure that such activities do not harm other Customers within the environment. Any such constraints should be detailed in the Service Contracts, Terms of Service or Acceptable Use policy.

Appendix A: Sample PCI DSS Responsibilities for Different Cloud Service Categories

This Appendix expands on Table 2 (in Section 4) and provides examples of the ways in which responsibilities for PCI DSS requirements may be shared between Customers and Providers across some of the various cloud service categories. There will of course be exceptions and variations across each individual service, and this table is provided as a guideline for Customers and Providers to help plan discussions and negotiations.

The descriptions in this table are intended to reflect the Provider's responsibilities with respect to the services it provides, and do not consider the Provider's responsibilities for its internal infrastructure and operations not directly involved in providing services to their Customers. Similarly, Customer responsibilities do not include consideration for the Customer systems used to access the cloud service, or for any Customer systems in scope for PCI DSS that are outside the cloud service.

PCI DSS Requirements	Common Considerations	Example Responsibility Assignment for Management of Controls		
		IaaS	**PaaS**	**SaaS**
Requirement 1: *Install and maintain a firewall configuration to protect cardholder data.*	**IaaS:** Typically, network security is a shared responsibility: The Customer is responsible for securing networks within and between its own environments, while the Provider provides network security at the cloud perimeter and between the Provider's Customers. The Provider manages firewalls on the Provider-managed network and any infrastructure firewalls not visible to the Customer. Any firewalls above the infrastructure layer may be the responsibility of the Customer. The Provider-managed firewalls could also be shared by multiple Customers. **PaaS:** Firewalls above the infrastructure layer may be the responsibility of the Customer or Provider. The Customer could be directly responsible for implementing and managing firewalls on the provided platform, or it may define firewall configurations, which the Provider then implements for the Customer's environment. The Provider-managed firewalls could also be shared with other Customers. **SaaS:** The network is wholly owned and managed by the Provider, and consequently all firewall functions are typically managed by the Provider. **In all scenarios,** the Customer may still need to define, approve and periodically review the services, protocols, and ports permitted into its environment, even if the Provider is managing the firewalls in question.	**Customer and Provider**	**Customer and Provider**	**Provider**

PCI DSS Requirements	Common Considerations	Example Responsibility Assignment for Management of Controls		
		IaaS	PaaS	SaaS
Requirement 2: *Do not use vendor-supplied defaults for system passwords and other security parameters.*	**IaaS:** Secure configuration of OS and applications is typically the responsibility of the Customer, while secure configuration of underlying devices is the responsibility of the Provider. There may also be virtual devices that the Customer is responsible for maintaining. **PaaS:** The OS is often controlled by the Provider, but some services may include a level of Customer access to the OS—both parties will need to clarify which entity is applying secure configuration and hardening at the OS level. Applications and software above the OS are likely to be controlled by the Customer. Secure configuration of network devices will be managed by the Provider. **SaaS:** The Provider typically manages configuration and hardening of all devices, OS and applications.	**Customer and Provider**	**Customer and Provider**	**Provider**
Requirement 3: *Protect stored cardholder data.*	**IaaS and PaaS:** The Customer is generally responsible for the manner in which information is secured (such as the use of encryption mechanisms) and in what format—for example, flat files, database entries, etc. Physical locations of the information stores might be unknown to the Customer, and storage locations may need to be identified. Data retention is defined by the Customer; however, the Provider controls the actual storage areas. The use of controls to prevent unintended or additional retention (for example, via snapshots, backups, etc.) also needs to be considered. **SaaS:** Stored cardholder data is typically controlled and managed by the Provider as part of the predefined service. The Provider may also define the retention periods. Customers may have very little to no control over how or where their data, including CHD, is stored.	**Customer and Provider**	**Customer and Provider**	**Provider**

PCI DSS Requirements	Common Considerations	Example Responsibility Assignment for Management of Controls		
		IaaS	PaaS	SaaS
Requirement 4: *Encrypt transmission of cardholder data across open, public networks.*	**IaaS and PaaS:** Mechanisms for transmission are typically controlled by the Customer, while the underlying technology is managed by the Provider; however, this will depend on the technologies in use. Controls to prevent unintended transmission of data outside the Customer environment are generally maintained by the Provider, depending on the particular service. The Customer should be aware of how data is transmitted between components in order to ensure that data is encrypted for all transmissions over non-private channels. This may include transmissions within the Customer's own environment (for example, between Customer VMs). **SaaS:** The Provider retains full control over transmission mechanisms. The Customer has little to no control over how or where data is transmitted within the cloud environment. The Customer is responsible for ensuring that clear-text data is not passed to the Provider for transmission to public networks or untrusted environments (such as other cloud Customers).	**Customer**	**Customer and Provider**	**Provider**
Requirement 5: Protect all systems against malware and regularly update anti-virus software or programs.	**IaaS:** Protection of the OS and Customer VMs is typically the responsibility of the Customer. Anti-virus updates apply to the host OS as well as to any VMs in the Customer environment running their own OS. There may also be virtual devices that the Customer is responsible for keeping up to date. Anti-malware protection for underlying devices/infrastructure remains the responsibility of the Provider. It is important to confirm that the selected anti-virus solution used by the Customer is compatible with the underlying infrastructure managed by the Provider. **PaaS:** Anti-malware protection is generally managed by whoever controls the OS; some PaaS services include Customer responsibility for OS maintenance. Anti-virus updates will apply to the underlying OS as well as to any VMs in the Customer environment running their own OS. **SaaS:** The Provider typically manages the security and anti-virus for the environment.	**Customer**	**Customer and Provider**	**Provider**

PCI DSS Requirements	Common Considerations	Example Responsibility Assignment for Management of Controls		
		IaaS	**PaaS**	**SaaS**
Requirement 6: Develop and maintain secure systems and applications.	**IaaS:** Patching and maintenance of OS and applications are typically the responsibility of the Customer, while patching and maintenance of underlying devices remains the responsibility of the Provider. There may also be virtual devices that the Customer is responsible for maintaining. Secure coding is typically the Customer's responsibility (it may either use its own applications or choose secure commercial applications). **PaaS:** Patching and maintenance of underlying devices remain the responsibility of the Provider. OS patching and maintenance may also be controlled by the Provider; however, some PaaS services include Customer responsibility for OS maintenance: entities will need to determine which party is responsible for applying patches/updates. If the Provider provides patching, the Customer should verify that patches are deployed in a timely manner. Patching of applications is typically managed by the Customer, depending on the service and agreements. Secure coding of applications is the responsibility of whoever develops/controls the applications, which may be either the Customer or the Provider, and may vary for different applications. **SaaS:** The Customer may control or manage the APIs, or it may share responsibility with the Provider. The Provider typically manages patching and updates of all devices, OS and applications, and is also responsible for secure coding of software; however, the Customer should verify that patches are deployed in a timely manner.	**Customer and Provider**	**Customer and Provider**	**Customer and Provider**
Requirement 7: Restrict access to cardholder data by business need to know.	**IaaS and PaaS:** Generally, the Customer is responsible for defining access to its data files. Physical location of the information stores might be unknown to the Customer and may need to be identified. The Provider controls the physical storage areas, and Provider-managed access controls are often cumulative to the Customer-defined controls. The use of controls to prevent unintended access to data (for example, to data captured via snapshots, backups, etc.) should also be considered. **SaaS:** The Customer defines data access needs for its own personnel; however, access to data is ultimately controlled by the Provider.	**Customer and Provider**	**Customer and Provider**	**Customer and Provider**

PCI DSS Requirements	Common Considerations	Example Responsibility Assignment for Management of Controls		
		IaaS	PaaS	SaaS
Requirement 8: Identify and authenticate access to system components.	**IaaS and PaaS:** The Customer is responsible for ensuring that all accounts under its control use unique IDs and strong authentication. The Provider is responsible for ensuring that strong authentication is used for the underlying infrastructure. Compared to the IaaS cloud service category, the Provider retains significant administrative access rights In SaaS and PaaS cloud service categories. **SaaS:** The Provider has ultimate control of accounts at all levels. Depending on the particular service, the Customer may have the ability to create user-level accounts within the application or service, or it may be assigned user accounts that the Provider maintains on its behalf. The Customer is responsible for ensuring that all the accounts it uses have strong passwords.	**Customer and Provider**	**Customer and Provider**	**Customer and Provider**
Requirement 9: Restrict physical access to cardholder data.	**All cloud service categories:** Physical access to CHD is generally managed by the Provider for all cloud service categories. The Customer rarely has any physical access to cloud systems; and the Provider might not permit onsite visits or Customer audits. This will depend on the particular Provider as well as on the distribution of data across different locations; Customers may not know which location houses their data.	**Provider**	**Provider**	**Provider**
Requirement 10: Track and monitor all access to network resources and cardholder data.	**IaaS and PaaS:** The Provider typically manages monitoring and logging for underlying devices and infrastructure, including hypervisors, while the Customer is responsible for monitoring and logging within its own virtual environments. The ability to associate various log files in order to reconstruct events may require correlation between Customer-controlled logs and those controlled by the Provider. Some monitoring activities may be built into the service agreement for the Provider to manage on behalf of Customers. Details of what data will be captured and what will be made available to the Customer will need to be defined. **SaaS:** The Customer typically relies on the Provider for all monitoring and logging, but may have limited application-level logging such as user logon/logoff, account management and basic reporting.	**Customer and Provider**	**Customer and Provider**	**Provider**

PCI DSS Requirements	Common Considerations	Example Responsibility Assignment for Management of Controls		
		IaaS	**PaaS**	**SaaS**
Requirement 11: Regularly test security systems and processes.	**IaaS and PaaS:** Testing is generally managed by whoever has control of the particular aspect of the environment. However, Providers may prohibit Customer testing, in which case Customers may need to rely on the Provider. If the Provider is performing scans, the Customer needs to verify which instances/VMs are covered. Intrusion detection systems (IDS)/intrusion prevention systems (IPS) may not be provided by the Provider. Generally, the Customer can use file integrity monitoring (FIM) to monitor its own virtual environments (including data, applications and logs), while monitoring of system/device files is managed by the Provider. **SaaS:** The Customer does not have visibility or permission to perform scans and typically relies on the Provider for all scans, testing and monitoring.	**Customer and Provider**	**Customer and Provider**	**Provider**
Requirement 12: Maintain a policy that addresses information security for all personnel.	**All cloud service categories:** While the Provider and Customer may define agreed-upon procedures (for example, in the SLA), each party maintains its own security policies and internal procedures. Defined roles and responsibilities, training and personnel security requirements are the responsibility of each party for its respective personnel. Customers should ensure that the Provider policies and procedures are appropriate for the Customer's risk and security needs. Incident response in particular requires awareness and coordination between both parties.	**Customer and Provider**	**Customer and Provider**	**Customer and Provider**
Appendix A1: *Additional PCI DSS Requirements for Shared Hosting Providers*	The requirements for shared hosting Providers to ensure separation between Customers apply to third-party provided cloud services.	**Provider**	**Provider**	**Provider**

Appendix B: Sample Inventory

This appendix provides an example inventory for components used in cloud environments. Use of an inventory can help to identify the types of components involved in delivery of the service and responsibility for securing them. This example is not intended to be applicable to any particular scenario and is intended to provide a starting point for scoping discussions between Customers and Providers.

When preparing an inventory, consider the following:

- The type of information collected should be relevant for the Customer's business needs as well as the Provider's.
- The level of detail collected should be appropriate for both parties to reach a clear understanding of the components involved, their use and who manages/secures them.

Type/Layer	Component Description/Purpose	Type of Component	Number of Components	Implementation Notes	Responsibility for Securing Component
Note: Actual layers will vary depending on structure of Provider service offerings.	*For example: firewall, OS, application, web server, hypervisor, router, database, etc.*	*For example: is component physical, logical or virtual? Static or dynamic?*	*Number of components used in relation to Customer's service*	*Defined usage, location, etc., as applicable*	*For example: Provider only, Customer only, or shared*
Data					
Interfaces – APIs/GUIs					
Applications					
Programming stack					
Operating Systems					
Virtual Machines					
Virtual networking					

Type/Layer	Component Description/Purpose	Type of Component	Number of Components	Implementation Notes	Responsibility for Securing Component
Note: Actual layers will vary depending on structure of Provider service offerings.	*For example:* *firewall, OS, application, web server, hypervisor, router, database, etc.*	*For example:* *is component physical, logical or virtual?* *Static or dynamic?*	*Number of components used in relation to Customer's service*	*Defined usage, location, etc., as applicable*	*For example:* *Provider only, Customer only, or shared*
Hypervisors					
Containers					
Processing/Memory					
Data Storage					
Network devices					
Physical servers					
Physical facilities					

Note: *This is intended as a general example only. It may be necessary to reorganize the different technology layers or define additional component characteristics as applicable to a particular environment. Additionally, entities may wish to identify responsibilities for each system component in greater detail than provided for here.*

Appendix C: Sample PCI DSS Responsibility Management Matrix

A PCI DSS responsibility matrix may help to clarify and confirm how responsibilities for maintaining PCI DSS requirements are shared between the Customer and Provider. Responsibilities should always be defined in written agreements.

The table below is an example of a template that can be used by Providers and Customers to communicate the responsibilities and considerations for each PCI DSS requirement include:

- Does the Provider perform/manage/maintain the required control?
- How is the control implemented, and what are the supporting processes—e.g., process for patch updates would include details of testing, scheduling, approvals, etc.?
- What layers of the cloud architecture are covered by the Provider for the requirement? What layers of the architecture are not covered by the Provider and are specifically the responsibility of the Customer?
- How will the Provider provide ongoing assurance or evidence to the Customer that controls are met—for example, periodic reports, real-time notifications, results of testing, etc.?

PCI DSS Requirement	**Responsibility** (Provider only, Customer only, or shared)	Specific Coverage/ Scope of Customer Responsibility	Specific Coverage/ Scope of Provider Responsibility	How and When Provider Will Provide Evidence of Compliance to Customer
1.1 Establish firewall and router configuration standards that include the following:				
1.1.1 A formal process for approving and testing all network connections and changes to the firewall and router configurations				
1.1.2 Current network diagram that identifies all connections between the cardholder data environment and other networks, including any wireless networks				

PCI DSS Requirement	**Responsibility** (Provider only, Customer only, or shared)	**Specific Coverage/ Scope of Customer Responsibility**	**Specific Coverage/ Scope of Provider Responsibility**	**How and When Provider Will Provide Evidence of Compliance to Customer**
1.1.3 Current diagram that shows all cardholder data flows across systems and networks				
1.1.4 Requirements for a firewall at each Internet connection and between any demilitarized zone (DMZ) and the internal network zone				
1.1.5 Description of groups, roles, and responsibilities for logical management of network components				
…And so on.				

Note: *This is intended as an example only to provide a starting point for discussions between Customers and Providers. It is not intended as a requirement or an extension of PCI DSS compliance responsibilities. However, it may provide a useful tool to help to clarify responsibilities in agreements between Customers and Providers.*

Appendix D: PCI DSS Implementation Considerations

The questions in this appendix are intended as suggestions to help start conversations between Customers and Providers in order to understand the characteristics of a particular cloud environment, which may in turn help determine whether and how PCI DSS requirements can be met in that environment. These questions alone will not determine whether or not applicable PCI DSS requirements can be met; however, they may be a useful addition to questions directly related to specific PCI DSS requirements.

Information in this table incorporates guidance from the following sources:

- CSA Consensus Assessments Initiative Questionnaire
- ENISA Information Assurance Requirements

Please also refer to the PCI DSS Virtualization Guidelines for additional PCI DSS considerations for virtualization technologies.

PCI DSS Requirement	Considerations for Cloud Environments
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.*	▪ How is separation between tenants assured? ▪ How are boundaries enforced between trusted (internal to the Customer) networks and untrusted networks (such as Provider, other Customer or public-facing networks)? ▪ Are physical or virtual firewalls used? ▪ Who manages and audits firewall configurations? ▪ How are changes to firewall and network configurations tracked and managed? ▪ What technologies are used in the provision of the cloud service—e.g., hardware, software, virtual technologies? ○ Is there a current list of all hardware and software components in the environment? ○ Can the actual components used by a particular Customer be identified? ▪ How are configuration standards assured on different components of the infrastructure? ○ Are API interfaces standardized? ○ What is the process for provisioning new components? ○ Are virtual images hardened before being enabled? ○ Are hardened images protected from unauthorized access? ▪ How are systems with high security classifications segregated from systems with low security classifications? ▪ How are shared resources (such as processing, memory and storage) managed to ensure that they cannot be manipulated—for example, by overloading—in order to gain access to other Customer environments or data?

PCI DSS Requirement	Considerations for Cloud Environments
Protect Cardholder Data *Requirement 3: Protect stored cardholder data.* *Requirement 4: Encrypt transmission of cardholder data across open, public networks.*	▪ Where are the known data storage locations? Where are data centers located? ▪ Which legal jurisdiction(s) applies to Customer data? ▪ Does the Provider have any business, legal or regulatory requirements that could affect retention of Customer data? ▪ How is access to Customer data restricted to only that Customer's users and applications? ▪ How are VM images, snapshots and backups managed to prevent unnecessary capture of sensitive data? ▪ How is data securely deleted from memory and stored images? Will data remnants exist in terminated VMs? ▪ If cryptographic keys are provided by Provider, are unique keys generated for each Customer? ▪ Where are encryption/decryption processes being performed? Who controls each process? ▪ Where are cryptographic keys stored, and who controls the keys? Are data-encryption keys stored and managed separately from the data they protect? ▪ Where is encrypted data stored, and who has access to the keys and encrypted data? ▪ How are security and access defined for the virtualized resources used for generation of cryptographic keys? ▪ What process is followed in the event of a suspected key compromise? ▪ Is all Customer data securely purged from all Provider systems upon termination of the agreement? ▪ How are communications secured between Customer and other environments? How are communications secured within the cloud itself? ▪ Are APIs configured to enforce strong cryptography and authentication? ▪ Is mutual authentication implemented between Provider and Customer systems?

PCI DSS Requirement	Considerations for Cloud Environments
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.*	▪ Are VMs protected from within the VM or from the hypervisor? ▪ How is it ensured that VM images (including inactive and replicated VMs) have up-to-date anti-malware and patches before they are enabled for use? ▪ How are patches managed (e.g., prioritized, tested, approved and deployed), for both underlying Provider systems and provisioned Customer environments? ○ What is the process for each layer of the cloud service—e.g., physical network devices, firmware, host operating systems, hypervisors, virtualized components (including VMs, virtual network devices), applications, etc.? ▪ How are APIs and web services protected from vulnerabilities? ▪ Are standardized interfaces and coding languages used? ▪ How are development/test systems and data prevented from being inadvertently migrated into production environments, and vice versa (e.g., through virtual replication, imaging or snapshot mechanisms)?

PCI DSS Requirement	Considerations for Cloud Environments
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.*	▪ How is user authentication applied at different levels? ▪ How are layers of access controls managed to ensure that the aggregate access is not more than intended? ▪ Which Provider personnel have ability to access Customer data? ▪ How are Provider privilege assignments reviewed and monitored? ▪ How is segregation of duties maintained (for example, between administrative and auditing functions)? ○ Is administrative access to systems or hypervisor separate from access to Customer VMs and data stores? ○ Are separate credentials used for different security functions? ▪ How are least privilege and need to know determined for Provider personnel? ▪ How are credentials de-provisioned? ○ Does de-provisioning apply across all geographically distributed locations? ○ Could de-provisioned credentials be retained in offline images? ▪ Is remote access for Provider personnel permitted from untrusted networks? ▪ Are controls in place to prevent the capture of passwords in active memory, and to ensure that virtualized images do not contain authentication credentials? ▪ Is two-factor authentication required for Customer access? ▪ Does the Provider use any shared passwords (e.g., for maintenance)? ▪ Does the Provider maintain direct ownership and control over all data storage systems and facilities? ▪ Who has physical access to data centers and systems? ▪ How are data-storage systems protected from physical or direct console access? ▪ How are backups of VMs and data secured? ▪ How is physical media inventoried, secured, monitored, and tracked? ▪ Is media reused? How is data permanently removed from end-of-life or reusable media?

PCI DSS Requirement	Considerations for Cloud Environments
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.*	▪ How are activities traced back to individual Customer personnel or individual Provider personnel? ▪ Can the specific system components used by a Customer at a particular time be identified? ▪ What types of events are recorded in audit logs? ▪ How are audit logs correlated between Customer environments (such as a VM image) and Provider infrastructure (such as the hypervisor or underlying system)? ▪ How are audit logs monitored and reviewed? ▪ How are clocks synchronized between virtual instances and underlying systems/hardware? ▪ How is testing for wireless technologies performed and managed? ▪ How are all variations of VM images (including inactive VMs) scanned for vulnerabilities? ▪ What defenses are in place to protect against internal attacks (originating from Provider's or other Customer network) and external attacks (originating from the internet or other public network)? ▪ Is penetration testing performed across different layers of the environment (e.g., between VMs and the Provider's management network, or between Customers on shared infrastructure)? ▪ How is security testing managed for Provider infrastructure vs. Customer environments? ○ What testing are Customers allowed to perform on their internet-facing systems? ○ How are Customers prevented from performing penetration testing on other Customers' environments?

PCI DSS Requirement	Considerations for Cloud Environments
Maintain an Information Security Policy *Requirement 12: Maintain a policy that addresses information security for all personnel.*	▪ How does the Provider identify potential risks? ▪ Are Customers notified upon changes to the Provider's security and privacy policies? ▪ Does the Provider have mechanisms in place to ensure that secure operational procedures are followed? ▪ How does the Provider screen personnel? o Are different levels of screening used for different roles or regions? o Does screening cover all personnel with physical access to data centers at all locations? ▪ Does the Provider outsource any aspect of the cloud service to other Providers (e.g., data storage, security services, etc.)? ▪ What measures are taken to ensure that the Provider's security policies are maintained by its third-party providers? ▪ What processes are in place to detect, assess, escalate and respond to potential breaches? o What mechanisms are in place for Customers to report a suspected breach? o What criteria are used to define whether an incident or a breach has actually occurred? o What notifications are provided and when? o How would a breach at one Customer affect other Customers on the same infrastructure? o How is evidence collected, managed and shared? ▪ What happens to Customer data in the event of a breach to Provider systems? ▪ Can a Customer's data be collected as part of another Customer's (or Provider's) breach investigation (either by authorities or third-party investigators)? ▪ Are disaster-recovery processes, systems and facilities implemented with the same security controls as production environments?
***Appendix A1:** Additional PCI DSS Requirements for Shared Hosting Providers*	▪ How is isolation maintained across different layers, including between virtual machines, physical machines, networks, storage systems (e.g., storage area networks), management networks and support systems? ▪ What controls are in place to prevent data leakage between Customers, and between Customer and Provider? ▪ Are resource-isolation mechanisms in place?

Appendix E: Technical Security Considerations

Technical security considerations for cloud environments generally include all those that apply to virtualization technologies, as well as those directly related to the different deployment models and cloud service categories.

E.1 Evolving Security Technologies

As mentioned above, virtualization security considerations will also apply to cloud environments. There are many industry resources—including the "PCI DSS Virtualization Guidelines," available from the PCI SSC website—that discuss security considerations for the use of virtual technologies. Some of these considerations are:

- It is difficult to maintain up-to-date, secure configurations on virtual machines when they are being activated and deactivated in rapid cycles—virtual machines that are dormant for any period of time may be improperly secured or may introduce security vulnerabilities when activated.
- Security and monitoring solutions for virtual networks are still evolving and are not as mature as those available for physical networks; for instance, continuous segmentation testing between the cloud tenants' networks.

Additionally, traditional security software and security device functions often do not scale well to a cloud environment. For example:

- Management of VM-to-VM traffic that does not pass through traditional network-based security controls may require the use of additional host-based security controls to monitor and control the traffic.
- Traditional agent-based software security solutions that are not designed for virtualized environments may cause operational issues. For example, software agents, such as those often used for anti-virus protection, each use a small percentage of memory and processing resources; this can result in a large overhead when multiple agents are installed on multiple VMs on the same host.
- Scheduled scans or updates occurring simultaneously across multiple VMs may result in an extreme load on the underlying system and reduce overall performance of all hosted VMs.

E.2 Multi-tenancy

In a multi-tenant cloud environment, Customers generally have no knowledge of the other Customers with whom they share resources (for example, virtual infrastructure, data stores, etc.), or how other Customers are securing (or not securing) their environments that access the shared resources. Provider should perform a segmentation testing in a multi-tenant environment to ensure that cloud tenants are isolated from each other (see Section 6.5.3.3, "Segmentation Testing," for further information).

Whether unsavory Customers can pose a risk to other Customers using the same Provider will largely depend on the controls the Provider has in place to segregate Customers from one another, and to monitor and detect suspicious activity on the shared infrastructure and between Customer environments. Before engaging with a Provider, Customers should consider how the Provider verifies that their Customers are who they say they are, and how the Provider detects potentially suspicious behavior once the Customers are on board. Customers should also ask the Provider what controls it has in place to ensure that the security posture of one Customer cannot affect the security posture of another Customer.

E.3 Internet of Things and Fog Computing

"Smart" devices, such as mobile phones, tablets, wearables, smart sensors and IoT devices are increasingly used to accept and process payments. While these devices rely on the cloud-based ecosystem, they often use fog computing ("fog") as a layer between themselves and the cloud back end.[20]

Fog computing or fog networking is an emerging architecture for computing, storage, control and networking that distributes these services closer to end-users along the cloud-to-things continuum.[21] From an architectural standpoint, fog provides computing resources closer to the data-producing end-points at the edge. Devices in the fog computing (fog nodes) tend to be in a widely distributed deployment, with a very large number of nodes positioned for ingestion and processing of the data close to the source (i.e., IoT devices), providing interplay with the cloud back end.

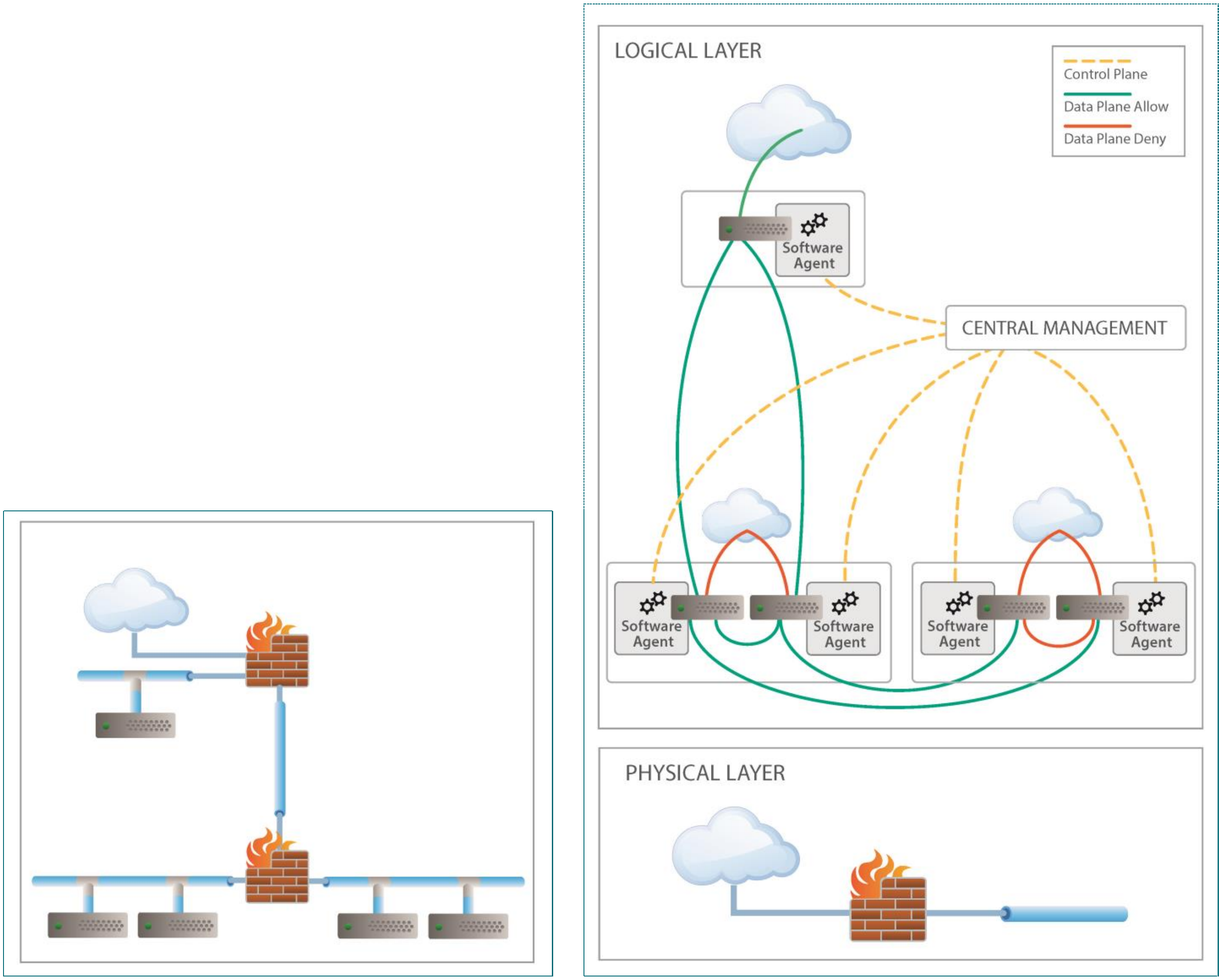
Fog computing can be seen as an extension of the traditional cloud-based computing architecture, service models and categories. As in cloud computing, fog nodes are deployed as private, community, public or hybrid nodes, supporting SaaS, PaaS and IaaS service categories. Therefore, principles and guidance in this document are applicable to IoT devices and fog computing ecosystems.

E.4 Software Defined Networking

Typically, a network’s structure and segmentation are defined with the use of network devices including firewalls and switches. SDN is the capability to abstract lower-level network functions by exposing the high-level capability through an API. SDN separates network activity into a control plane and a data plane, with the data plane directly performing data transport functions, and the control plane being a separate, central point for API calls to define the management of the data layer. SDNs are often used for micro-segmentation, which is the ability to define a point-to-point circuit between two nodes, preventing them from interacting with other systems, and preventing other systems from interacting with them, without an explicitly added policy.

[20] Michaela Iorga, Larry Feldman, Robert Barton, Michael J. Martin, Nedim Goren, and Charif Mahmoudi, *Fog Computing Conceptual Manual, NIST Special Publication 500-325,* (Gaithersburg: National Institute of Standards and Technology, March 2018). https://doi.org/10.6028/NIST.SP.500-325.

[21] Open Fog Consortium, https://www.openfogconsortium.org.



Traditional Defined Network

Software Defined Network

Figure E-1: Comparison of SDN and traditional networking

SDN environments should be evaluated in a consistent manner whether they are provided by an external Provider (in a public or hybrid cloud scenario) or an internal Provider (in an on-premises private cloud scenario). The SDN Provider must maintain a separation of duties from the SDN consumer (whether an external Customer or a business unit within an organization), even if they are in the same organization, to ensure that the SDN Provider is not able to alter the network topology or rules outside what is agreed upon with the Customer.

In multi-tenancy SDN environments, SDN consumers should be logically separated from each other within the SDN, and each SDN consumer should only be able to update the network policies applicable to its systems, and should not have visibility to the policies managed by other SDN consumers.

Typical evidence to demonstrate the strength of a solution should include:

- The actual code and syntax passed to the APIs to produce the traffic policies
- The change management supporting the code, including the processes to review, approve, commit and test it
- A comparison of the code provided, and the policies activated on the control plane
- Penetration tests of the data plane to confirm that one cannot bypass the policies applied and micro-segmentation controls
- Penetration tests of the control plane to validate that policies cannot be updated outside the established review process

In observing best practices, the boundaries between an SDN and a traditional network should be well demarcated to assure consistency in the way that controls are applied.

When using SDN, it is necessary to ensure that the intent of all requirements is fulfilled by either the SDN Provider or the SDN Customer; for example IDS/IPS functions, obfuscation of Internal IP addresses, hardening of the SDN infrastructure, time-synchronization, securing of centralized audit trail and periodic log review.

E.5 Intrusion Detection Systems (IDS)/Intrusion Prevention Systems (IPS)

As the Customer's access to network level data can be severely restricted in cloud environments, the responsibility for tracking intrusions at the network layer will often reside with the Provider, as the only entity that has sufficient privileges to do this across the underlying infrastructure.

Depending on the type of service layer (i.e., IaaS, PaaS or SaaS), Providers may be able to offer access to the intrusion detection system (IDS) or intrusion prevention system (IPS), or the data from the tools they use, to allow cloud service Customers to have auditing and alerting capabilities:

- **SaaS**: Since Customer access to low level network traffic is impossible, it must rely on Providers for IDS/IPS, monitoring and alerting.
- **PaaS**: Customers have no direct access to IDS/IPS functionality, as they are typically outside the platform, and therefore must rely on Providers for IDS/IPS.
- **IaaS**: Customer and Providers have shared responsibilities and should evaluate deploying IDS/IPS in key locations in the IaaS environment—for example, within virtual machines or containers, hypervisor/host systems, virtual network or underlying network infrastructure.

Other alternatives to implement intrusion detection or intrusion prevention in the cloud environments include locking traffic to specific defined resources based on role rather than IP (dynamic changes in the cloud make locking to IPs problematic: the range is either too big or too small), using a host-based IPS/IDS solution on the instance image or container, or routing network traffic through a third-party IPS/IDS service provider (i.e., a SECaaS service provider). These and other alternatives should be evaluated by the Provider, Customer and the assessor involved in compliance validation to ensure that they meet the intent of all applicable requirements of PCI DSS.

E.6 Hypervisor Access and Introspection

In large cloud environments it can be difficult to keep track of which hypervisors are running which VMs, as VMs can be dynamically assigned across a pool of hypervisors based upon load-balancing needs. Hypervisor configuration and access are particularly important, as a hypervisor provides a single point of entry to all its VMs and can potentially be used to gain access to sensitive data and resources on separate VMs.

An additional consideration is the degree to which the hypervisor is used to deliver security functionality to the VMs. For example, a simple, hardened hypervisor may be very secure but offer limited security capabilities, whereas a more complex, security-capable hypervisor with improved functionality could potentially present a greater risk if compromised.[22]

Functionality that allows the hypervisor to control and monitor individual VM activity from outside the VMs is known as introspection. Hypervisor introspection expands the functionality of the hypervisor to allow a deeper analysis of the data being processed by the VM, and typically includes visibility into stored data files as well as monitoring of network traffic, memory and program execution, and other elements of the VM.

Depending on the particular technology implemented, introspection can provide the Provider[23] with a level of real-time auditing of VM activity that may otherwise be unattainable. This can help the Provider to monitor for and detect suspicious activity within and between VMs. Additionally, introspection may facilitate cloud-efficient implementations of traditional security controls—for example, hypervisor-managed security functions such as malware protection, access controls, firewalling and intrusion detection between VMs.

Two potential challenges with introspection are that it can bypass role-based access controls and that it can be used without leaving a forensic audit trail within the VM itself. For example, to view a data file, a user typically authenticates to the VM, resulting in an authentication audit trail and ensuring that the user’s access is controlled according to that user's defined permissions. If file-access logging is enabled in the VM and the user views a file, the access is recorded to show what was accessed by whom and when.

With introspection, files can be accessed from within the privileged state of the hypervisor. As no authentication to the VM itself is required, file access leaves no audit trail on the VM, and the VM contains no evidence that the file was accessed. In this example, the access would need to be logged via the introspection tool itself, which would typically not be in the Customer’s control. While this may be less of an issue within a private cloud environment, it is an important consideration for Customers

[22] Tim Mather, *Don’t Bloat the Hypervisor! What to know about Introspection* (Webinar), 2011.

[23] Joe Security LLC, *Joe Security's Blog*, “Level Up: Introducing Hypervisor based Inspection in Joe Sandbox,” https://joesecurity.org/blog/68779205757215410 (20 June 2017).

of public cloud services. Research has been presented on detecting hypervisor activity within a VM [24] and hardware-assisted hypervisor introspection.[25]

Additionally, since introspection is designed to have full visibility into each VM, it may be difficult to restrict such access to only specific files or programs in memory. Any personnel (for example, Provider employees or possibly other hosted Customers) with access to the introspection function could potentially have access to data and processes on any VM running on that hypervisor. Introspection access must therefore be carefully managed, controlled and monitored to ensure that role-based access and segregation of duties are maintained. For example, the ability to configure introspection auditing should not be available to personnel with the ability to access hosted VMs via the introspection tool.

Providers can leverage products that provide Virtual Machine Introspection (VMI, also referred to as Guest Introspection). Providers have marketplaces with security products, such as those for AV, IPS and FIM, that provide protection from a central source appliance to virtual machines with minimal or no agent implementations. These products produce logs that can be reviewed by security information and event management SIEMs and participate in an alerting implementation.

Providers using introspection-based products should be able to provide their Customers with all applicable introspection logs for that Customer's environment including, but not limited to, authentication details, disk and memory access requests and API calls. All introspection activity should be mapped to the individual user account performing the activity, and logs should be reviewed on a continual basis to ensure that the integrity and confidentiality of Customer data have been maintained.

Where introspection is used by a third-party Provider, the Customers may wish to consider implementing data-level security controls (such as strong cryptography with all key storage and encryption/decryption operations external to the cloud service) to avoid exposing sensitive data to the enhanced monitoring features that introspection provides.

E.7 Containers

Containerization is an increasingly popular technology for efficiently running many instances of a server or application. It has resource, security and security isolation properties similar to those of virtual machines, but not the memory and performance overhead inherent in system-level hypervisors. Container orchestration platforms are becoming an increasingly popular service offering that allows Customers to spin an instance or a swarm of containers, and dynamically control the computing output capacity.

[24] Gary Wang, Zachary J. Estrada, Cuong Pham, Zbigniew Kalbarczyk, Ravishankar K. Iyer, "Hypervisor Introspection: A Technique for Evading Passive Virtual Machine Monitoring," University of Illinois at Urbana-Champaign, https://www.usenix.org/system/files/conference/woot15/woot15-paper-wang.pdf

[25] Jiangyong Shi, Yuexiang Yang, and Chuan Tang, "Hardware assisted hypervisor introspection," 17 May 2016, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4870477/#Sec21.

Containers are designed to be deployed as very lightweight, short-lived groups of systems. This provides speed and scalability by increasing the quantity of containers in the group dynamically. Because of this, controls should focus on the image templates and group as an aggregate, not the individual containers within the group.

Organizations should examine their container orchestration technology to confirm that it includes at least the following:

- Access controls to both the orchestration framework and the containers themselves such that different workloads do not have access to keys, identity tokens and other sensitive information used by other containers in the cluster.
- Process isolation of the running containers based on industry standard technologies such as kernel namespaces () and CGroups(). Best practices include controls to enforce kernel permission checks on privileged processes (sometimes referred to as capabilities).
- Access restriction of containers to host file systems and access restriction to container file systems by other containers.
- A fully functioning and *container-specific* system-call firewall that has a standard *default deny* rule and allows only known safe system calls. As an alternative, kernel security features such as AppArmor, SELinux, RSEC and Secure Computing Mode (seccomp) can be used to allow only system calls known to be safe.
- Strong network and administrative isolation between containers hosting different workloads based on a container-specific network interface such as the docker0 interface and Software Defined Networking (see Section E.4, “Software Defined Networking”).
- Where possible, kernel isolation, as offered by some hypervisor-based container solutions, to address the recommendations of the PCI DSS Virtualization Guidelines[26] for mixed-mode workloads.
- The ability to generate an audit of access approvals and review for the container orchestration system, demonstrating that access is limited to the least number of people appropriate for the in-scope workloads.
- Because most container solutions leverage Continuous Integration/Continuous Deployment development methodologies, the integrity of the CI/CD pipeline and change control must be evaluated to validate the strength of the above controls (see Section E12, “Change Detection for Cloud-based Systems”).
- The repository storing the container images must be managed securely, and creation of the images handled under strict change management.
- Images used for containers should be patched during the process of image creation (i.e., image template), and should undergo standard vulnerability assessment prior to committing the image to the repository.

[26] Virtualization Special Interest Group and PCI Security Standards Council, *PCI DSS Virtualization Guidelines* (PCI SSC, June 2011), https://www.pcisecuritystandards.org/documents/Virtualization_InfoSupp_v2.pdf.

- Where supported by the application, read-only containers should be used.

As containers can be used to achieve a similar level of execution, organizations may choose to use them to segment their PCI DSS scope in a manner similar to virtual machines. The Customer must validate that the container orchestration technology or solution offered by the Provider has all the features required to fully isolate containers. If isolation of workloads within a cluster cannot be assured through technical controls, the Customer should consider deployment of workloads into separate, workload-specific clusters. Some Providers may deliver the ability for an individual Customer to orchestrate multiple clusters, but some Providers may not offer this, and all workloads are deployed into shared, mixed workload clusters. In the latter case, a Customer may need to consider workload-specific Customer accounts to provide the isolation necessary.

Validating that a container solution has the necessary isolation strength in place will mean choosing a publicly vetted and trusted open source or commercial off-the-shelf solution that has demonstrated a strong security track record over time, rather than building a container technology in-house.

Many organizations have struggled to address quarterly and annual PCI DSS requirements for containers that may be created, run, retired and destroyed in a matter of days or hours. In some cases, a check of the container fleet may occur every week, such as for file integrity monitoring for PCI DSS Requirement 11.5, or over the course of weeks or months in the case of a penetration test. In cases where the container life cycle is shorter than the duration of a given PCI DSS control, consider sampling running instances across all in-scope container images (see Section E.9, "Elastic Resources Inventory and Control," for further information).

In addition, as part of the container instantiation pipeline, an organization may want to perform a vulnerability testing (see Section, 6.5, "Vulnerability Management) to address the challenge of implementing security controls in a highly fluid and elastic environment, as well as to evaluate the impact on PCI DSS requirements per PCI DSS Requirement 6.4.5.

E.8 Virtual Desktop Infrastructure in the Cloud

Virtual Desktop Infrastructure (VDI) is a virtualization technology that has its own compliance complexities that are discussed in the *PCI DSS Virtualization Guidelines* information supplement. When provided by a Provider as a cloud-based service offering, these complexities are compounded. The use of this technology is discussed in this context strictly as it pertains to delivery of VDI services by a Provider, henceforth referred to a Desktop-as-a-Service (DaaS).

Different host configurations may be provided by a Provider, which may uniquely affect environment scoping or service provider responsibilities pertaining to the many components involved, including the workstation, host system, guest OS, virtual network, hypervisor, image storage and file storage:

- **Traditional Remote Desktop:** While not commonly considered VDI, traditional remote desktops, terminal services or jump servers may also be delivered using virtualization or as a service by a

Provider. This model is addressed in the PCI Information Supplement on Scoping and Segmentation[27] for administrator workstations.

- **Traditional VDI:** The virtual desktop server is installed on a guest OS. All users share a single host and memory space, although images and files may be stored and accessed from other network locations, such as Network Attached Storage (NAS), Storage Area Networks (SAN) or Virtual Storage Area Network (VSAN).
- **Application-level VDI Workspace:** Stateless VDI workspaces no longer rely on a single guest OS. Instead, components of the virtual desktop may be hosted directly by the hypervisor, making it possible for the individual applications that comprise the "desktop experience" to be hosted and served from separate hypervisors, hosts and storage repositories residing in distinct virtual network segments or security zones. Providers providing stateless VDI should consult existing guidance on implications of mixed-mode service delivery to confirm the scope of each application and data store, and meet all isolation requirements and segmentation testing requirements necessary to limit scope (if applicable).

DaaS resources such as disk images and hypervisors may exist on separate networks. To ensure enumeration of all in-scope systems, all network provisions for CDE systems should follow designated paths that are identified by the responsible party's network and data flow diagrams.

When host systems and hypervisors are maintained and provisioned by a Provider, responsibility for secure configuration, patching, monitoring and testing generally falls to the Provider, and should be identified in the AOC of the Provider or responsibility matrix, or both. Similarly, where images are maintained by the Provider, storage and network access to images must be confirmed by the Provider's AOC (see Sections 5.2, "Verifying the Scope of PCI DSS Validated Services and Components," and 5.3, "Verifying PCI DSS Controls Managed by the Cloud Service Provider," for further information).

It is recommended that VDI disk images for CDE and non-CDE not be comingled, to ensure that non-CDE system components cannot mount CDE volumes—systems that house or access both CDE and non-CDE disk images should be considered in scope for PCI DSS. To reduce the scope of the PCI DSS assessment, the Provider must have necessary segmentation in place, and take express responsibility for these controls. For Providers providing shared hosting of VDI services, the hosting solution should enforce cloud tenant separation that is appropriate to multi-tenant cloud environments (see Section E2, "Multi-tenancy," for further information).

As server configurations vary, so too Customer workstation configurations may have a significant range of applicable controls. The three most common VDI workstation types include:

- **Zero Client:** These systems provide little more than video, keyboard and mouse interface to the network host. Without a processor or memory space to host malware, zero-client configurations offer minimal attack surface, and interception of sensitive data would require physical intrusion

[27] PCI Security Standards Council, *Guidance for PCI DSS Scoping and Network Segmentation* (PCI SSC, December 2016), https://www.pcisecuritystandards.org/documents/Guidance-PCI-DSS-Scoping-and-Segmentation_v1.pdf.

into the environment. Once physical controls and terminal capabilities are confirmed, an assessor may deem that no additional controls are required.

- **Thin Client:** A lightweight operating system is generally installed on these systems to support the connection to the host but may also include support for configurable applications that run locally. For this reason, thin-client workstations may be considered in scope and subject to many workstation controls. Certain PCI DSS requirements, such as Requirement 5, may be deemed inapplicable if the underlying OS is not commonly affected by malware. However, it is not sufficient to assume that thin-client workstations are out of scope entirely for these controls.
- **Conventional Workstation:** A full workstation that supports remote desktop connections to the CDE is considered fully in scope as an administrator workstation. The PCI Security Standards Council has published the Information Supplement *Guidance for PCI DSS Scoping and Network Segmentation,* which is intended to provide further understanding of scoping and segmentation principles for administrator workstations as applicable to the PCI DSS environment.[28]

In each of the scenarios above, it is crucial for the Provider and Customer to have clearly communicated responsibilities and network diagrams detailing segmentation boundaries and network locations pertaining to the physical hardware, hypervisor, virtual networks, guest OS, VDI/remote desktop service, disk images and all application binaries and configurations. Furthermore, as multi-tenant DaaS offerings are built upon mixed-mode environments and delivered as a service using varied levels of Customer access capability, thorough documentation and clear communication of responsibilities become critical to ensure that all relevant protections are confirmed in place and maintained by the responsible parties.

E.9 Elastic Resources Inventory and Control

Cloud architectures that are designed to elastically scale can present challenges that affect several PCI DSS Requirements, including Requirements 2.4, 6.4, 10.2.7, 11.3.1 and 11.3.2, among others. In order to best evaluate adherence of elastically generated environments to the requirements of the PCI Data Security Standard, an organization should assess the controls around the scaling automation.

- The template images from which elastic resources are instantiated should be reviewed for adherence to system configuration standards and security controls (e.g., PCI DSS Requirements 2.1, 2.2 and 5.1).
- The environment to which elastic resources are deployed should be evaluated for adherence to network segmentation rules (e.g., PCI DSS Requirements 1.2, 1.3 and 7.2).
- Because the inventory of system components (as required by PCI DSS Requirement 2.4) can change dynamically, the elastic automation should log all system creation and destruction activities to provide the ability to report on the inventory over time or at a specific instance in time.

[28] PCI Security Standards Council, *Guidance for PCI DSS Scoping and Network Segmentation* (PCI SSC, December 2016), https://www.pcisecuritystandards.org/documents/Guidance-PCI-DSS-Scoping-and-Segmentation_v1.pdf.

- The system should be able to provide a snapshot report of the currently deployed inventory (PCI DSS Requirement 2.4) to compare against the log files as a demonstration of inventory audit integrity.
- As an alternative to reviewing change management (PCI DSS Requirement 6.4) around individual elastic events, an organization could review the change management around the rules that govern elastic events, and the change management around the automation systems including the template images and the provisioning systems.
- Administration of the elastic automation, including development of the elastic automation pipeline, should be reviewed for strong authentication control and audit of activities (PCI DSS Requirements 8.1, 8.2 and 8.3).
- Consider including a vulnerability scan at the start of an instance, container API or service launch to provide a record of the artifact and a review of change impact.
- Penetration tests (PCI DSS Requirement 11.3) should include the provisioning and orchestration automation as well as the provisioned environment, including controls around the image templates and dynamic network creation.

E.10 Data Encryption and Cryptographic Key Management

In a public cloud environment, one Customer's data is typically stored with data belonging to multiple other Customers. This makes a public cloud an attractive target for attackers, as the potential gain may be greater than that to be attained from attacking a number of organizations individually. Strong data-level encryption should be enforced on all sensitive or potentially sensitive data stored in a public cloud. Because compromise of a Provider could result in unauthorized access to multiple data stores, it is recommended that cryptographic keys used to encrypt/decrypt sensitive data be stored and managed independently from the cloud service where the data is located. At a minimum, key-management servers should be located in a separate network segment and protected with access credentials separate from the VMs that are using the keys and the data encrypted with them.

Only defined, authorized key custodians should have access to cryptographic keys. Because access to both keys and encrypted data provides the ability to decrypt the data, Customers will need to verify who has access to cryptographic keys, who has access to the encrypted data and who has access to both. If a Customer shares encryption keys with the Provider, or engages the Provider as a key custodian, details of Provider access permissions and processes will also need to be reviewed and verified.

This consideration is particularly critical if cryptographic keys are stored or hosted by a third-party Provider that also hosts the encrypted data. If Provider personnel have access to a Customer's keys and the Customer's encrypted data, the Customer may have unintentionally granted the Provider ability to decrypt its sensitive data.

Any data that is decrypted in the cloud may be inadvertently captured in clear text in process memory or VMs via cloud maintenance functions (such as snapshots, backups, monitoring tools, etc.). To avoid this risk, Customers may choose to keep all encryption/decryption operations and key management on their own premises, and use a public cloud only for storage of the encrypted data.

Applicable controls must be applied to the encryption, decryption, and key-management processes to ensure that data can only be retrieved (decrypted) by those who are authorized with a defined business need.

Providers providing cryptographic-key management services for their Customers, such as Cloud HSM (see Section E.11, "Secure Cryptography Devices in the Cloud") should ensure that secret or private keys are not shared among Customers—each Customer should be provided with a unique key or set of keys. Secure channels for access to the cloud environment also require proper key management. If the Provider uses images or cloning for protecting VMs, it is strongly recommended that keys not be cloned as part of the VM image—each clone or VM instance should have its own key(s).[29]

Finally, the field of cryptography is continually evolving, and new cryptographic techniques and technologies (e.g., bit splitting and homomorphic encryption) are emerging to protect sensitive information. As these technologies are new, proper due diligence has to take place to weigh the pros and cons before deciding to acquire or build a solution utilizing these technologies.

E.11 Secure Cryptography Devices in the Cloud

The need for specialized cryptographic services to perform key management, random number generation, encryption, or decryption is applicable to cloud systems, microservices and other specialized workloads that must protect account data in the cloud or perform other security services. To accommodate this need, Providers may provide cryptographic functions that are accessible via API. Some such services may employ the use of actual PCI PTS- or FIPS-approved secure cryptography devices (SCDs), such as hardware security modules (HSM). Others may use specialized software-based services or non-certified hardware to perform these services. Due to the nature of cloud services, it may not be readily apparent whether the cryptography function is performed by an actual SCD, whether required certifications are in place or whether certain physical and logical controls are used by the Provider to protect the physical hardware.

The roles and responsibilities for all key management functions when using a cloud SCD must be documented to assure coverage of all requirements, including generating keys of appropriate strength, secure distribution of keys from the SCD and operational policies. It is the responsibility of the Customer to confirm that any applicable hardware requirements are being met by the Provider performing cryptographic functions on its behalf. For instance, PCI DSS specifies an SCD as one way in which a key used for decryption of stored account data at rest should be stored (PCI DSS Requirement 3.5.3). A Customer may meet this requirement using a Provider's cryptography service only if the Provider is using actual SCD hardware to deliver the service.

Similarly, the Provider is also responsible for physical security controls, key storage and distribution procedures, and documented cryptosystem policies. The controls being met by the Provider in the delivery of the compliant service(s) should be clearly identified in the Provider's AOC and responsibility matrix.

[29] The need for unique host keys has been well documented in a number of vendor technical papers.

Where the Customer uses keys generated by the SCD, it is responsible for documenting the key usage, custodianship, access controls and protection mechanisms after the keys have been retrieved.

In addition to PCI DSS, there are other PCI SSC programs that have stringent requirements for the use of SCDs that meet specific industry certifications; e.g., PCI PTS HSM v2 and FIPS 140-2 level 3. Examples of current standards that include SCD hardware stipulations include PCI P2PE, PCI PIN, PCI TSP, PCI SPoC, and PCI 3DS Core Security Standard. Furthermore, the requirements for these programs may impose additional physical and logical security requirements related to the protection of approved devices. Entities subject to these requirements should confirm that the Provider has been assessed to all relevant physical security requirements necessary to achieve compliance to applicable standards. Similarly, Providers may wish to proactively understand and implement the relevant security controls for these PCI programs if they intend to service markets that have compliance requirements under their programs (e.g., merchant acquirers, gateways).

E.12 Change Detection for Cloud-based Systems

To meet the objective of timely detection of unauthorized changes to system and configuration files for instances in the cloud, it is important to include monitoring for changes to provisioning code (i.e., scripts or templates) used for deployment of instances to production, as well as implementing controls around images used for deployment of new instances. Controls around images need to include permitting only authorized hardened images developed in accordance with industry best practices (e.g., NIST-800-190[30] or The Docker Security Benchmark[31]) and company configuration standards to be used for deployment of new instances. Prior versions of approved images, as well as generic images that have not been securely configured and approved by authorized personnel, must not be available for use for deployment of production systems.

In cases where an instance is based on a read-only operating system, such as CoreOS, use of traditional file integrity monitoring tools for monitoring system files within the running instance may no longer be applicable. However, such change-detection tools may still be required for monitoring integrity of system files for running instances that are based on operating systems where system and configuration files could be modified.

E.13 Security of Software Interfaces and APIs

Application programming interfaces (APIs) and other software interfaces and protocols (e.g., SOAP and RESTful) are an integral component of cloud computing, supporting interoperability and rapid delivery of cloud services. APIs can be configured to provide access to a variety of functions, allowing Customers and Providers to interact and manage their interactions within the cloud service.

[30] Murugiah Souppaya, John Morello, and Karen Scarfone, *Application Container Security Guide,* NIST Special Publication 800-190 (Gaithersburg, MD: National Institute of Standards and Technology, September 2017). http://csrc.nist.gov/publications/drafts/800-190/sp800-190-draft.pdf.

[31] Docker, Inc., *Docker Bench for Security, v1.3.3,* (Docker, Inc., 2015), https://github.com/docker/docker-bench-security.

As web services and APIs are by nature publicly accessible, their security is critical to the security of the resources to which they provide access. If not properly developed, managed and secured, these interfaces can be exploited or compromised, resulting in unexpected behavior and potentially unauthorized access. For example, a poorly coded API could result in weak authentication protocols, poor access controls or limited auditing capability. Such weaknesses could lead to the exposure of service functionality or sensitive data. If the APIs are not properly secured, they could also be exploited or altered by an attacker to redirect data flows or alter application behavior.

APIs and other public interfaces should be designed to prevent both accidental misuse and malicious attempts to bypass security controls. Resilient authentication and access controls, strong cryptography and real-time monitoring are examples of controls that should be in place to protect these interfaces.

When consuming APIs exposed by a Provider, it is important to ensure that all applicable PCI DSS requirements are met. For example, all API calls affecting cardholder data and the cardholder data environment must be logged and reviewed per PCI DSS Requirement 10. Or, when invoking a web services call that transmits cardholder data, it should be over an encrypted tunnel (e.g., SOAP native encryption or a TLS tunnel).

E.14 Identity and Access Management

Individual user identification and authentication for both Provider and Customer personnel is essential for access control and accountability (see PCI DSS Requirements 7 and 8). Shared credentials (such as user accounts and passwords) should not be used in the Provider environment—for example, for system administration and maintenance—nor should generic or shared accounts be assigned to or used by Customers.

The use of a single Customer credential that covers multiple cloud services for that Customer is also a potential concern. For example, let us say that a Provider issues a Customer a user account and password that has administrator privileges in one environment and user-level privileges for a separate, unrelated cloud service. Compromise of the Customer's user-level account in the second environment could therefore result in the attacker gaining administrator-level access to the first environment. Customer accounts and passwords should be unique for each service, and any account with elevated privilege (such as administrator) should be restricted for a specific service or function, and not used for activities or access that do not require such privilege. In certain scenarios, a multi-factor authentication may be required to access resources hosted in the cloud. For example, PCI DSS Requirement 8.2.2 requires multi-factor authentication for all remote network access to the CDE, and when public cloud services are part of a Customer's CDE, all such access will be considered remote access and will require multi-factor authentication.

The PCI Security Standards Council has published the Information Supplement *Multi-Factor Authentication,* which provides further guidance on the topic of multi-factor authentication.[32]

[32] PCI Security Standards Council, *Multi-Factor Authentication* (PCI SSC, February 2017), https://www.pcisecuritystandards.org/pdfs/Multi-Factor-Authentication-Guidance-v1.pdf.

E.15 Logging and Audit Trails

The ability to maintain an accurate and complete audit trail may require logs from all levels of the infrastructure, requiring involvement from both the Provider and the Customer.

For example, the Provider could manage system-level, operating system and hypervisor logs, while the Customer configures logging for its own VMs and applications. In this scenario, the ability to organize various log files into meaningful events would require correlation of Customer-controlled logs with those controlled by the Provider.

Providers are responsible for providing log data for resources managed by the Provider (for example, logs for infrastructure components (IaaS), logs for platform components (PaaS), logs for software components (SaaS), etc.). Providers should be able to segregate log data applicable to each Customer and provide it to each respective Customer for analysis without exposing log data from other Customers. In addition, the Provider must implement controls to protect the collected log data, including protection from unauthorized viewing, copying, printing, forwarding, editing and deleting.

Customers are responsible for ensuring that logging is enabled for components that are not managed by the Provider (e.g., application logs, event logs, etc.). In addition, Customers should ensure that log aggregation, correlation and monitoring are in place as required for all log sources, including the determination of log correlation and monitoring criteria. In an example scenario of shared logging responsibility, the Provider could manage system-level, operating system and hypervisor logs, while the Customer configures logging for its own VMs and applications.

The ability to organize various log files into meaningful events would often require correlation of Customer-controlled logs with those controlled by the Provider. Customers may choose to implement third-party correlation and monitoring tools or implement an in-house solution. At a minimum, the aggregated audit trail should contain information to reconstruct the events listed in PCI DSS Requirement 10.2 with sufficient details for the auditable events as required in PCI DSS Requirement 10.3.

Finally, the collected logs should be retained for at least one year, per PCI DSS Requirement 10.7.

The PCI Security Standards Council has published the Information Supplement *Effective Daily Log Monitoring,* which is intended to provide further information and guidance to help organizations address the challenges of maintaining effective log-management processes as applicable to the PCI DSS environment.[33]

[33] Effective Daily Log Monitoring Special Interest Group and PCI Data Security Standards Council, *Effective Daily Log Monitoring,* (PCI SSC, May 2016), https://www.pcisecuritystandards.org/documents/Effective-Daily-Log-Monitoring-Guidance.pdf.

Acknowledgements

PCI SSC would like to acknowledge the contribution of the Cloud Special Interest Group (SIG) in the preparation of this document, which is a revision of the document prepared by the 2013 Cloud SIG. The 2017 Cloud SIG consists of representatives from the following organizations:

12Feet Inc.
24 Solutions AB
3Delta Systems, Inc.
A1PlusSoft, Inc.
AccorHotels
Acumera, Inc.
Adobe
Advam Pty Ltd
Agio, LLC
Air Liquide USA LLC
Allstate Insurance
Ascension
AT&T Consulting services
Ather Technology Limited, dba Cianaa Technology
Automobile Club of Southern California
Bank of America N.A.
Bank of Montreal
Barclaycard
Basefarm A/S
BBPOS
BCD Travel
BDO USA, LLP
Beijing UGTech Co. Ltd.
Bl4ckswan S.r.l.
Bravecraft (Pty) Ltd
BT PLC.
California State University, Fullerton
Capita PLC
Capital One Financial Corporation
Card Security LLC
CardConnect
Carlson Wagonlit Travel
Cautela Labs, Inc.CBIZ Security & Advisory Services, LLC
CenturyLink
Chase Paymentech
Cimpress
Cisco
Citigroup Inc.
Coalfire
Comsec
Conferma Limited
Convergent Network Solutions, Ltd
CSC Government Solutions/CSRA LLC
Datatrans AG
Delap LLP
Delaware North Companies, Inc.
Direct Line Insurance Group PLC
Dixon Hughes Goodman, LLP
Dixons Carphone PLC
Elavon Merchant Services
Electronic Transactions Association
Emirates/Dnata
Equifax
Espion LTD
Experis Finance US LLC
Fiserv Solutions Inc.
FiveSec Labs Limited, dba Five Security
Focal Point Data Risk, LLC
Foregenix
Foresight IT Consulting Pty Ltd.
Games Workshop Ltd
Global Payments Direct Inc.

Gotham Technology Group, LLC

Grant Thornton

Habib Bank Limited

Harbour IT Pty Ltd

Heartland Payment Systems

Herman Miller Inc.Hewlett Packard Enterprise Company

HostedPCI

IBM Corporation

International Certificate Authority of Management System

IQ Information Quality

Irdeto B.V.

K3DES

Kirkpatrick Price, Inc. dba Raven Eye

KnowIT Secure AB

KYTE Consultants, Ltd.

L.L. Bean, Inc.

Little Caesars Enterprises, Inc

Lloyds Banking Group

Macy's, Inc.

Market America Inc

McGladrey LLP

Merchant Preservation Services, LLC d/b/a CampusGuard

National Bank of Abu Dhabi

NC Department of Natural and Cultural Resources

NCC Group PLC

NCC Services

NCI Secured Intelligence

Netsurion

Nettitude

Nintendo of America

NTT DATA INTELLILINK Corporation

NTT Security Ltd.

Online Business Systems

Optiv Security

Optomany, LTD.

Oracle Corporation

Orvis Company Inc, The

Paladion Networks Private LTD

PAN-Nordic Card Association

Parkingsoft

Payment Software Comapny (PSC)

PayPal Inc

PCI-PAL Limited Philips Electronics North America Corporation

Price and Associates CPAs, LLC, dba A-LIGN

PricewaterhouseCoopers LLP (PWC)

Pricewaterhousecoopers Private Limited - India

Protiviti

Reliant Info Security Inc.

Rock Pte. Ltd.

Rockwell Collins

RSM US LLP

SavvyPCI

Schellman & Company, LLC

Sec-1 Ltd.

Secure Technology Group

SecureState LLC

Securisea

Security Metrics

ServerChoice

SERVIRED

Shaw Cable Systems

Sikich LLP

SISA Information Security

SIX Payment Services Ltd

Skoda Minotti Risk Advisory Services, LLC

SLM Corporation

Sovereign Secure Ltd.

Sprint Nextel

Square

Stripe, Inc.
Sword & Shield Enterprise Security, Inc.
Sysxnet Limited DBA Sysnet Global Solutions
Target Corporation
Telstra
Tevora
Thales e-Security
The Herjavec Group Inc.
The Liquor Control Board of Ontario
The Regents of the University of California
Trustwave
TSYS
Uber Technologies, Inc.Ubitrak
UBS Card Center AG
UL Transaction Security
University of Colorado
University of North Carolina at Chapel Hill, The
Urbane Security
Verizon
VISTA InfoSec Private Limited
VMware, Inc.
Vodat International Limited
VTEX Cloud e-Commerce Platform
Wells Fargo
WestNet Consulting Services, Inc.
Worldline
WorldPay
XAC Automation Corporation
Yusufali & Associates LLC
ZeroFOX

Additional References

This document draws upon the following additional references. These sources are recommended as further guidance on securing cloud-computing environments.

Source[34]	URL
PCI Security Standards Council (PCI SSC)	https://www.pcisecuritystandards.org
Cloud Security Alliance (CSA)	https://cloudsecurityalliance.org/
European Network and Information Security Agency (ENISA)	http://www.enisa.europa.eu/
National Institute of Standards and Technology (NIST)	http://csrc.nist.gov/publications/
Information Commissioners Office (ICO)	http://www.ico.gov.uk/
ISACA	http://www.isaca.org/
ISC2 International Information Systems Security Certification Consortium, Inc.,	https://www.isc2.org/
The Open Web Application Security Project (OWASP)	https://www.owasp.org
Cloud Computing Security Research Library	http://searchcloudsecurity.com

[34] Links to third-party websites are subject to change.

About the PCI Security Standards Council

The PCI Security Standards Council is an open global forum that is responsible for the development, management, education and awareness of the PCI Data Security Standard (PCI DSS) and other standards that increase payment data security. Founded in 2006 by the major payment card brands American Express, Discover Financial Services, JCB International, Mastercard Worldwide, and Visa Inc., the Council has over 600 Participating Organizations representing merchants, banks, processors and vendors worldwide. To learn more about playing a part in securing payment card data globally, please visit pcisecuritystandards.org.