TECHNICAL WORKBOOK

PCI Compliance in the AWS Cloud

Report Date: June 19, 2015
Authors: Adam Gaydosh, QSA
Jordan Wiseman
Andrew Plato, QSA

ANITIAN

COPYRIGHT

For permission requests, write to the publisher, at the address below:

Anitian Corporation
1915 NW AmberGlen Parkway, Suite 105
Beaverton, OR, 97006
info@anitian.com
www.anitian.com

TABLE OF CONTENTS

1. Executive Summary

This workbook provides guidance on building an environment in Amazon Web Services that is compliant with the Payment Card Industry Data Security Standard (PCI DSS).

1.1. Intended Audience

The intended audience for this workbook includes:

- Organizations looking to build a PCI-compliant environment in AWS.
- PCI Qualified Security Assessors (QSA) and others assessing Cardholder Data Environments (CDEs) running in AWS.

The guidance in this workbook is written from the point of view of the customer building a compliant AWS environment.

1.2. Premises

This section lists Anitian's assumptions and premises that influence the content of this workbook.

1.2.1. As Is Disclaimer

Anitian is a Qualified Assessor Company (QSAC) and the author of this workbook. The content in this workbook is based upon Anitian's interpretations of the PCI. This content is provided "as is" with no guarantees expressed or implied. The content of this document is subject to change without notice. Likewise, future changes to the AWS environment may alter some of the guidance in this document.

Your PCI assessor may have different interpretations than Anitian and the guidance in this workbook.

None of the content in this workbook is intended to replace or supersede the requirements of the PCI DSS.

1.2.2. Intent

The purpose of this workbook is to provide guidance on deploying a PCI-compliant environment in AWS. The sections below outline how different AWS services can help support compliance with various PCI requirements.

While this workbook discusses AWS aspects useful for validating PCI compliance readiness as well as formal compliance, it does not offer step-by-step instructions on conducting an assessment of an AWS environment. However, it should assist QSAs in understanding how an AWS environment can be PCI-compliant.

1.2.3. Prerequisite Knowledge

Readers are expected to have an understanding of the following:

- The PCI DSS, currently at version 3.2
- How to manage an AWS environment
- The PCI Standards Council's Cloud Computing Guidelines

1.2.4. PCI Scoping

While this workbook discusses PCI scope reduction and segmentation within AWS, it is not a comprehensive guide on these issues. Consult with your PCI assessor or the PCI Standards Council for more information on scope reduction strategies.

1.2.5. Compensating Controls

This workbook does not address compensating controls for AWS implementations. However, you may use compensating controls within AWS, provided your assessor has validated them according to PCI rules.

2. AWS PCI COMPLIANCE OVERVIEW

This section provides a general overview of AWS PCI compliance.
For additional details, see Amazon's AWS PCI Level 1 FAQ.

2.1. AWS PCI Compliance Status

AWS is currently a PCI-compliant Level 1 Service Provider. Merchants and other service providers can use AWS to establish their own PCI-compliant environments. However, AWS operates on a shared responsibility model. Just because AWS is PCI DSS compliant, compliance does not automatically extend compliance to the hosted customer's environment.

AWS customers are responsible for all aspects of PCI compliance related to their environment within AWS. This includes AWS service configurations, guest operating systems, and requisite security controls (IDS, anti-virus, etc.).

Because AWS is a PCI-compliant service provider, it is not necessary for organizations hosting at AWS to assess the AWS infrastructure as part of the organization's PCI compliance. AWS's Attestation of Compliance (AOC) and Responsibility Matrix documents are all an assessor must review to validate the compliance of the infrastructure.

2.2. AWS PCI Compliance Scope

Amazon's AWS Service Provider validation assessment for PCI compliance includes the AWS Management Environment and underlying infrastructure, including the AWS GovCloud (US) region.

The majority of the AWS Services were included in the most recent AWS PCI DSS assessment. The list below shows those compliant services as well as a description of their function:

Service	Description
Auto Scaling	Automated, event-based instance provisioning
AWS CloudFormation	Creates and deploys templates of AWS resources
Amazon CloudFront	Content delivery web service
AWS CloudHSM	Cloud access to hardware security modules
AWS CloudTrail	Reporting on AWS API calls
AWS Direct Connect	Direct, private, dedicated connection to AWS
Amazon DynamoDB (DDB)	Scalable and highly available NoSQL data store
Amazon Elastic Beanstalk	Web application deployment and provisioning
Amazon Elastic Block Store (EBS)	Block-level storage for EC2 instances
Amazon Elastic Compute Cloud (EC2)	Scalable cloud machine instances

Service	Description
Elastic Load Balancing (ELB)	Application fault tolerance and load balancing
Elastic MapReduce (EMR)	Big data services
Amazon Glacier	Data archival storage
AWS Management Console	Web interface for managing all AWS services
AWS Identity and Access Management (IAM)	Access controls and key management
AWS Key Management Services (KMS)	Data encryption key management
Amazon Redshift	High-capacity data warehousing
Amazon Relational Database Service (RDS)	Database as a service
Amzaon Route 53	Scalable and highly available Domain Name System
Amazon Simple Storage Service (S3)	Store and retrieve any amount of data
Amazon SimpleDB (SDB)	Highly available and flexible non-relational data store
Amazon Simple Queuing Service (SQS)	Message queuing service
Amazon Simple Work Flow (SWF)	Service for coordinating application components
Amazon Virtual Private Cloud (VPC)	Isolated cloud resources within EC2

PCI compliance for AWS applies to the following regions, availability zones and edge locations (as of June 2015):

- US East (Northern Virginia)
- US West (Oregon)
- US West (Northern California)
- AWS GovCloud (US) (Oregon)
- EU (Ireland)
- Asia Pacific (Singapore)
- Asia Pacific (Tokyo)
- Asia Pacific (Sydney)
- South America (Sao Paulo)

2.2.1. Out of Scope AWS Services

AWS is constantly developing and deploying new services. However, not all of those new services are covered under AWS's current PCI attestation. That does not mean you cannot use those services. It means that if you do use them, then your assessor must review their configuration to ensure it is PCI-compliant.

For example, *Config* is a new service AWS relased recently. It is not in scope as of the most recent AWS Service Provider Assessment. If you use Config to track changes to CDE hosts, you must demonstrate that your usage meets the relevant PCI requirements. However, Config does run on AWS's compliant infrastructure. Therefore, while the service itself might not be certified as compliant, the infrastructure it runs on is compliant.

2.3. AWS PCI Compliance Responsibility

Determining which party is responsible for PCI requirements is one of the more complex aspects of cloud hosting. This section outlines how to define and organize a PCI compliance assessment for an AWS hosted environment.

This workbook outlines the areas where AWS can cover compliance requirements, and where you must cover them yourself. It is important that you consult the AWS PCI DSS "Responsibility Matrix," which defines exactly what AWS covers. Appendix A contains a summary of this matrix. If AWS does not cover a requirement, or if there is shared coverage, then your organization has responsibility to ensure the requirement is met.

Furthermore, you cannot arbitrarily choose to ignore a PCI requirement; you must meet all the requirements. However, it is possible that not all requirements are relevant to your organization. Your PCI assessor can clarify those that apply and those that do not.

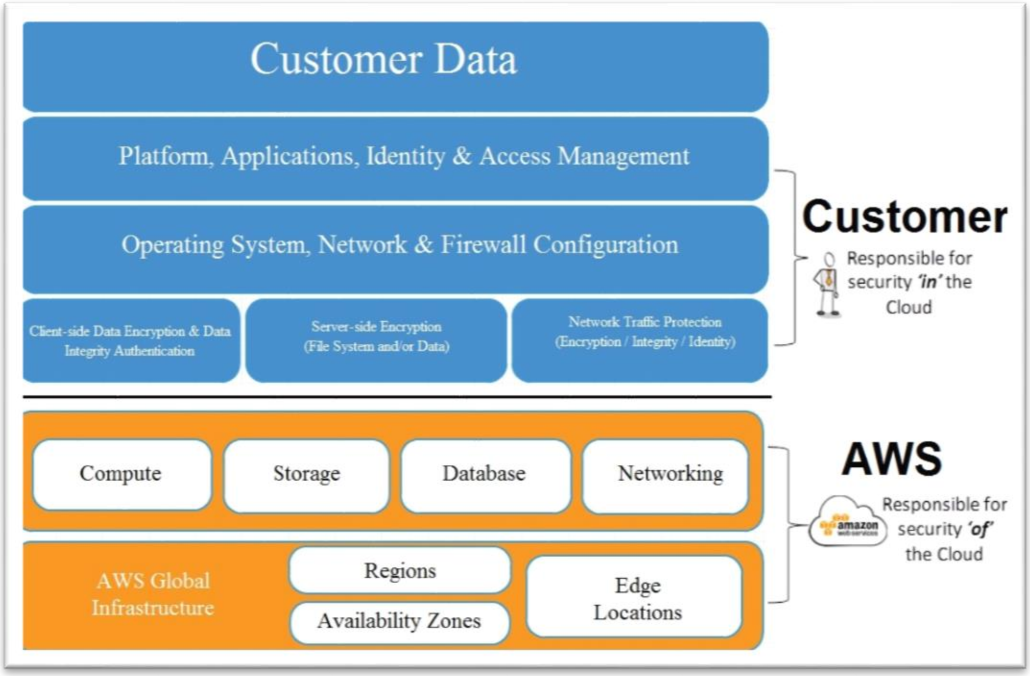

Customer Data
Platform, Applications, Identity & Access Management
Operating System, Network & Firewall Configuration
Client-side Data Encryption & Data Integrity Authentication
Server-side Encryption (File System and/or Data)
Network Traffic Protection (Encryption / Integrity / Identity)
Customer
Responsible for security 'in' the Cloud
Compute
Storage
Database
Networking
AWS Global Infrastructure
Regions
Availability Zones
Edge Locations
AWS
amazon web services
Responsible for security 'of' the Cloud

Figure 1 – Overview of AWS Shared Responsibilty

2.3.1. Amazon Responsibility - Security *of* the Cloud

Amazon is responsible for maintaining a PCI-compliant environment which you can use to aid in your compliance. This is referred to as "security ***of*** the cloud." AWS validates compliance annually. This is documented in AWS's Attestation of Compliance (AOC) document. As an AWS customer, you may request a copy (with a signed non-disclosure agreement).

2.3.2. Customer Reasonability - Security *in* the Cloud

You are responsible for designing, building, and maintaining a compliant environment in AWS. This is referred to "security ***in*** the cloud."

When you build your environment in AWS, part of that environment will be compliant because it uses AWS's compliant infrastructure. However, the final responsibility for PCI compliance rests with your organization (not AWS). The specifics are defined in the "Responsibility Matrix" as shown in Appendix A.

3. GENERAL PCI GUIDANCE

This section contains general guidance and strategies for meeting the twelve top-level PCI requirements using AWS services.

3.1. Requirement 1: Install and maintain a firewall configuration to protect cardholder data

The following AWS services can help support the firewall and network segmentation requirements of PCI:

- Amazon Virtual Private Cloud (Amazon VPC)
- Amazon EC2 Security Groups
- VPC Network ACLs

The topics below describe the strategies and considerations for utilizing these services for compliance with Requirement 1.

Amazon VPCs

VPCs are isolated customer cloud networks. VPCs allow for customers to have multiple environments with no connectivity between them, as if they were air-gapped physical networks. Alternately, routing tables can be used to connect VPCs (peering) or using VPNs and Direct Connect.

NOTE: *All subnets within the same VPC have a default route between them that cannot be removed.*

EC2 Security Groups

Security Groups are stateful firewall components in AWS EC2, which track established connections and only allow return traffic associated with the session. Security Group access control lists (ACLs) can be used to restrict traffic to and from instances at the IP address, port, and protocol level, for compliance with Requirement 1.3.6.

VPC Network ACLs

VPC Network ACLs apply at the subnet level, but are not stateful and (on their own) cannot be used to meet Requirement 1.3.6.

Other Strategies and Considerations

Anitian recommends using a dedicated cloud firewall AMI. Not only are these clearly stateful firewalls, but they can also offer many additional (and important) security functions, like intrusion prevention (Requirement 11.4 specifies the need for IDS/IPS).

There are several firewall Amazon Machine Images (AMIs) in the AWS Marketplace from companies such as Fortinet, Palo Alto, and CheckPoint. These firewall instances may require specific licensing from the vendor, but can provide familiar management interfaces and advanced capabilities.

3.2. Requirement 2: Do not use vendor-supplied defaults for system passwords and other security parameters

The following AWS services can help support the host hardening requirements of PCI:

- Amazon Elastic Compute Cloud (Amazon EC2)

The strategies and considerations for utilizing these services for compliance with Requirement 3 are discussed below.

Amazon EC2

When you use an Amazon-provided AMI to create an EC2 instance, AWS generates unique administrator and root passwords, encrypted with uniquely generated private keys. This helps support compliance with Requirement 2.1.

Furthermore, there are no default user accounts, since you must explicitly create them.

Other Strategies and Considerations

If you use non-Amazon images, then you are responsible for ensuring the defaults are changed. Consult with the relevant documentation for those images.

The AWS AOC covers the underlying security configuration management for the AWS services. However, it is your responsibility to create and implement security configuration standards for your EC2 instances. There are various solutions in the AWS marketplace that may assist with this requirement.

NOTE: *Anitian has created hardened AMIs for all available OSes, as both base servers and with a hardened web server. They include a supporting Security Configuration Standard documenting the hardening steps performed, as required by PCI DSS Requirement 2.2.*

3.3. Requirement 3: Protect stored cardholder data

The following AWS services can help support the encryption and key management requirements of PCI for cardholder data (CHD) at rest:

- Amazon Elastic Block Store (Amazon EBS)
- Amazon Simple Storage Service (Amazon S3)
- AWS Key Management Services (KMS)
- Amazon Relational Database Service (Amazon RDS)

The discussions below describe strategies and considerations for utilizing these services for compliance with Requirement 3.

Amazon EBS

AWS supports several different ways to store information securely. EBS non-root volumes and S3 buckets support volume-level encryption using AES-256. For EBS volumes, EBS manages encryption keys using a FIPS 140-2 compliant infrastructure.

If you store CHD (such as DB files on the file system of a dedicated DB server instance) on an instance's encrypted volume, additional encryption is required for the CHD in order to comply with Requirement 3.4.1. This is not unique to AWS, but is cited for completeness and clarification.

NOTE: *Not all EC2 instance types support encrypted EBS volumes. See EBS encryption.*

Amazon S3

Amazon S3 is a simple data storage service. It can encrypt stored objects with AES-256, and supports three different mechanisms for key management (see Server Side Encryption).

- **Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3)**

  When SSE-S3 is enabled for an object in an S3 bucket (in the Management Console), S3 encrypts the object with a unique data encryption key. This data encryption key is itself encrypted with a master key that the S3 service rotates annually.

- **Server-Side Encryption with AWS KMS-Managed Keys (SSE-KMS)**

  SSE-KMS uses an envelope key to encrypt each object's data encryption key. This allows greater control of who can decrypt data, and provides an audit log of key usage. KMS is discussed in the next section.

- **Server-Side Encryption with Customer-Provided Keys (SSE-C)**

  SSE-C allows you to use your own key and manage yourself. S3 never stores this key, only a message authentication hash of it to validate later use of the key when attempting to retrieve data.

By default, S3 is configured to use the SSE-S3. To use KMS or customer-supplied keys, you must specify the key management type when uploading the object via the console or the REST API. For further details, refer to S3 Upload Objects.

AWS KMS

KMS is AWS's encryption key management service. KMS provides automatic key rotation on an annual basis through the Management Console. See Amazon's KMS Cryptographic Details document for additional information.

AWS KMS also has a documented API for programmatic or third-party vendor support.

KMS uses a customer master key (CMK) as the key encrypting key (KEK), and a backing key as the data encryption key. Enabling key rotation rotates the backing key.

When you enable key rotation, a new CMK and associated backing key (HBK) are generated annually. These new keys are used going forward, and the old CMK/HBKs remain available for decryption only. You can also manually create a new CMK/HBK at any time and set it as the currently active key.

NOTE: *If you disable a CMK/HBK, it is no longer available to use, but attached EBS volumes that rely on the now-disabled key will continue to work. If that volume is detached from an Instance, you will have to re-enable the key to use the volume again.*

CloudTrail logs all of the AWS KMS actions (key creation, data encryption, key rotation, etc.) to the CloudTrail log files in the user-specified S3 bucket.

Amazon RDS

In addition to encrypted storage, Amazon RDS also supports two different methods of database encryption. RDS encrypts the underlying storage using Amazon KMS managed keys. This protects the data at rest. RDS also supports Transparent Data Encryption (TDE) for Microsoft SQL and Oracle instances.

IAM policies control who can access RDS instances, and what actions they can perform. The databases within an RDS instance, however, rely on their own internal platform-specific mechanisms to manage access to data. Make sure to configure PCI-relevant account and password policies within CDE databases in RDS.

Other Strategies and Considerations

If you run your own DB on an EC2 instance, you are fully responsible for managing the encryption of any CHD within the DB. This encryption should be performed using the standard strategies appropriate for the particular DB in use. Common examples are programmatic encryption of CHD at the field or column level, or encryption at the DB instance level, such as TDE for MS SQL.

3.4. Requirement 4: Encrypt transmission of cardholder data across open, public networks

The following AWS components can help support the transit encryption requirements of PCI:

- Elastic load balancers
- Network ACLs
- Security Groups
- Customer Gateways
- Virtual Private Gateways
- VPN Connections
- AWS Direct Connect

The strategies and considerations for utilizing these services for compliance with Requirement 4 are discussed below.

Elastic Load Balancers

Elastic load balancers support SSL/TLS and can offload processing encryption for secure communications for both internal and external connections. SSL/TLS negotiation is configured using security policies. AWS provides a number of predefined security policies (see the ELB Security Policies Table for further information). You can also create your own. Security policies allow you to define the SSL protocols and ciphers, as well as the order preference for the client server negotiation during the SSL handshake.

NOTE: *PCI DSS 3.2 states that "SSL and early TLS are not considered strong cryptography and cannot be used as a security control after June 30, 2016."*

Security Groups and Network ACLs

Security Groups and Network ACLs can block the use of insecure protocols based on network port.

Customer Gateways, Virtual Private Gateways, and VPN Connections

Customer Gateways, Virtual Private Gateways, and VPN Connections enable you to set up encrypted VPN tunnels into an AWS VPC. AWS supports a wide range of common VPN solutions (see the VPC FAQ), as well as a generic text configuration file. The VPN settings are automatically created by AWS, so that you can configure your endpoint to match. After creating a VPN connection (in the VPN Connections section of the VPC dashboard), you can download the configuration file needed to set up the customer end point. You can view the configuration file to validate the encryption used (SHA1/AES 128); see Section 4.3.4.5 below for implementation details.

AWS Direct Connect

Direct Connect provides a dedicated high-speed connection between customer environments and AWS, similar to MPLS. Direct Connect itself is not an encrypted connection, so you will need to verify the privacy of the circuit. Dependng on implementation, additional controls might be needed to comply with Requirement 4.1.

Other Strategies and Considerations

It is your responsibility to configure secure transit encryption for Internet-facing services running on EC2 instances, such as web servers. This should be included as part of the host hardening for Requirement 2.

Additionally, VPNs can be implemented on commercial firewalls or VPN AMIs running in the environment. You would be responsible for configuring the device to ensure alignment with Requirement 4.

3.5. Requirement 5: Protect all systems against malware and regularly update anti-virus software or programs

AWS does not provide anti-virus protection for EC2 instances. You are responsible for ensuring that all instances run appropriate anti-virus scans as well as log and report, as defined in PCI Requirement 5.

There are numerous anti-virus solutions available in the AWS Marketplace.

3.6. Requirement 6: Develop and maintain secure systems and applications

AWS does not provide vulnerability and patch management of EC2 instances. While the AMIs get updated periodically, launched and running instances must be managed like any other host. For example, for the list of updates to the Amazon Linux AMI see the AWS Linux Security Center.

There are vulnerability and patch management solutions available in the AWS Marketplace which can assist with complying with Requirement 6.1 and 6.2.

Additionally, there are no AWS services that directly address the PCI requirements for secure software development (Req. 6.3) and change control (Req. 6.4). While not directly related to the PCI requirements, CodeDeploy and CodeCommit can assit with general source code management and deployment.

However, network segmentation technologies (discussed in Requirement 1 above) can separate production and development environments (Req. 6.4.1).

NOTE: *AWS Config records changes to resources within AWS itself, but not to applications and within EC2 instances.*

3.7. Requirement 7: Restrict access to cardholder data by business need to know

The following AWS components can help support the access control requirements of PCI:

- AWS Identity and Access Management (IAM)
- AWS Directory Service

IAM

IAM service supports role-based access control within AWS. However, it is your responsibility to manage user roles and rights within the IAM service.

IAM supports users, groups and roles, as well as management of encryption keys.

Directory Service

Directory Service is a Microsoft Active Directory (AD) compatible directory service. It allows you to create one (or more) instances called Simple AD. It can be deployed as a stand-alone Simple AD or can connect to an on-premise Microsoft AD infrastructure.

Directory Service can manage access to AWS resources as well Microsoft AD-compatible systems and applications.

You must use a third-party tool to administer the AWS Directory Service, such as the Microsoft Active Directory Administration tools included with Windows Server. For more information see the Admin Guide Directory Management.

NOTE: *Simple AD directories do not support Microsoft Active Directory Web Services interfaces.*

Other Strategies and Considerations

It is your responsibility to ensure all user roles and rights are documented with least privilege rights clearly explained.

3.8. Requirement 8: Identify and authenticate access to system components

The following AWS services can help support the account management requirements of PCI:

- IAM
- Directory Service

IAM

IAM supports password policies in accordance with Requirement 8, with the exception of account lockouts for invalid login attempts (Req. 8.1.6), minimum lockout durations (Req. 8.1.7), and idle session timeouts (Req. 8.1.8). Meeting these requirements with IAM requires using a PCI compliant external identity provider that can enforce these requirements, or Directory Service.

NOTE: *IAM is only used for identity and access management to AWS resources, not authentication to EC2 instances and applications.*

Directory Service

Simple AD supports all of the password and account policy settings that Microsoft AD uses, which fully supports Requirement 8 (Create a Directory).

Other Strategies and Considerations

For any directory services running on an EC2 instance, it is the customer's responsibility to ensure that all password policies are configured to meet Requirement 8.

3.9. Requirement 9: Restrict Physical Access to Cardholder Data

AWS's AOC fully covers the physical security of AWS for Requirement 9. As long as you host your entire PCI environment in AWS, this requirement is covered.

AWS's AOC does not cover any in-scope assets hosted outside of AWS.

3.10. Requirement 10: Track and monitor all access to network resources and cardholder data

The following AWS services can help support the log management requirements of PCI:

- AWS CloudTrail
- S3

CloudTrail

The AWS CloudTrail service can assist with tracking and monitoring access to a CDE in EC2. The primary components supported by CloudTrail are log aggregation, alerting, and retention (Req. 10.5 to 10.7).

It is your responsibility to ensure that all EC2 instances generate the required security events (Req. 10.2) and detail (Req. 10.3) for management in CloudTrail.

The CloudTrail Event Record Body supports all specific elements in Requirement 10.3; for further information, see the Event Reference Record.

S3

Retention policies for CloudTrail data are configured in S3. By default, the retention period is infinite, but is fully configurable (see Lifecycle Configuration).

NOTE: *For a cost-effective way to comply with Requirement 10.7, you can use S3 Lifecycle Configuration to set the retention period to 90 days and automatically archive older data to the Amazon Glacier storage service for long-term retention (required to be at least one year).*

Additionally, you must enable access control on the S3 bucket storing the CloudTrail logs. This must include limiting bucket write access to CloudTrail and bucket read access to authorized users.

Other Strategies and Considerations

Amazon's CloudTrail is a basic logging service that can fulfill the PCI requirements for logging. However, if you already use a Security Information and Event Management (SIEM) product, you can likely find an AMI for it in the Amazon Marketplace. Alternatively, CloudTrail has an API, which your SIEM product may be able to use for advanced data correlation. Check with your SIEM vendor for additional information.

You must configure EC2 instances for network time protocol (NTP) to comply with Requirement 10.4.

3.11. Requirement 11: Regularly test security systems and processes

AWS's AOC fully covers detection of rogue wireless access points (Req. 11.1).

AWS does not provide vulnerability scanning, (Req. 1.2), penetration testing (Req. 11.3), intrusion prevention (Req. 11.4) or file change detection (Req. 11.5) within EC2 instances. However, there are numerous solutions in the AWS marketplace supporting many of these requirements.

NOTE: *Penetration testing must be scheduled and approved through AWS. See AWS Penetration Testing for further details*

3.12. Requirement 12: Maintain a policy that addresses information security for all personnel

AWS does not provide any of the policy documentation as defined in Requirement 12 (and other PCI requirements). You will need to write this material on your own.

3.13. Requirement A.1: Shared hosting providers must protect the cardholder data environment

If you provide shared hosting as part of your EC2 instances, you are fully responsible for protecting your customers' CHD. You will need to segment and isolate the CDE correctly to comply with Requirement A.1. The following services can assist with this:

- Requirement 1 – VPCs, Security Groups
- Requirements 7 and 8 – IAM and Directory Service

4. Reference Architectures

This section defines three common AWS reference architectures to help you build or assess a PCI-compliant environment.

1. **Dedicated**: An AWS PCI environment that is not connected to anything else
2. **Segmented**: A CDE and in-scope systems within a larger AWS environment
3. **Connected**: An environment that has both AWS and on-premise items

These reference architectures use Microsoft Windows platforms for the web and application tiers, and Amazon RDS for the database tier. While other OS platforms may have slightly different configurations, the architectures are generally the same.

NOTE: *Determining the scope of compliance in an AWS hosted environment is largely the same as scoping an on-premise environment. The scope of compliance is dependent upon the cardholder data flows and segmentation strategies in use.*

4.1. Architecture 1: Dedicated

This architecture demonstrates an e-commerce website hosted in a dedicated Amazon AWS environment that is isolated from any other network.

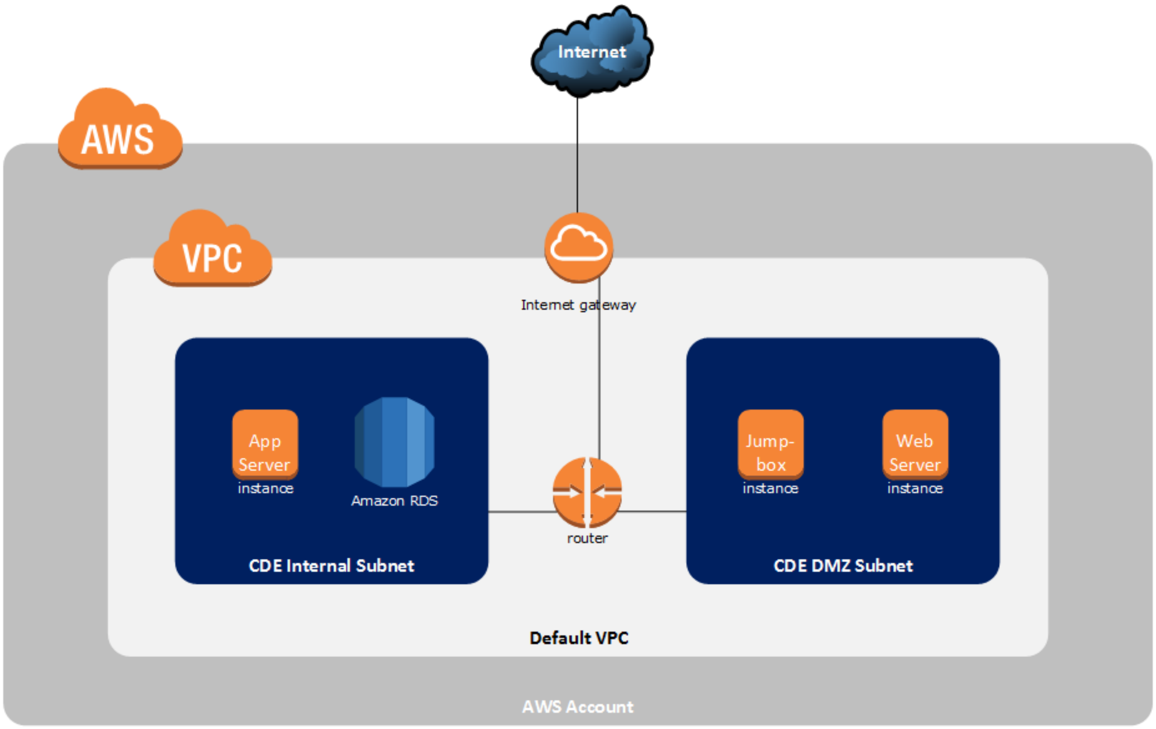


Figure 2 - Stand-alone e-commerce website architecture

4.1.1. Overview

In the Dedicated reference architecture, there are two CDE subnets in the default VPC:

- DMZ subnet
- Internal subnet

The DMZ is an Internet-facing network containing two EC2 instances, a web server and Jumpbox (used for remote management).

The Internal subnet is only accessible by the DMZ instances via Security Groups (described in detail below), and contains an application server instance and RDS.

4.1.2. PCI Scope

The CDE is comprised of all of systems in this architecture:

- Web Server
- Application Server
- RDS DB instance
- Jumpbox instance

For this scope, the web server accepts CHD, which then flows through the application tier to the DB for storage. While the Jumpbox does not touch CDE in this architecture, it is in the CDE because it resides in the same network segment as the web server. Moving it to a dedicated management network would remove it from the CDE, but not from the PCI assessment scope because it directly connects to hosts on the CDE (and is not included in this architecture).

4.1.3. Applicable AWS Services

The following AWS services help support compliance with PCI 3.2 requirements for this architecture:

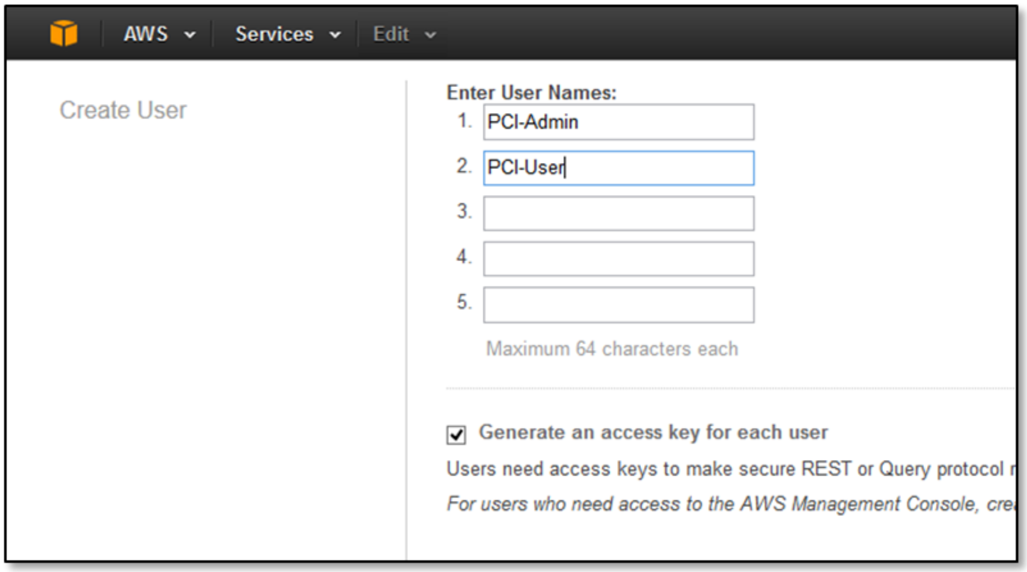
AWS Service	PCI Requirement(s) Supported
– IAM – KMS	1.3.8, 2.2.4, 3.4.1, 3.5.1, 3.6.1-5, 3.6.7, 6.4.1-2, 7.1.1-3, 7.2.1-3, 8.1.1-2, 8.2.1-6, 8.3, 8.7
– S3	3.1, 3.4.1, 10.5.1-5, 10.7
– CloudTrail – CloudWatch	10.1-3, 11.5
– EC2 – Security Groups – AMIs – EBS	1.1.4, 1.2.1, 1.3.1-3, 1.3.5-7, 2.1, 3.4.1, 4.1, 6.4.1
– RDS	3.4.1
– Config	11.5

4.1.4. Build Out

This section describes the primary steps for building out the reference architecture.

4.1.4.1. Create IAM Groups and Assign Permissions

First, define who can access and who can manage the environment. AWS requires you to explicitly define all your accounts and passwords, which ensures there are no shared defaults.



Enter User Names:

1. PCI-Admin
2. PCI-User
3.
4.
5.

Maximum 64 characters each

☑ Generate an access key for each user

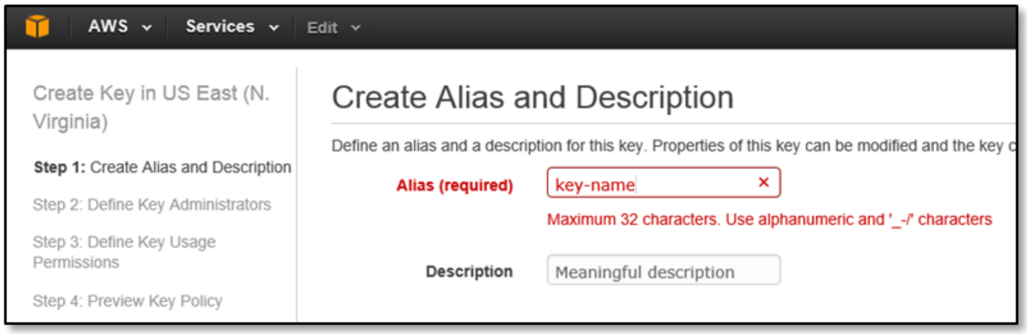
Users need access keys to make secure REST or Query protocol r

For users who need access to the AWS Management Console, cre

Figure 3 – Create users

4.1.4.2. Create Storage Encryption Keys

Use KMS to create keys for encrypting data storage locations. In this architecture, there will need to be at least one key created for the database instance that will contain cardholder data (CHD).



Create Key in US East (N. Virginia)

Step 1: Create Alias and Description

Step 2: Define Key Administrators

Step 3: Define Key Usage Permissions

Step 4: Preview Key Policy

Create Alias and Description

Define an alias and a description for this key. Properties of this key can be modified and the key c

Alias (required): key-name ×

Maximum 32 characters. Use alphanumeric and '_-/' characters

Description: Meaningful description

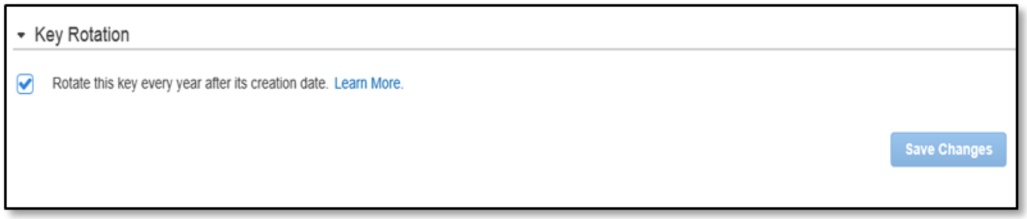
Figure 4 – KMS configuration

In AWS, you can separately assign permissions to manage an encryption key and to use the key for encryption, which allows for enforcing least-privilege.

NOTE: *While non-root volumes attached to AWS instances can also be encrypted using KMS keys, the disk encryption is transparent to the operating system running in the instance. Management of access to the encrypted disk is not separate and independent from the operating system, as required by PCI Requirement 3.4.1.*

It is best practice, and a PCI requirement (Req. 3.6.4), to change encryption keys used to protect data on a regular basis. This limits how long a compromised key is usable.

After creating the key, click on its URL in the Encryption Keys section of the Identity and Access Management AWS service. The Key Rotation setting is in the listed properties for the selected key. This allows you to automate key rotation within a designated cryptoperiod in alignment with Requirement 3.6.4.



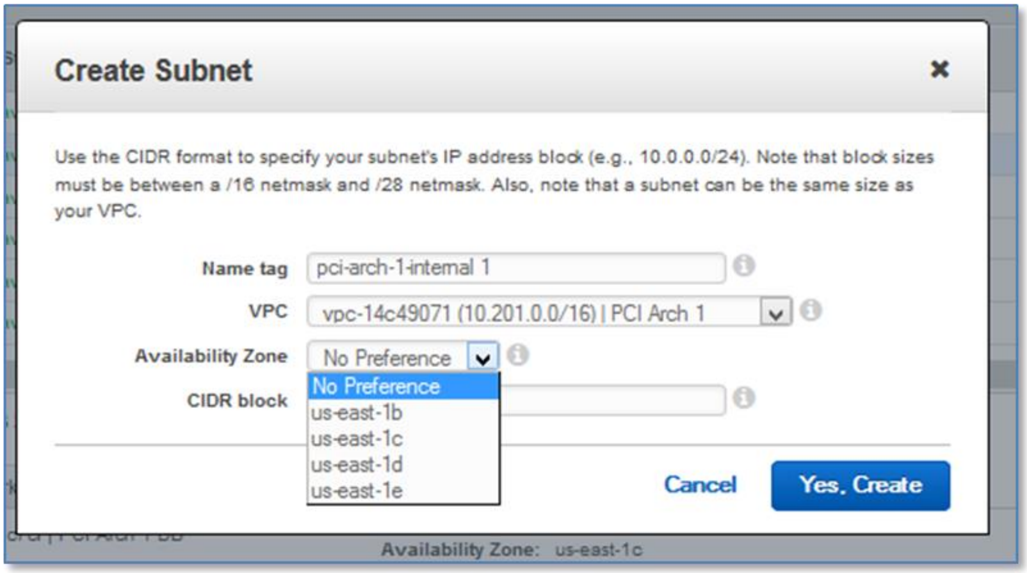
Key Rotation

☑ Rotate this key every year after its creation date. Learn More.

Figure 5 – Key rotation option

4.1.4.3. Create Subnets

For this architecture, you need at least three separate subnets (a DMZ subnet for the web servers and management instances, and two protected subnets for application and database systems). These also need be in different availability zones.



Create Subnet

Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC.

Name tag: pci-arch-1-internal 1

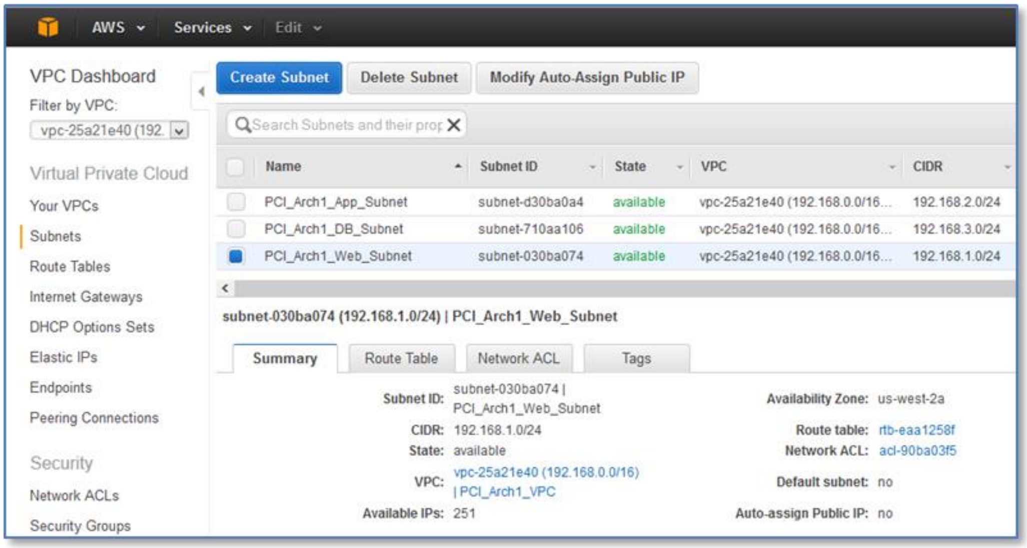
VPC: vpc-14c49071 (10.201.0.0/16) | PCI Arch 1

Availability Zone: No Preference

CIDR block:

Figure 6 – Configuring a subnet

Use the VPC service management page in the AWS console to configure subnets, even for the Default VPC EC2 Classic uses.



	Name	Subnet ID	State	VPC	CIDR
□	PCI_Arch1_App_Subnet	subnet-d30ba0a4	available	vpc-25a21e40 (192.168.0.0/16...	192.168.2.0/24
□	PCI_Arch1_DB_Subnet	subnet-710aa106	available	vpc-25a21e40 (192.168.0.0/16...	192.168.3.0/24
■	PCI_Arch1_Web_Subnet	subnet-030ba074	available	vpc-25a21e40 (192.168.0.0/16...	192.168.1.0/24

subnet-030ba074 (192.168.1.0/24) | PCI_Arch1_Web_Subnet

Subnet ID: subnet-030ba074 | PCI_Arch1_Web_Subnet

CIDR: 192.168.1.0/24

State: available

VPC: vpc-25a21e40 (192.168.0.0/16) | PCI_Arch1_VPC

Available IPs: 251

Availability Zone: us-west-2a

Route table: rtb-eaa1258f

Network ACL: acl-90ba03f5

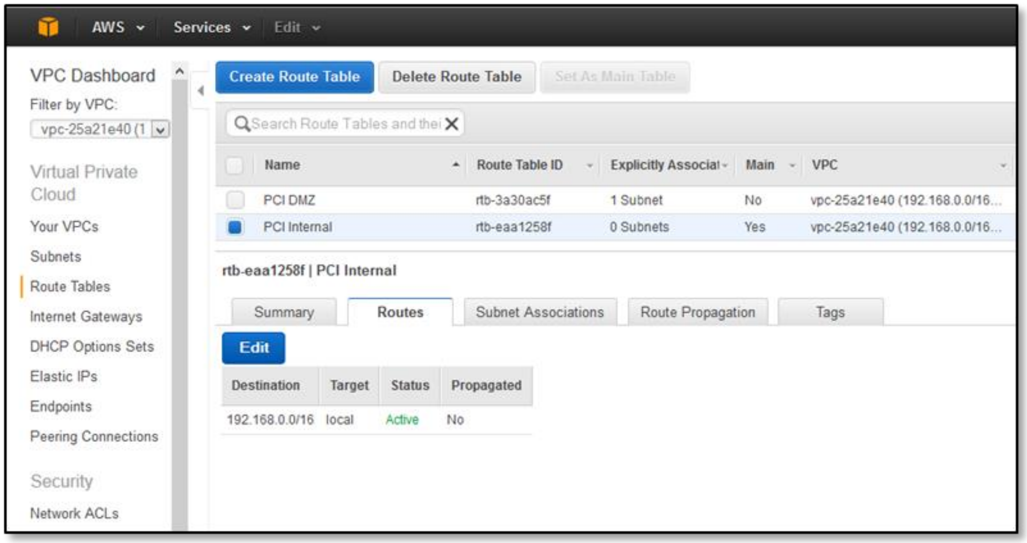
Default subnet: no

Auto-assign Public IP: no

Figure 7 - VPC Service Management page with three new subnets

4.1.4.4. Configure Routing

When creating a new subnet, it will use the main route table for the VPC that will include one route allowing internal traffic for that subnet only.



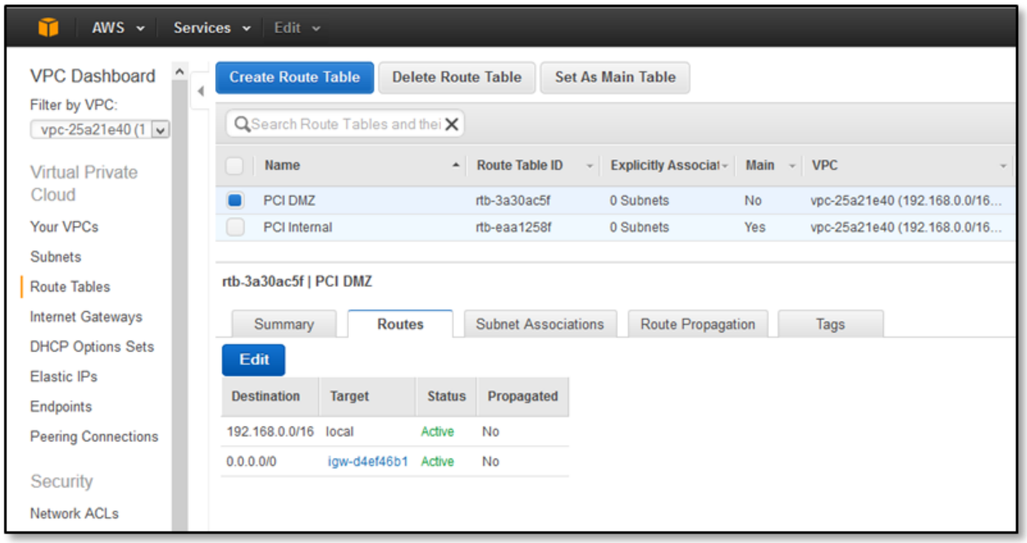
	Name	Route Table ID	Explicitly Associat	Main	VPC
□	PCI DMZ	rtb-3a30ac5f	1 Subnet	No	vpc-25a21e40 (192.168.0.0/16...
■	PCI Internal	rtb-eaa1258f	0 Subnets	Yes	vpc-25a21e40 (192.168.0.0/16...

rtb-eaa1258f | PCI Internal

Destination	Target	Status	Propagated
192.168.0.0/16	local	Active	No

Figure 8 - Internal route

Only the DMZ subnet needs to have a route to the Internet Gateway. This ensures only instances in the DMZ subnet support direct inbound or outbound Internet connections.



	Name	Route Table ID	Explicitly Associat	Main	VPC
■	PCI DMZ	rtb-3a30ac5f	0 Subnets	No	vpc-25a21e40 (192.168.0.0/16...
□	PCI Internal	rtb-eaa1258f	0 Subnets	Yes	vpc-25a21e40 (192.168.0.0/16...

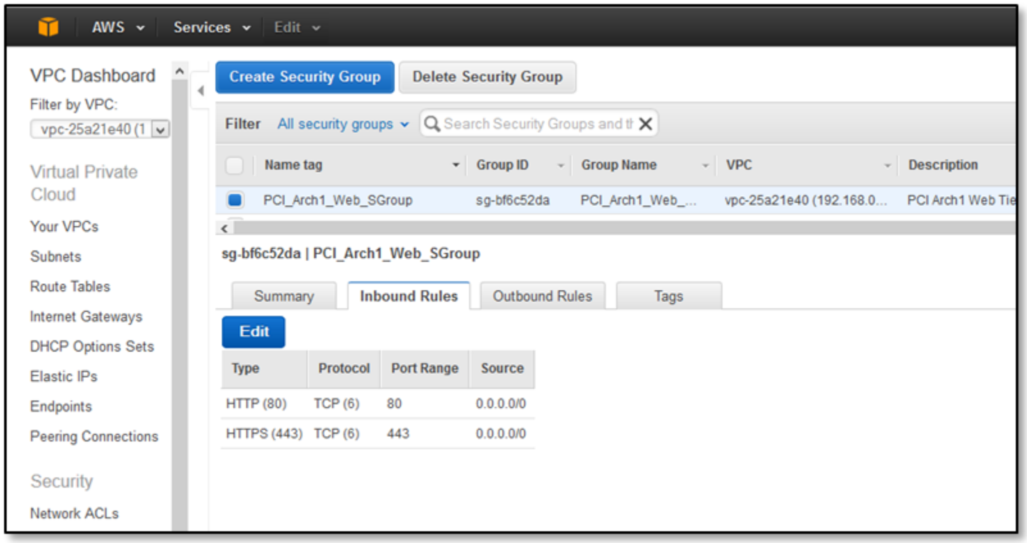
rtb-3a30ac5f | PCI DMZ

Destination	Target	Status	Propagated
192.168.0.0/16	local	Active	No
0.0.0.0/0	igw-d4ef46b1	Active	No

Figure 9 - Gateway route for DMZ

4.1.4.5. Create Security Groups

Security groups function like an inbound firewall. They restrict incoming instance network access to predefined sources, IP protocols, and TCP or UDP ports.



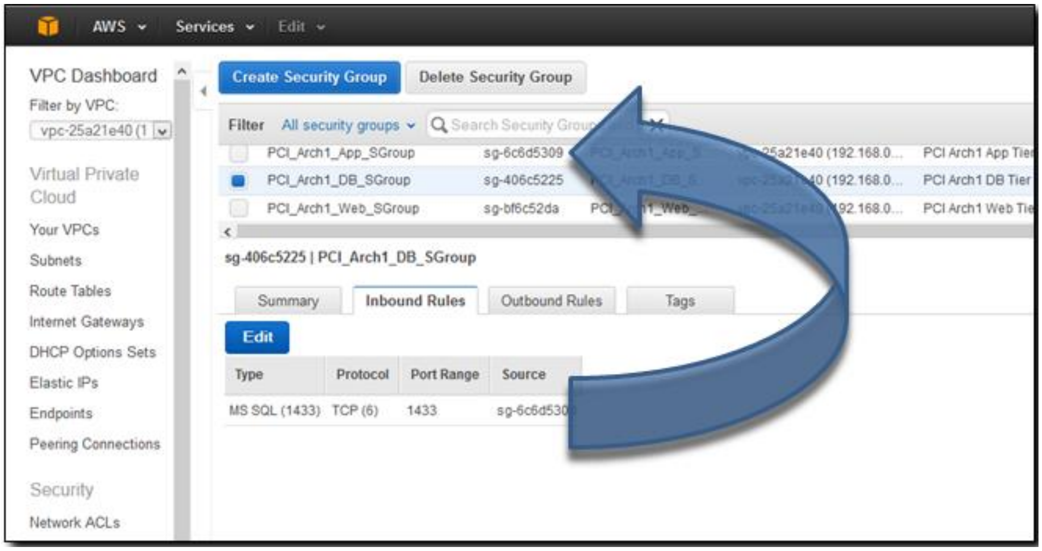
Name tag	Group ID	Group Name	VPC	Description
PCI_Arch1_Web_SGroup	sg-bf6c52da	PCI_Arch1_Web_...	vpc-25a21e40 (192.168.0....	PCI Arch1 Web Tie

sg-bf6c52da | PCI_Arch1_Web_SGroup

Type	Protocol	Port Range	Source
HTTP (80)	TCP (6)	80	0.0.0.0/0
HTTPS (443)	TCP (6)	443	0.0.0.0/0

Figure 10 - Security Group Rules

They can also reference other Security Group(s) in the same VPC as allowed sources. For example, you can reference the application servers Security Group to restrict access to a Microsoft SQL Server to only the instances in that group.



Name tag	Group ID	Group Name	VPC	Description
PCI_Arch1_App_SGroup	sg-6c6d5309	PCI_Arch1_App_S	25a21e40 (192.168.0....	PCI Arch1 App Tier
PCI_Arch1_DB_SGroup	sg-406c5225	PCI_Arch1_DB_S	vpc-25a21e40 (192.168.0....	PCI Arch1 DB Tier
PCI_Arch1_Web_SGroup	sg-bf6c52da	PCI_Arch1_Web_	vpc-25a21e40 (192.168.0....	PCI Arch1 Web Tie

sg-406c5225 | PCI_Arch1_DB_SGroup

Type	Protocol	Port Range	Source
MS SQL (1433)	TCP (6)	1433	sg-6c6d530

Figure 11 - Security Groups are stateful and block all access not explicitly allowed

This architecture uses five Security Groups to accomplish the architecture depicted:

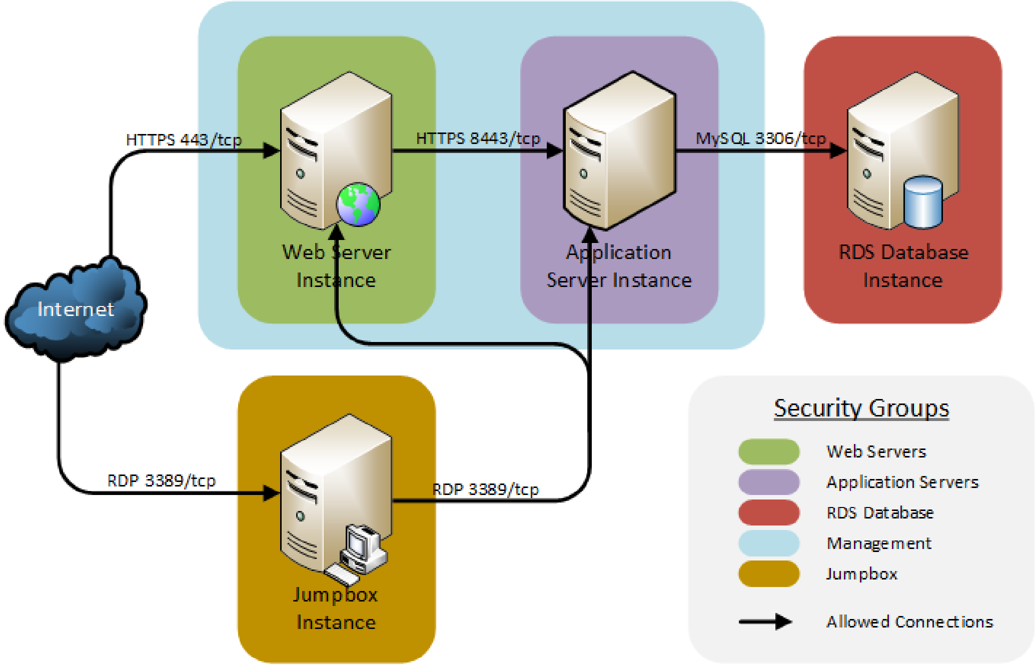
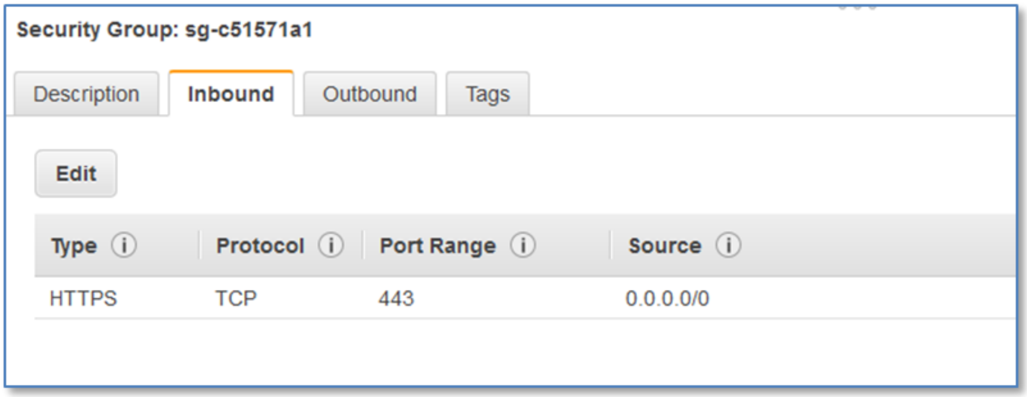


Figure 12 - Logical Firewall / Security Group Design

Web Server Security Group

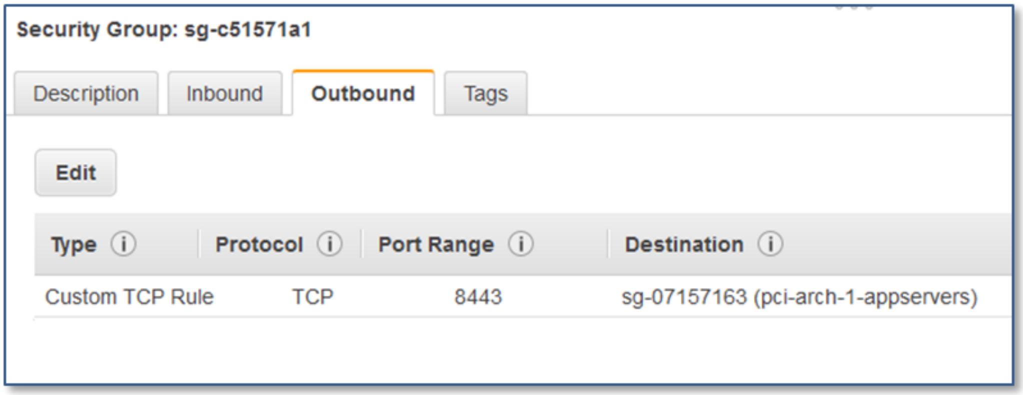
This group allows inbound web client connections from anywhere and outbound web services connections to internal application servers.



Security Group: sg-c51571a1

Type	Protocol	Port Range	Source
HTTPS	TCP	443	0.0.0.0/0

Figure 13 - Web Server Security Group Inbound Rules



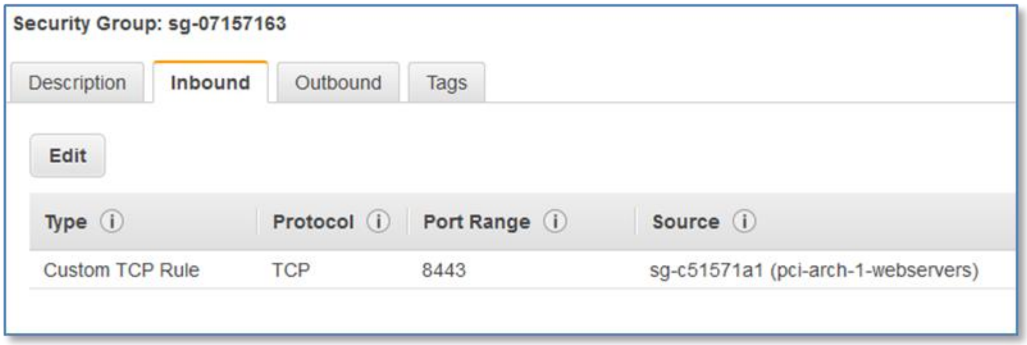
Security Group: sg-c51571a1

Type	Protocol	Port Range	Destination
Custom TCP Rule	TCP	8443	sg-07157163 (pci-arch-1-appservers)

Figure 14 - Web Server Security Group Outbound Rules

Application Server Security Group

This Security Group allows incoming web service connections from the web servers to the application servers.

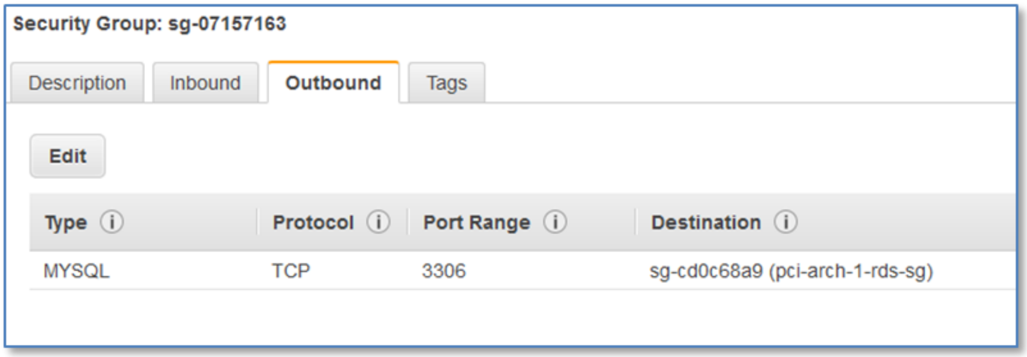


Security Group: sg-07157163

Type	Protocol	Port Range	Source
Custom TCP Rule	TCP	8443	sg-c51571a1 (pci-arch-1-webservers)

Figure 15 - App Server Security Group Inbound Rules

The group allows outbound MySQL connections to the RDS database instances.



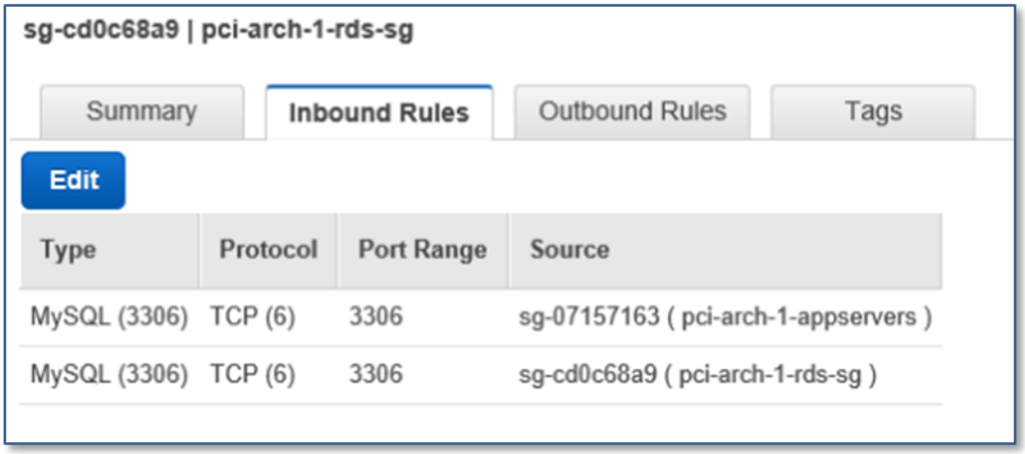
Security Group: sg-07157163

Type	Protocol	Port Range	Destination
MYSQL	TCP	3306	sg-cd0c68a9 (pci-arch-1-rds-sg)

Figure 16 - Application server security group outbound rules

Database (RDS) Security Group

Security Groups secure network access to RDS instances, although only inbound rules are used. These rules allow MySQL connections from the application servers and from other RDS instances in the group, to support database replication.

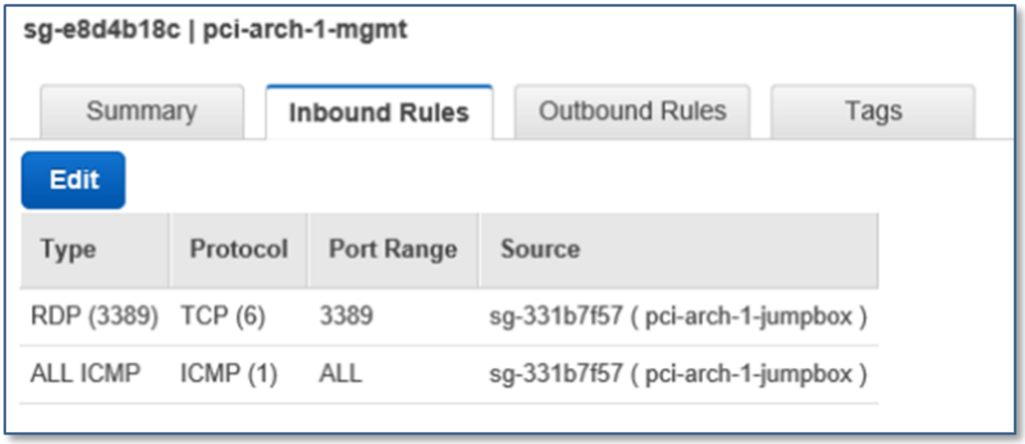


Type	Protocol	Port Range	Source
MySQL (3306)	TCP (6)	3306	sg-07157163 (pci-arch-1-appservers)
MySQL (3306)	TCP (6)	3306	sg-cd0c68a9 (pci-arch-1-rds-sg)

Figure 17 - DB Server Security Group Inbound Rules

Management Security Group

The Management Security Group is a special group that allows connections from the Jumpbox to all instances, for management purposes.



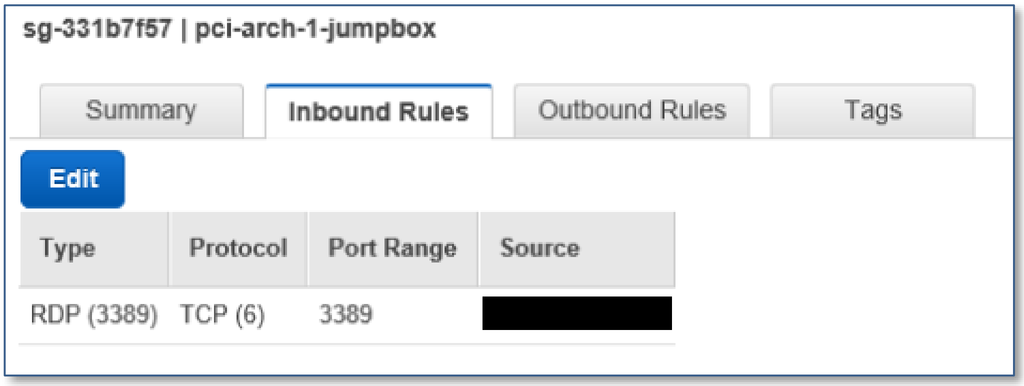
Type	Protocol	Port Range	Source
RDP (3389)	TCP (6)	3389	sg-331b7f57 (pci-arch-1-jumpbox)
ALL ICMP	ICMP (1)	ALL	sg-331b7f57 (pci-arch-1-jumpbox)

Figure 18 - Management Server Security Group Inbound Rules

Jump Box Security Group

The Jumpbox itself only needs to allow connections from the public IP address of your company's network.

NOTE: *You must implement two-factor authentication for remote access to meet Requirement 8.3. There are numerous third-party products that can support this for a Windows or Linux system. AWS does not natively support two-factor authentication for remote access to EC2 instances. However, AWS does support multi-factor authentication to EC2. For more information, see AWS MFA details and pricing at http://aws.amazon.com/iam/details/mfa/.*

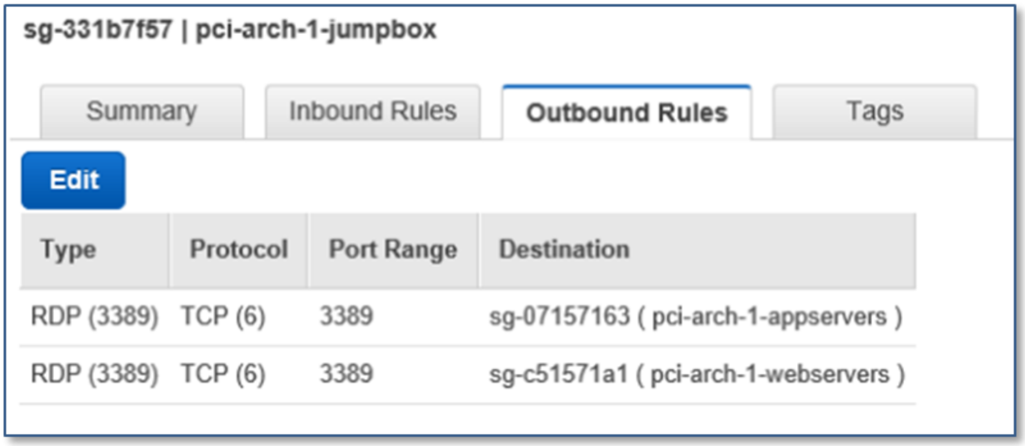


sg-331b7f57 | pci-arch-1-jumpbox

Type	Protocol	Port Range	Source
RDP (3389)	TCP (6)	3389	[redacted]

Figure 19 - Jumpbox Server Security Group Inbound Rules

The Jumpbox will need outbound communication to the web and application servers for management purposes.



sg-331b7f57 | pci-arch-1-jumpbox

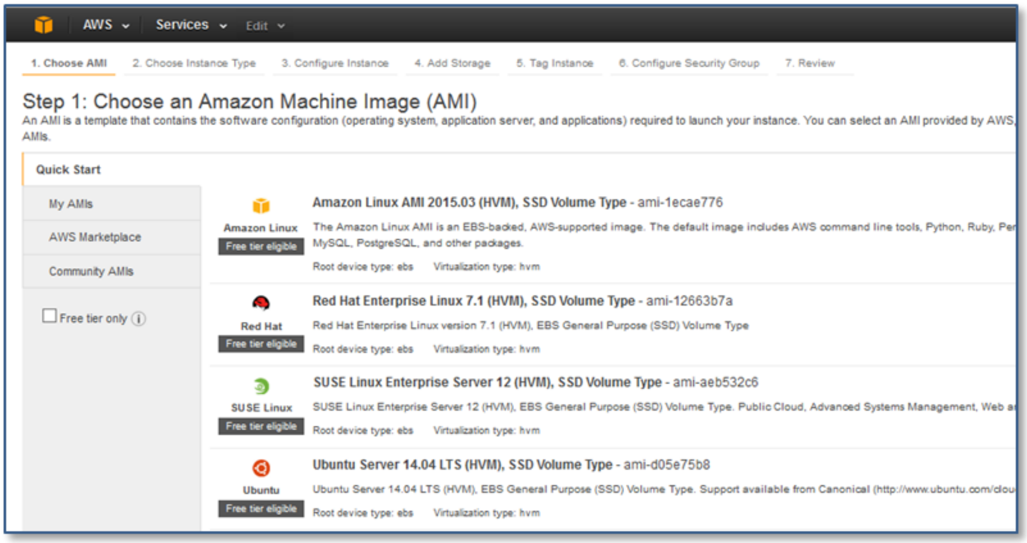
Type	Protocol	Port Range	Destination
RDP (3389)	TCP (6)	3389	sg-07157163 (pci-arch-1-appservers)
RDP (3389)	TCP (6)	3389	sg-c51571a1 (pci-arch-1-webservers)

Figure 20 - Jumpbox Server Security Group Outbound Rules

4.1.4.6. Create Hardened AMIs from secured instances

PCI requires the development of secure configuration standards for all system components. AWS allows for the creation of one secure instance that will serve as a template for the creation of pre-secured systems.

Launch a new instance for each of the types needed for deployment. For this architecture, a web server and an application server are required (the database will use the AWS RDS service).



Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, AMIs.

Amazon Linux AMI 2015.03 (HVM), SSD Volume Type - ami-1ecae776

The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Per MySQL, PostgreSQL, and other packages.

Root device type: ebs Virtualization type: hvm

Red Hat Enterprise Linux 7.1 (HVM), SSD Volume Type - ami-12663b7a

Red Hat Enterprise Linux version 7.1 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm

SUSE Linux Enterprise Server 12 (HVM), SSD Volume Type - ami-aeb532c6

SUSE Linux Enterprise Server 12 (HVM), EBS General Purpose (SSD) Volume Type. Public Cloud, Advanced Systems Management, Web a

Root device type: ebs Virtualization type: hvm

Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-d05e75b8

Ubuntu Server 14.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/clou

Root device type: ebs Virtualization type: hvm

Figure 21 - Choosing AMI for deploying instances

Make sure the instance is part of the Jumpbox Security Group, is in the DMZ subnet, and has an Elastic IP or a public IP, so that you can remotely connect to and manage it.

Public IPs are provisioned by AWS when instance is launched. This is automatic in the Default VPC, but configurable by subnet in other VPCs (see the Amazon VPC User Guide for further information).

Elastic IPs (EIPs) are managed by a customer and associated with the AWS account, not a specific instance. You can reassign which instance uses a specific EIP without the address changing.

Connect to and harden your instance. Make sure you document all steps taken to secure the instance, as your assessor will need to review these.

NOTE: *Alternately, you can use Anitian's pre-hardened AMIs, which include a security configuration standard document. These are available in the AWS Marketplace.*

Once you are finished, and the instance is ready, power it off and create a custom AMI from it.

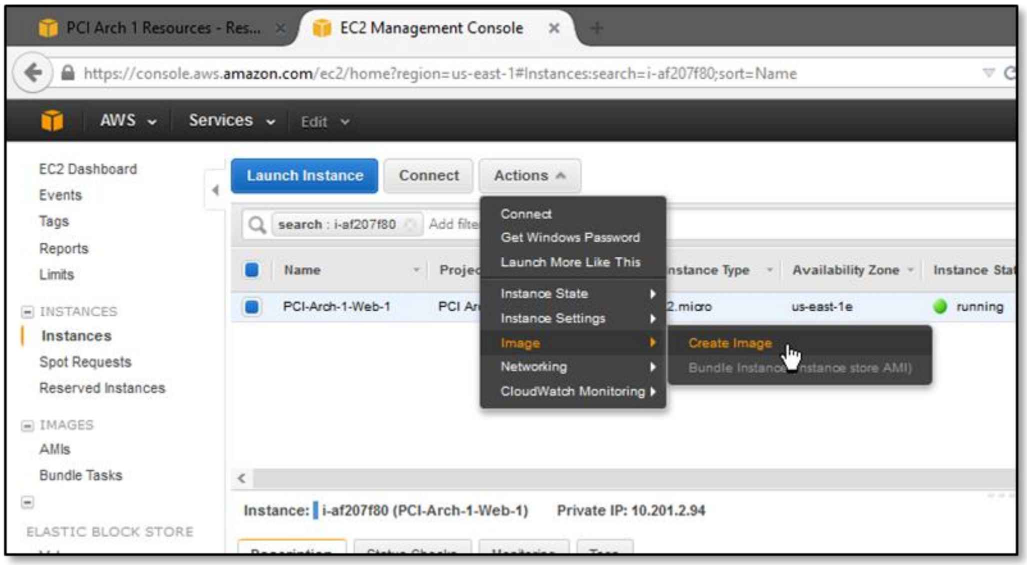


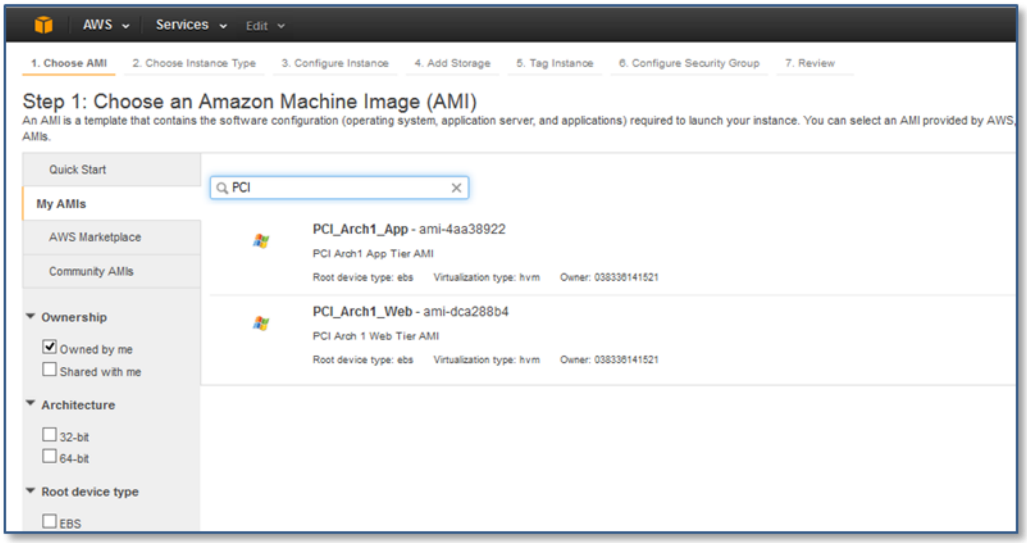
Figure 22 - Creating AMI from hardened instance

4.1.4.7. Launch Instances from Hardened AMI

This architecture needs a minimum of three instances:

- Jumpbox instance
  - Used to manage the environment remotely
  - In the DMZ subnet, it will need an EIP or a public IP
- Web server instance
  - Front-end to the e-commerce application
  - In the DMZ subnet, it will need an EIP or a public IP
- Application server instance
  - App tier running middleware that brokers connectivity between web servers and DB (RDS in this example)
  - In the internal subnet, it is only accessible by the web server, and is the only instance with access to the DB

Launch the EC2 instances from the newly created AMIs.



Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, AMIs.

PCI

PCI_Arch1_App - ami-4aa38922
PCI Arch1 App Tier AMI
Root device type: ebs Virtualization type: hvm Owner: 038336141521

PCI_Arch1_Web - ami-dca288b4
PCI Arch 1 Web Tier AMI
Root device type: ebs Virtualization type: hvm Owner: 038336141521

Ownership
☑ Owned by me
☐ Shared with me

Architecture
☐ 32-bit
☐ 64-bit

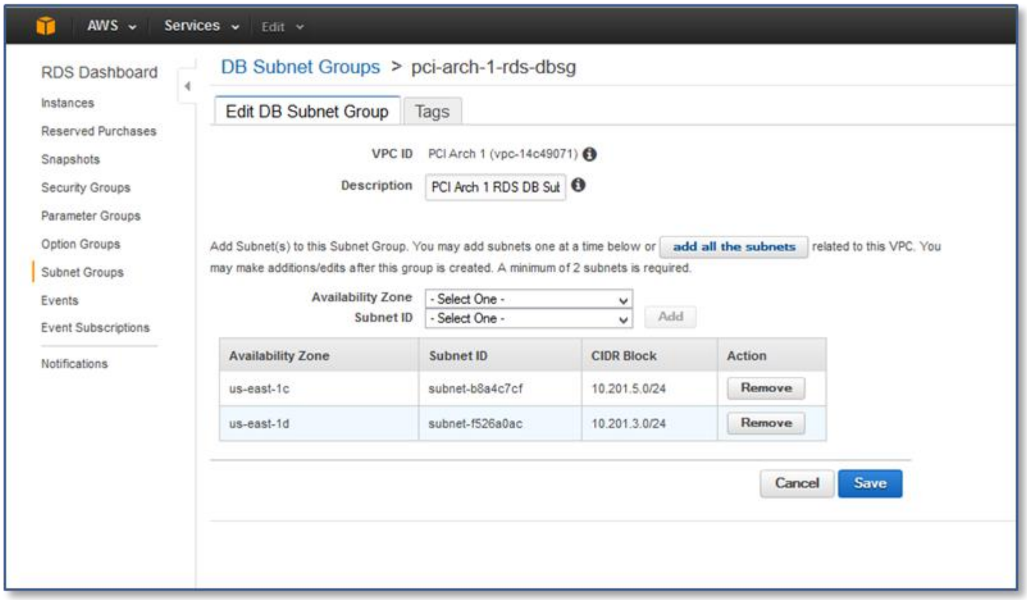
Root device type
☐ EBS

Figure 23 - Launching instances from AMIs

4.1.4.8. Create Subnet Group

Subnet Groups allow RDS to determine where redundant instances need to be located to survive the failure of the primary instance.

Manage Subnet Groups from the RDS service management page. Create one that contains the two internal subnets created earlier.



DB Subnet Groups > pci-arch-1-rds-dbsg

VPC ID PCI Arch 1 (vpc-14c49071)

Description PCI Arch 1 RDS DB Sut

Add Subnet(s) to this Subnet Group. You may add subnets one at a time below or add all the subnets related to this VPC. You may make additions/edits after this group is created. A minimum of 2 subnets is required.

Availability Zone - Select One -
Subnet ID - Select One -

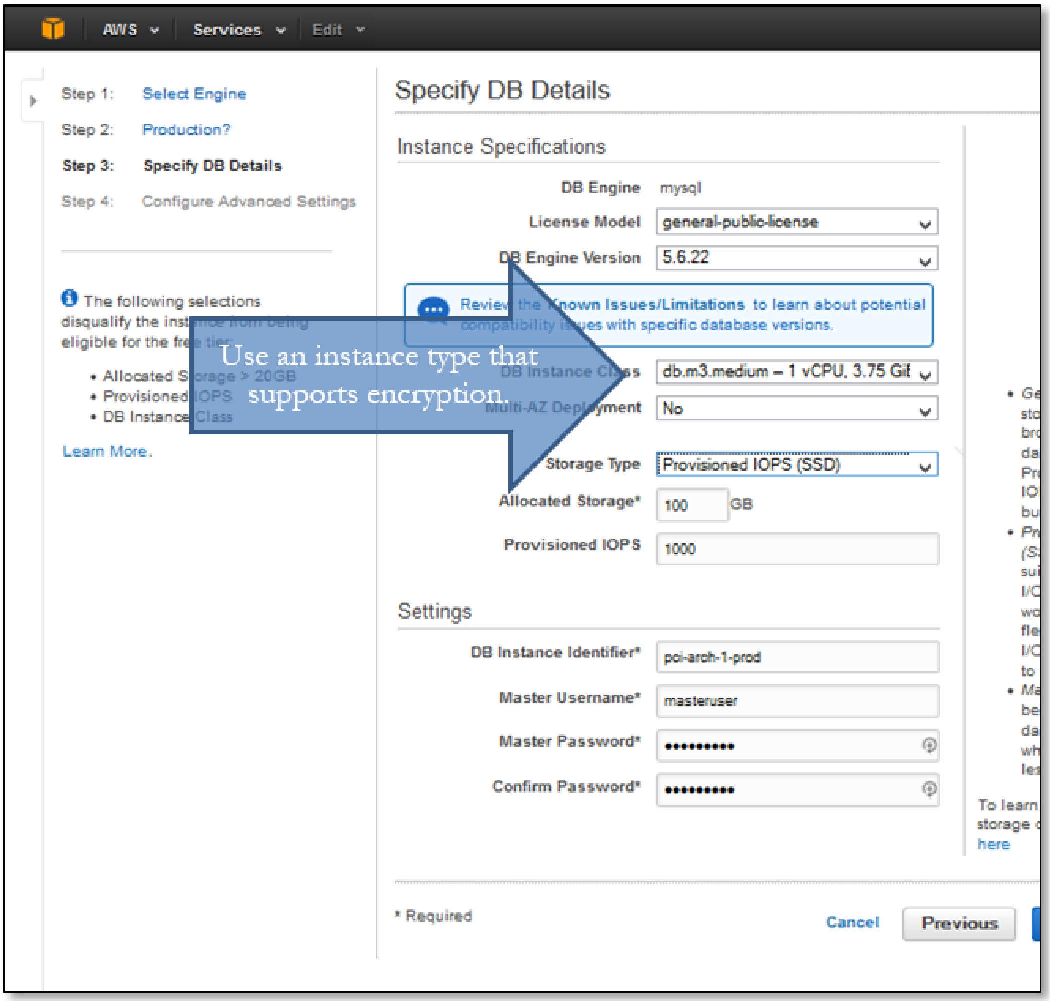
Availability Zone	Subnet ID	CIDR Block	Action
us-east-1c	subnet-b8a4c7cf	10.201.5.0/24	Remove
us-east-1d	subnet-f526a0ac	10.201.3.0/24	Remove

Figure 24 - Creating RDS subnets

4.1.4.9. Create Encrypted RDS Instance

Using the KMS key and Subnet Group already created, launch the encrypted RDS instance that will store CHD.

When creating the RDS instance, a few settings need special attention in order to meet PCI requirements. The DB instance Class needs to be db.m3.medium or higher to support encryption.



Step 1: Select Engine
Step 2: Production?
Step 3: Specify DB Details
Step 4: Configure Advanced Settings

The following selections disqualify the instance from being eligible for the free tier:

- Allocated Storage > 20GB
- Provisioned IOPS
- DB Instance Class

Learn More.

Specify DB Details

Instance Specifications

DB Engine	mysql
License Model	general-public-license
DB Engine Version	5.6.22

Review the Known Issues/Limitations to learn about potential compatibility issues with specific database versions.

DB Instance Class	db.m3.medium – 1 vCPU, 3.75 GiB
Multi-AZ Deployment	No
Storage Type	Provisioned IOPS (SSD)
Allocated Storage*	100 GB
Provisioned IOPS	1000

Settings

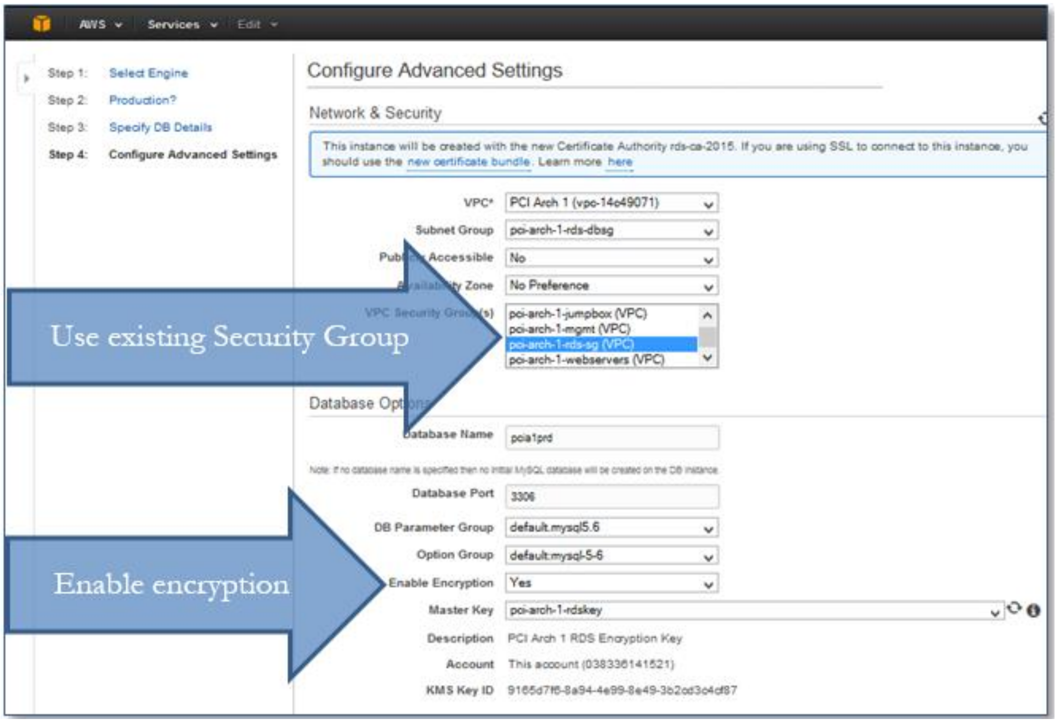
DB Instance Identifier*	pci-arch-1-prod
Master Username*	masteruser
Master Password*	•••••••••
Confirm Password*	•••••••••

* Required

Figure 25 - Selecting instance class that supports encryption

Additionally, ensure that you:

- Select the existing Security Group to ensure AWS does not create a new one with default settings.
- Select "Yes" for Enable Encryption, and choose the KMS key created earlier.



Step 1: Select Engine
Step 2: Production?
Step 3: Specify DB Details
Step 4: Configure Advanced Settings

Configure Advanced Settings

Network & Security

This instance will be created with the new Certificate Authority rds-ca-2015. If you are using SSL to connect to this instance, you should use the new certificate bundle. Learn more here

VPC*: PCI Arch 1 (vpc-14c49071)
Subnet Group: pci-arch-1-rds-dbsg
Publicly Accessible: No
Availability Zone: No Preference
VPC Security Group(s): pci-arch-1-jumpbox (VPC), pci-arch-1-mgmt (VPC), pci-arch-1-rds-sg (VPC), pci-arch-1-webservers (VPC)

Use existing Security Group

Database Options

Database Name: pcia1prd

Note: If no database name is specified then no initial MySQL database will be created on the DB instance.

Database Port: 3306
DB Parameter Group: default.mysql5.6
Option Group: default:mysql-5-6
Enable Encryption: Yes
Master Key: pci-arch-1-rdskey
Description: PCI Arch 1 RDS Encryption Key
Account: This account (038336141521)
KMS Key ID: 9165d7f6-8a94-4e99-8e49-3b2cd3c4cf87

Enable encryption

Figure 26 - Selecting Security Group and key for enabling encryption

4.1.4.10. Install application software

Once the RDS DB instance is finished provisioning, the environment is ready.

NOTE: *The steps in this build focus on leveraging the AWS services for compliance. Numerous additional PCI requirements will need to be addressed in order to ensure this environment is compliant, including anti-virus, patch management, log management, vulnerability management, and file integrity monitoring.*

4.2. Architecture 2: Segmented

This architecture builds upon the previous design. It demonstrates an e-commerce website segmented from other systems in an existing Amazon AWS environment.

Segmenting the CDE systems from the rest of an AWS account limits the scope of PCI compliance.

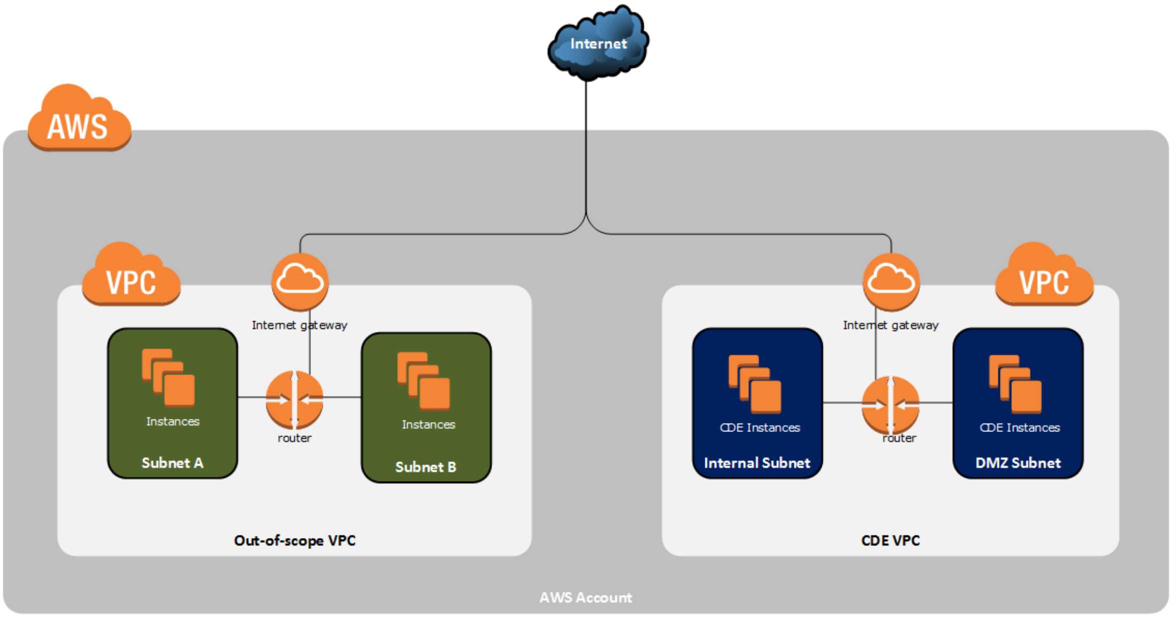


Figure 27 - Architecture of segmented CDE within broader AWS environment

4.2.1. Overview

In the Segmented reference architecture, there are two VPCs each with two subnets:

- CDE VPC: Contains CDE DMZ subnet and Internal CDE subnet
- Out-of-scope VPC: Contains two subnets segmented from the CDE.

The systems and subnets in the CDE VPC are the same as those in the first architecture. The new VPC represents additional system subnets in AWS that do not require connectivity to CDE systems. This architecture demonstrates how in remove the new VPC from the PCI assessment scope through segmentation.

To segment the out-of scope networks, they must not connect to the CDE. This is typically accomplished using firewall policies.

In AWS, Security Groups can control traffic into and out of CDE instances, but they do not meet the routing configuration requirements in PCI Requirement 1.2. Similarly, you cannot use the default VPC that EC2 creates, because all instances are part of the same supernet.

However, Virtual Private Clouds (VPCs) use their own private addresses, and are networks isolated from the rest of an AWS account. They provide the most direct way to implement true network segmentation for PCI scope reduction.

NOTE: *In order to ensure that your CDE VPC is segmented from your out-of-scope VPC, you must not implement VPC peering between them. This would bring the non-CDE VPC into your assessment scope (which is an appropriate strategy in some circumstances, but not used in this example).*

4.2.2. PCI Scope

The CDE is comprised of the following instances in this architecture, all contained within the CDE VPC:

- Web server
- App server
- RDS DB

The new VPC is out-of-scope for PCI due to network segmentation, as discussed below.

4.2.3. Applicable AWS Services

The following additional AWS service helps to support compliance with PCI requirements for this architecture:

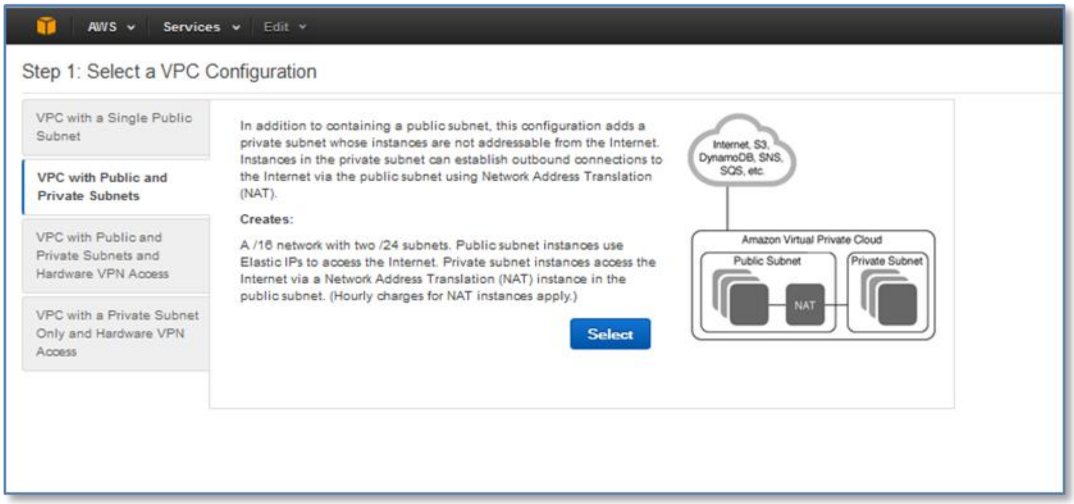
AWS Service	PCI Requirement Supported
– VPC	1.1.4, 1.2.1, 1.3.1-3, 1.3.5-7, 4.1, 6.4.1

4.2.4. Build Out

This section describes the primary steps to build out the reference architecture.

4.2.4.1. Create a VPC

Create a new VPC to segment this out-of-scope VPC from the CDE VPC created in the first architecture.


AWS
Services
Edit
VPC with a Single Public Subnet
VPC with Public and Private Subnets
VPC with Public and Private Subnets and Hardware VPN Access
VPC with a Private Subnet Only and Hardware VPN Access
Select
Internet, S3, DynamoDB, SNS, SQS, etc
Amazon Virtual Private Cloud
Public Subnet
Private Subnet
NAT

Step 1: Select a VPC Configuration

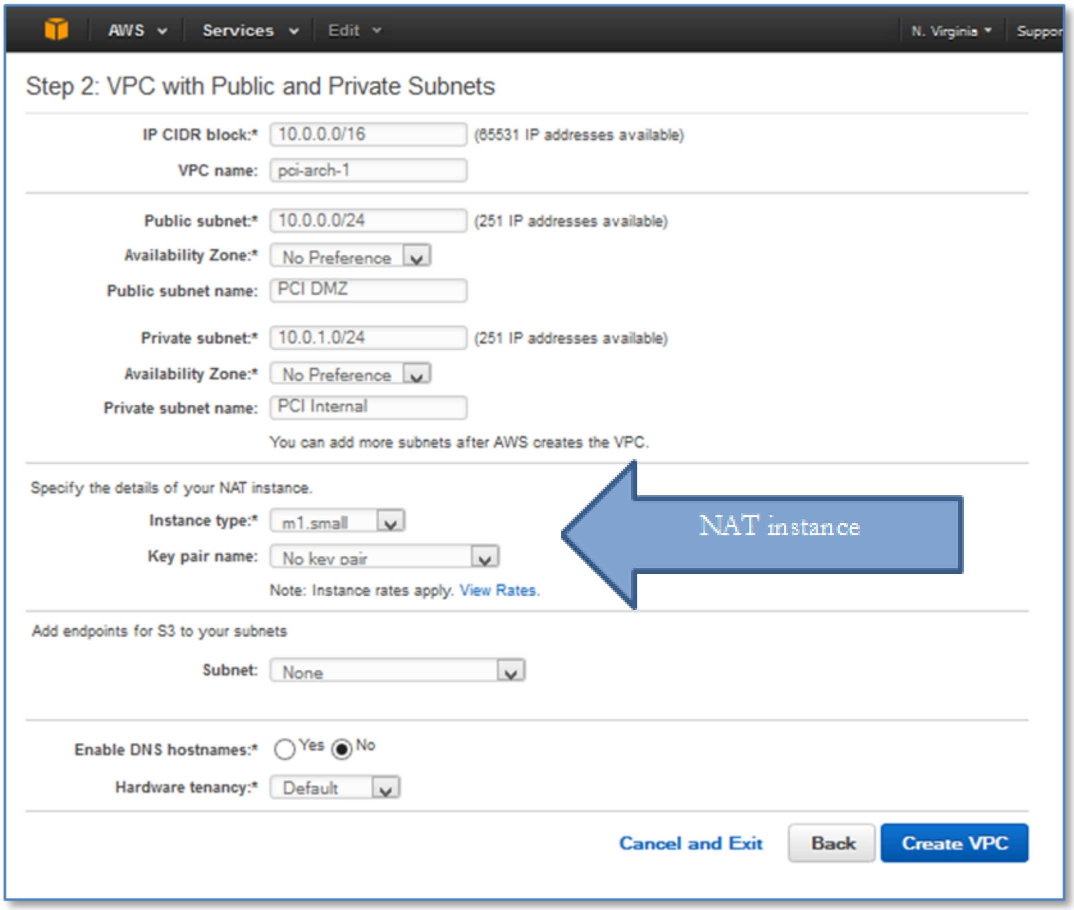
In addition to containing a public subnet, this configuration adds a private subnet whose instances are not addressable from the Internet. Instances in the private subnet can establish outbound connections to the Internet via the public subnet using Network Address Translation (NAT).

Creates:

A /16 network with two /24 subnets. Public subnet instances use Elastic IPs to access the Internet. Private subnet instances access the Internet via a Network Address Translation (NAT) instance in the public subnet. (Hourly charges for NAT instances apply.)

Figure 28 – Creating a new VPC

The new VPC includes a NAT instance designed to work like an Internet gateway router for the private subnet. You can delete it to prevent Instances in the internal subnet from accessing the Internet.



Step 2: VPC with Public and Private Subnets

IP CIDR block:* 10.0.0.0/16 (65531 IP addresses available)
VPC name: pci-arch-1

Public subnet:* 10.0.0.0/24 (251 IP addresses available)
Availability Zone:* No Preference
Public subnet name: PCI DMZ

Private subnet:* 10.0.1.0/24 (251 IP addresses available)
Availability Zone:* No Preference
Private subnet name: PCI Internal
You can add more subnets after AWS creates the VPC.

Specify the details of your NAT instance.
Instance type:* m1.small
Key pair name: No key pair
Note: Instance rates apply. View Rates.

Add endpoints for S3 to your subnets
Subnet: None

Enable DNS hostnames:* ○ Yes ● No
Hardware tenancy:* Default

Figure 29 – Configure the new VPC

4.2.4.2. Create IAM Users, Groups and KMS Keys

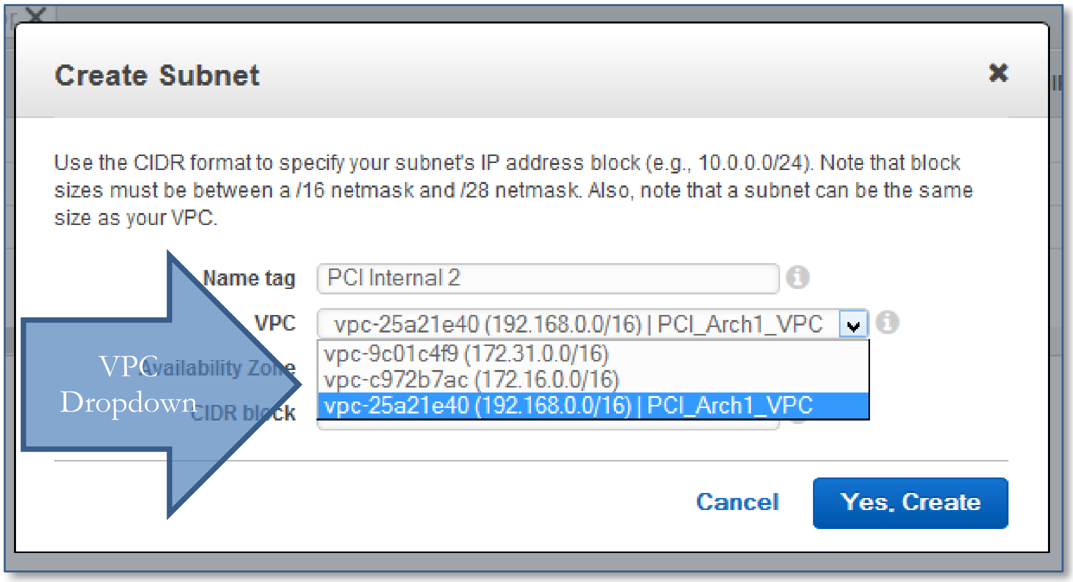
IAM resources are not region or VPC specific. All resources within an AWS account share the same IAM resources.

Create these as outlined above in Sections 4.1.4.1 and 4.1.4.2.

4.2.4.3. Create Resource in the VPC

When creating resources, be sure to select the newly created VPC in the "VPC" dropdown of each resource creation wizard.

NOTE: *Not all AWS resources are specific to a single VPC or region. If a resource cannot be found on the VPC Dashboard, try looking in the EC2 service management console.*



Create Subnet

Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC.

Name tag: PCI Internal 2

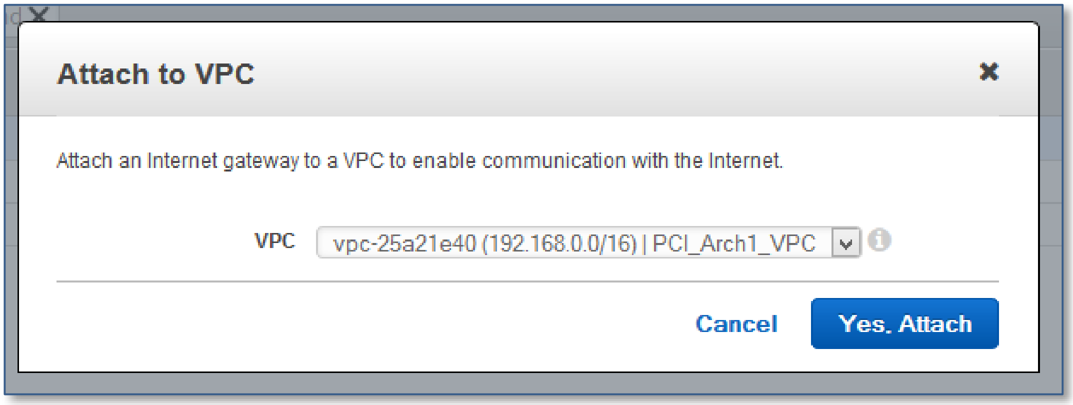
VPC: vpc-25a21e40 (192.168.0.0/16) | PCI_Arch1_VPC

×
vpc-9c01c4f9 (172.31.0.0/16)
vpc-c972b7ac (172.16.0.0/16)
vpc-25a21e40 (192.168.0.0/16) | PCI_Arch1_VPC
Availability Zone
CIDR block
VPC Dropdown
Cancel
Yes, Create

Figure 30 – Create VPC subnet

4.2.4.4. Internet Access

The VPC will need its own Internet gateway. Create the gateway, and then attach it to the VPC.



Attach to VPC

Attach an Internet gateway to a VPC to enable communication with the Internet.

VPC: vpc-25a21e40 (192.168.0.0/16) | PCI_Arch1_VPC

Figure 31 – Attaching an Internet Gateway to a VPC

4.3. Architecture 3: Connected

This architecture represents connecting an on-premise CDE into an Amazon AWS environment.

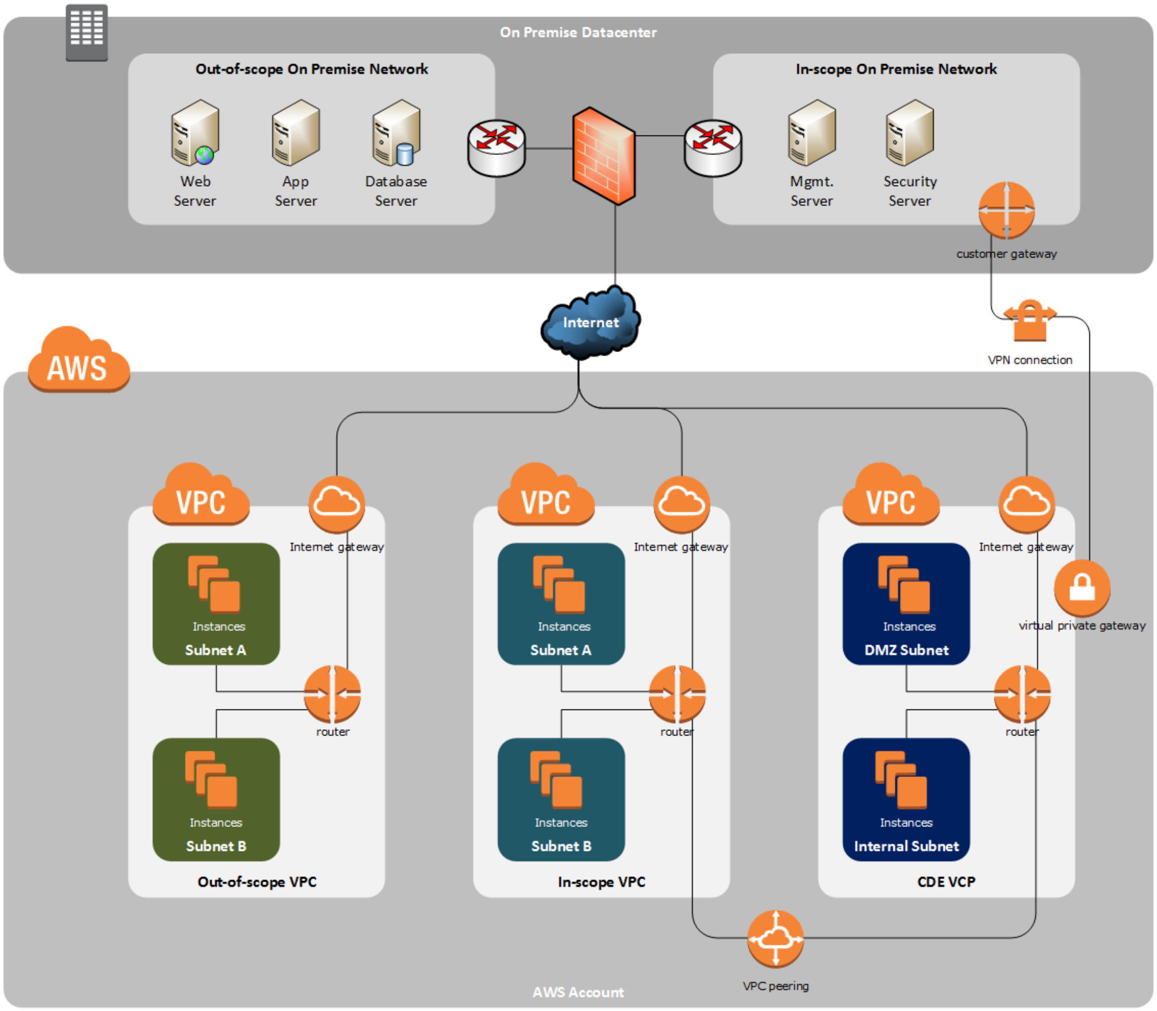


Figure 32 – Connected on-premise systems into AWS CDE

4.3.1. Overview

In the Connected reference architecture, there are five different Network Segments:

1. **CDE VPC**

   This is the VPC from Architecture 1, with a DMZ and internal CDE subnet.

2. **Out-of-scope VPC**

   This is the new VPC from Architecture 2, fully segmented with no VPC peering.

3. **In-Scope VPC**

   This is a new VPC. It is connected to the CDE via VPC peering.

4. **In-Scope On-Premise Network**

This is a customer network segment connected to the AWS CDE via a VPN.

5. **Out-of-scope On-Premise Network**

   This is a customer network segmented from the in-scope customer network and AWS.

Extending an on-premise network to a VPC in AWS is no different than setting up other business-to-business VPN connections.

It is possible to set up a direct, private, non-VPN connection into AWS using the Direct Connect service. Direct connect supports high bandwidth links and can be combined with 802.1q VLAN tagging to support logical segmentation. As Section 3.4 above notes, connections using Direct Connect are not encrypted. Additonal controls like VPNs may still be necessary to comply with PCI 4.1 if the connection is not private.

4.3.2. PCI Scope

The PCI assessment scope in this reference architecture consists of:

- The CDE VPC (web, app, and database tiers)
- The In-Scope VPC in AWS
- The In-Scope On-Premise Network

The two in-scope networks do not have CHD, but are connected to the CDE. This architecture demonstrates two common uses cases for connected in-scope systems:

- The In-Scope VPC in AWS has systems that perform analytics on the CDE web application (without accessing CHD).
- The In-Scope On-Premise Network has systems that provide security controls for the CDE, such as anti-malware and patch management.

The two out-of-scope networks in this reference architecture have no network connectivity to the AWS CDE, as discussed in Section 4.3.1 above.

4.3.3. Applicable AWS Services

The following AWS service help support compliance with PCI requirements for this architecture:

AWS Service	PCI Requirement Supported
– VPC	1.1.4, 1.2.1, 1.3.1-3, 1.3.5-7, 4.1, 6.4.1

4.3.4. Build Out

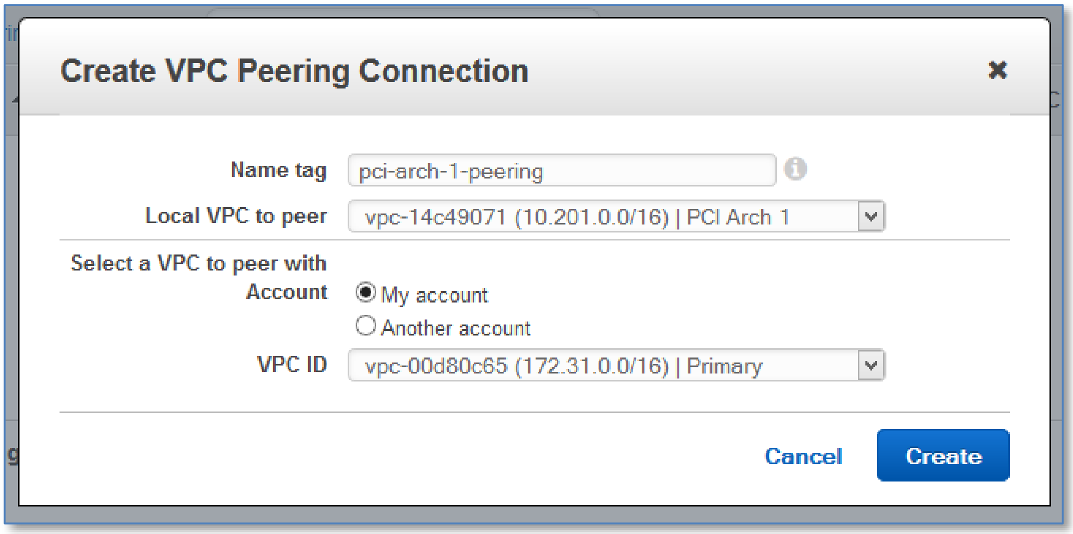
This section describes the primary steps for building out the reference architecture.

4.3.4.1. Create a VPC

Build the In-Scope VPC as per the steps in Architecture 2: Segmented CDE, described in Section 4.2 above.

4.3.4.2. Create a VPC Peering Connection

Create a VPC Peering connection between the CDE VPC and the newly created In-Scope VPC.



Create VPC Peering Connection

Name tag: pci-arch-1-peering

Local VPC to peer: vpc-14c49071 (10.201.0.0/16) | PCI Arch 1

Select a VPC to peer with

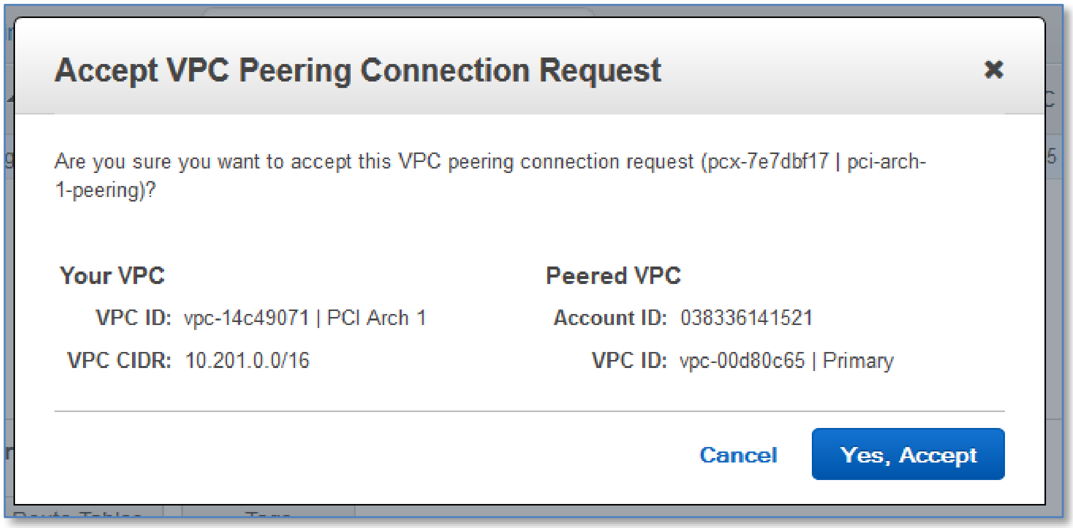
Account: ● My account ○ Another account

VPC ID: vpc-00d80c65 (172.31.0.0/16) | Primary

Figure 33 – Creating a VPC Peering Connection

4.3.4.3. Accept the VPC Peering Connection

After creating the VPC Peering connection, the peering request must be accepted. This is necessary as VPC Peering is supported to different AWS accounts.



Accept VPC Peering Connection Request

Are you sure you want to accept this VPC peering connection request (pcx-7e7dbf17 | pci-arch-1-peering)?

Your VPC

VPC ID: vpc-14c49071 | PCI Arch 1

VPC CIDR: 10.201.0.0/16

Peered VPC

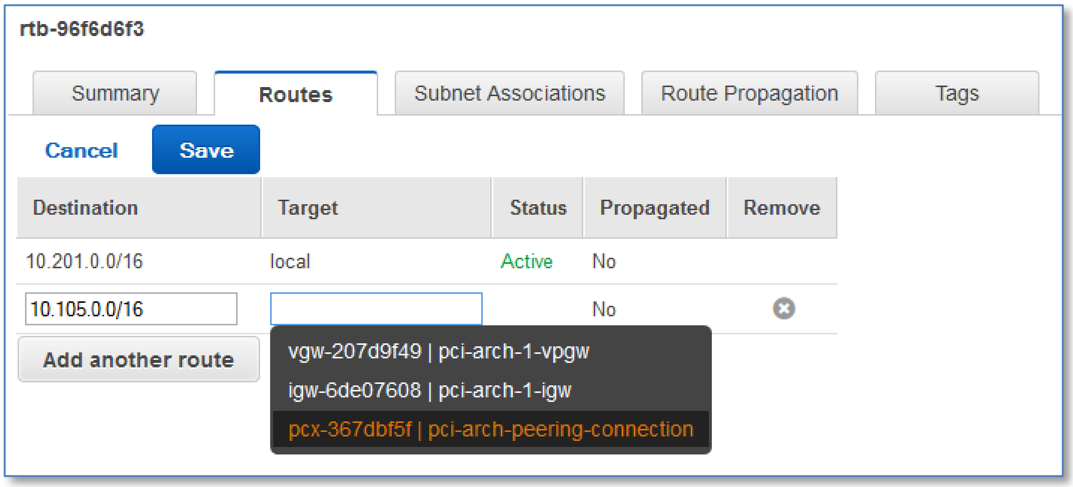
Account ID: 038336141521

VPC ID: vpc-00d80c65 | Primary

Figure 34 – Accepting a VPC Peering Request

4.3.4.4. Add Routes through the VPC Peering Connection

Once the VPC Peering connection is accepted, you will be able to add routes to the peered VPC to the routing tables in the CDE VPC. Do not forget to add return routes from the In-Scope VPC subnets back to the CDE.


rtb-96f6d6f3
Summary
Routes
Subnet Associations
Route Propagation
Tags
Cancel
Save
Add another route
vgw-207d9f49 | pci-arch-1-vpgw
igw-6de07608 | pci-arch-1-igw
pcx-367dbf5f | pci-arch-peering-connection

Destination	Target	Status	Propagated	Remove
10.201.0.0/16	local	Active	No	
10.105.0.0/16			No	⊗

Figure 35 – Adding a Route to a Peered VPC

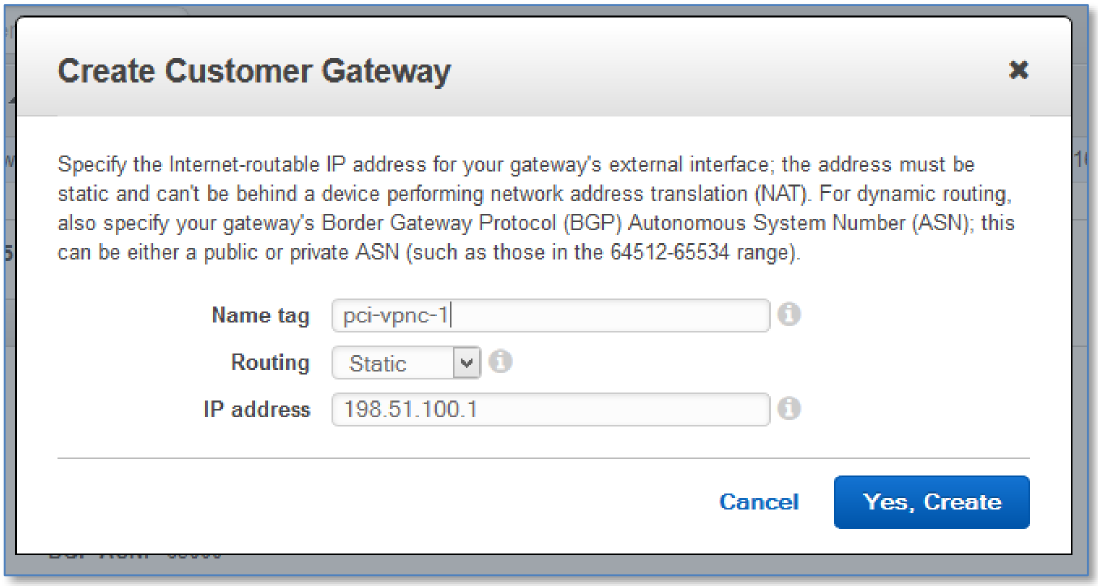
4.3.4.5. Leverage In-Scope VPC Resources

After the routes are added, the In-Scope VPC systems, such as the analytic systems cited in this example, will be able to access the CDE.

NOTE: *You will need to modify the CDE Security Groups, or create new ones, to allow connections from the In-Scope CDE to the appriopriate CDE Instances.*

4.3.4.6. Create a Customer Gateway

Within the CDE VPC, create a Customer Gateway. This resource in AWS represents a VPN concentrator at a client site.



Create Customer Gateway

Specify the Internet-routable IP address for your gateway's external interface; the address must be static and can't be behind a device performing network address translation (NAT). For dynamic routing, also specify your gateway's Border Gateway Protocol (BGP) Autonomous System Number (ASN); this can be either a public or private ASN (such as those in the 64512-65534 range).

Name tag: pci-vpnc-1

Routing: Static

IP address: 198.51.100.1

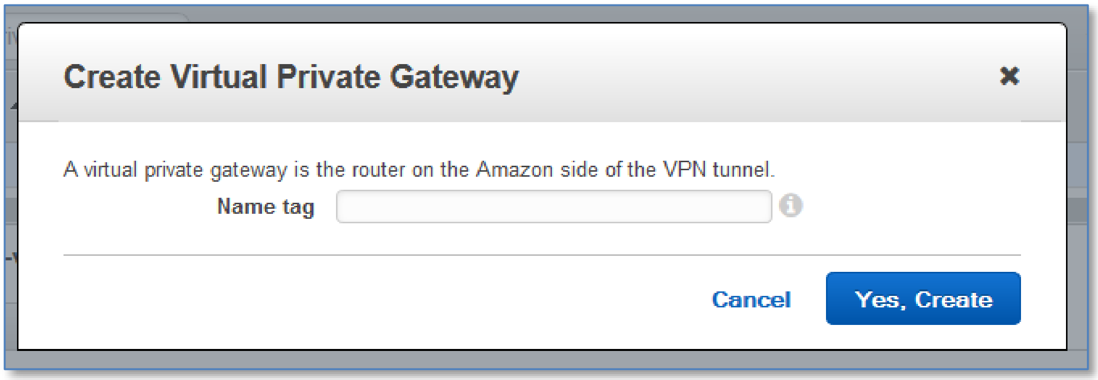
Figure 36 – Creating a Customer Gateway

NOTE: *Customer Gateways also support dynamic IP routing using BGP. If dynamic is selected. The gateway will also need the ASN for the network of the remote IP address.*

4.3.4.7. Create a Virtual Private Gateway

Within the VPC, create a Virtual Private Gateway (VPG). This resource in AWS represents an AWS routing target for a VPN connection.

Similar to an Internet Gateway, this a VPG is a specialized network interface used to send and receive external traffic. Attach the VPG to the VPC after creating it.



Create Virtual Private Gateway

A virtual private gateway is the router on the Amazon side of the VPN tunnel.

Name tag:

Figure 37 – Creating a Virtual Private Gateway

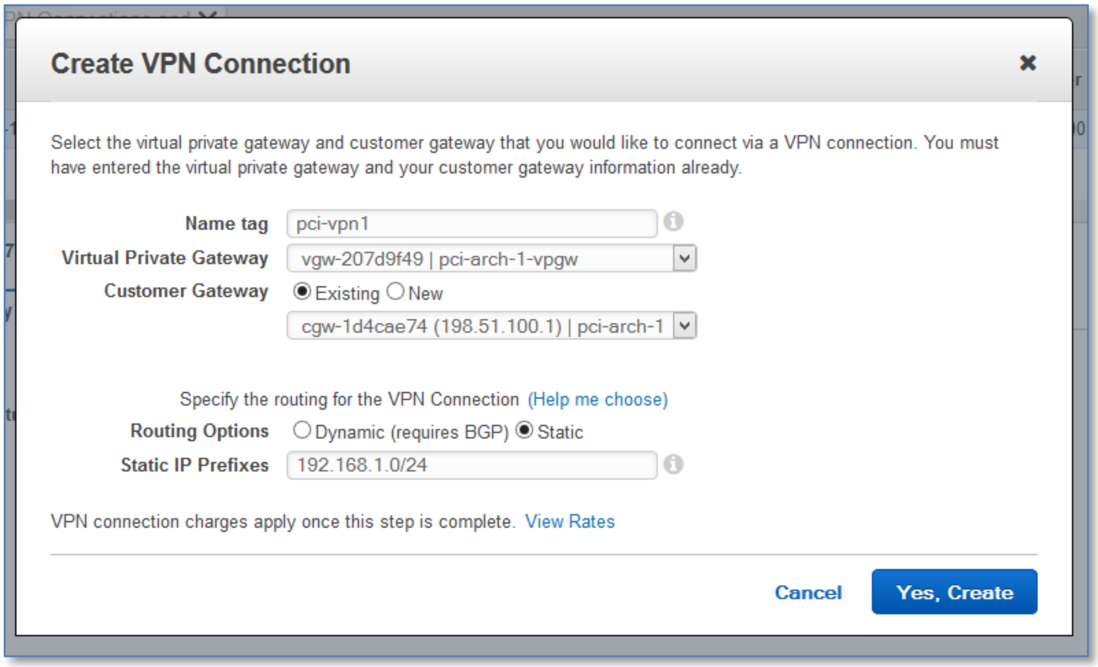
4.3.4.8. Create the VPN Connection

AWS supports industry-standard IPsec VPN connections. The VPN AWS resource provides the connection between the VPG and the Customer Gateway.

For the VPN, specify the VPG and the Customer Gateway created above.

The "Static IP Prefixes" item is the remote IP subnet to route through the VPN connection.

NOTE: *The VPN connection includes two AWS side end-points for redundancy.*



Create VPN Connection

Select the virtual private gateway and customer gateway that you would like to connect via a VPN connection. You must have entered the virtual private gateway and your customer gateway information already.

Name tag	pci-vpn1
Virtual Private Gateway	vgw-207d9f49 | pci-arch-1-vpgw
Customer Gateway	● Existing ○ New
	cgw-1d4cae74 (198.51.100.1) | pci-arch-1

Specify the routing for the VPN Connection (Help me choose)

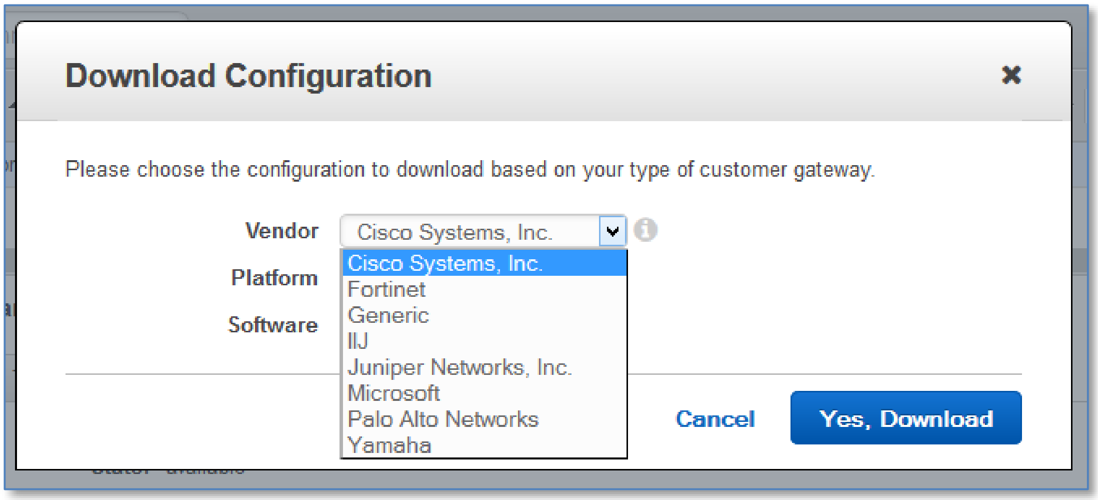
Routing Options	○ Dynamic (requires BGP) ● Static
Static IP Prefixes	192.168.1.0/24

VPN connection charges apply once this step is complete. View Rates

Figure 38 – Creating a VPN Connection

4.3.4.9. Download IPsec Configuration Details

Download the configuration needed for the On-Premise VPN concentrators. AWS supports native configuration files for a variety of firewall/VPN manufacturers such as Cisco and Fortinet.



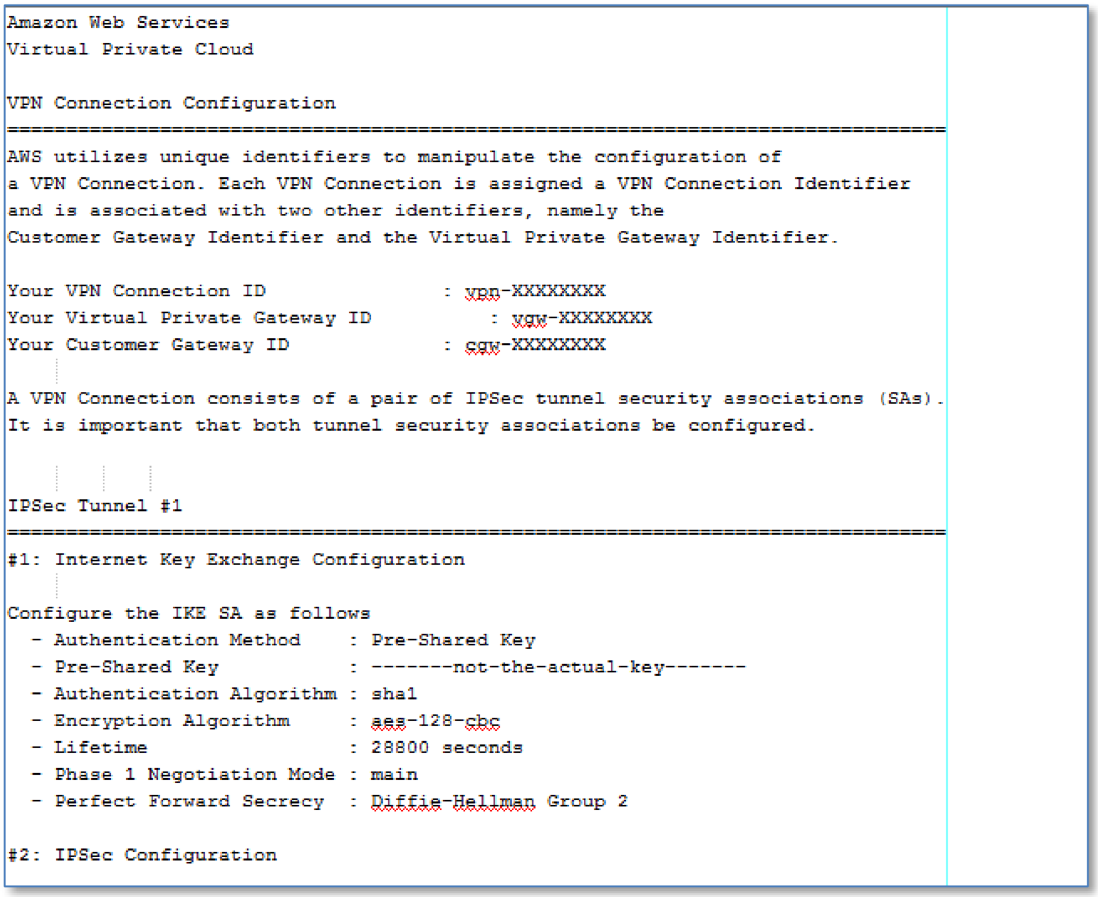
Download Configuration

Please choose the configuration to download based on your type of customer gateway.

Vendor: Cisco Systems, Inc.

Figure 39 – Downloading the VPN Configuration

There is a Generic option that allows the download of a text file containing the VPN connection details if you have a device not listed.

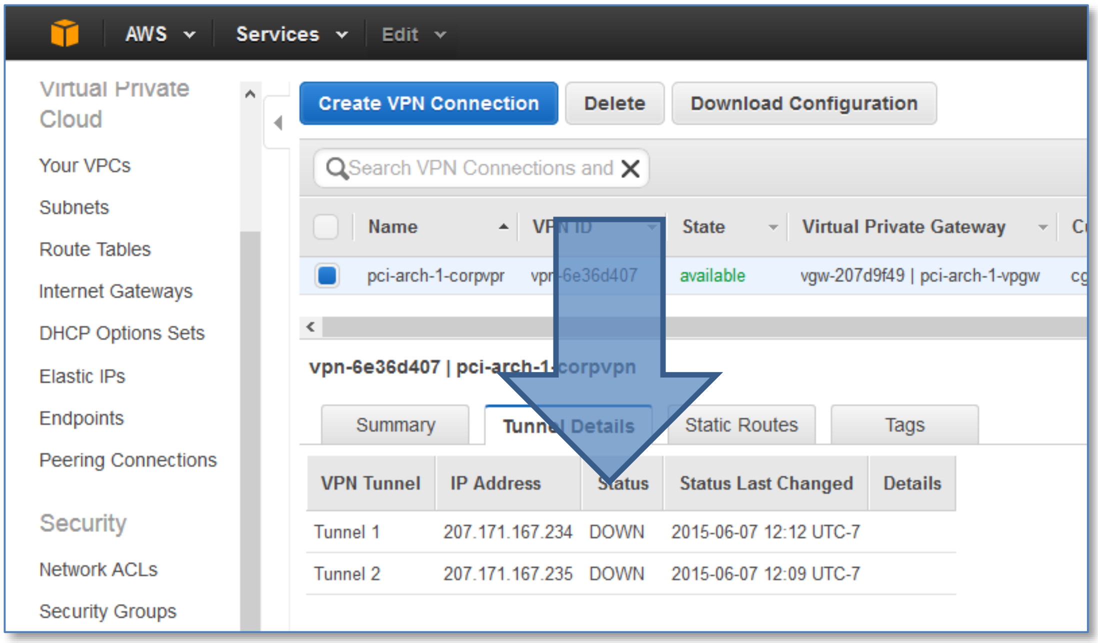


```
Amazon Web Services
Virtual Private Cloud

VPN Connection Configuration
================================================================================
AWS utilizes unique identifiers to manipulate the configuration of
a VPN Connection. Each VPN Connection is assigned a VPN Connection Identifier
and is associated with two other identifiers, namely the
Customer Gateway Identifier and the Virtual Private Gateway Identifier.

Your VPN Connection ID                : vpn-XXXXXXXX
Your Virtual Private Gateway ID           : vgw-XXXXXXXX
Your Customer Gateway ID              : cgw-XXXXXXXX

A VPN Connection consists of a pair of IPSec tunnel security associations (SAs).
It is important that both tunnel security associations be configured.


IPSec Tunnel #1
================================================================================
#1: Internet Key Exchange Configuration

Configure the IKE SA as follows
  - Authentication Method    : Pre-Shared Key
  - Pre-Shared Key           : -------not-the-actual-key-------
  - Authentication Algorithm : sha1
  - Encryption Algorithm     : aes-128-cbc
  - Lifetime                 : 28800 seconds
  - Phase 1 Negotiation Mode : main
  - Perfect Forward Secrecy  : Diffie-Hellman Group 2

#2: IPSec Configuration
```

Figure 40 – Example Generic VPN Configuration File

4.3.4.10. Verify VPN Tunnel Status

After configuring the On-Premise Client VPN endpoints, verify that the VPN was able to come up. View the tunnel status from the "Tunnel Details" tab of the VPN resource details pane.



	Name	VPN ID	State	Virtual Private Gateway	Cu
■	pci-arch-1-corpvpr	vpn-6e36d407	available	vgw-207d9f49 | pci-arch-1-vpgw	cg

vpn-6e36d407 | pci-arch-1-corpvpn

VPN Tunnel	IP Address	Status	Status Last Changed	Details
Tunnel 1	207.171.167.234	DOWN	2015-06-07 12:12 UTC-7	
Tunnel 2	207.171.167.235	DOWN	2015-06-07 12:09 UTC-7	

Viewing the VPN tunnels status

4.3.4.11. Leverage On-Premise Resources

After the VPN tunnels come up, the on-premise in-scope systems will be available for use within the AWS CDE, such as the AV and patch management consoles cited in this example.

5. CONCLUSION

AWS is a powerful cloud platform. It offers numerous capabilities to support a fully PCI-compliant environment. However, it is important that you and your PCI assessor understand these capabilities.

This workbook clarifies some of these issues. Ultimately, achieving PCI compliance is a function of how effectively you deploy, configure, manage, and document the environment. AWS offers an outstanding platform for PCI compliance, yet it requires understanding how the AWS natively supports the various PCI requirements so you can use them correctly.

5.1. Support

If you need support with AWS, you should contact Amazon's AWS technical support.

If you require consulting or assessment services, Anitian can help. We offer a comprehensive suite of PCI compliance services including penetration testing, vulnerability scanning, technology integration, and QSA assessment.

Contact us at: 888-264-8456, email info@anitian.com, or visit our site www.anitian.com.

APPENDIX A: RESPONSIBILITY MATRIX SUMMARY

Requirement	AWS Responsibility	Customer Responsibility
Requirement 1: Install and maintain a firewall configuration to protect cardholder data.	AWS maintains instance isolation for host operating systems and the AWS Management Environment including host operating system, hypervisor, firewall configuration and baseline firewall rules. AWS meets all requirements for implementing and managing firewalls for the AWS management environment.	AWS customers are responsible for security group definitions and network access control rules.
Requirement 2: Do not use Supplier-supplied defaults for system passwords and other security parameters.	AWS develops and maintains configuration and hardening standards for the AWS Management Environment that provides the virtualization technologies and applications for providing the cloud services.	AWS customers are responsible for secure and configuration for all customer-configurable items. This may include Operating System (OS) configuration for EC2 instances, logging and log retention for database services, and/or permissions for AWS management functions.
Requirement 3: Protect stored cardholder data.	AWS does not manage cardholder data or encryption technologies and keys for the customers' specific cardholder environment.	AWS customers are responsible for encrypting data stored in AWS. This includes key management requirements.
Requirement 4: Encrypt transmission of cardholder data across open, public networks.	AWS encrypts access and manages encryption within the AWS Management Environment.	AWS customers are responsible for implementing encryption on all applicable internal and external network connections.
Requirement 5: Use and regularly update anti-virus software or programs.	AWS manages anti-virus software for the AWS Management Environment and, where appropriate, for the identified services.	AWS customers are responsible for implementing anti-virus software on OS instances commonly subject to malware.

Requirement	AWS Responsibility	Customer Responsibility
Requirement 6: Develop and maintain secure systems and applications.	AWS maintains security patching, development and change control of the applications that support the services included in the assessment including web interfaces, APIs, access controls, provisioning and deployment mechanisms. AWS develops and manages changes to the applications that support the services included in the assessment including web interfaces, APIs, access controls, provisioning and deployment mechanisms.	AWS customers are responsible for monitoring published OS and application vulnerabilities and patching appropriately. Customers are required to use documented change control for all configurations and customer code. Customers who develop custom code that is used to transmit, process, or store credit card data must comply with requirements for secure development and testing.
Requirement 7: Restrict access to cardholder data by business need-to-know.	AWS maintains the access controls related to underlying infrastructure systems and the AWS Management Environment.	AWS customers are responsible for managing access to all AWS services that are included in their CDE. IAM can be used to manage resource management and AWS configuration roles and permissions. Customers are responsible for configuring AWS account and session controls to meet requirements. Customers must be aware of AWS guidelines for credentials and access control for AWS resource management.
Requirement 8: Assign a unique ID to each person with computer access.	AWS provides each user in the AWS Management Environment a unique ID. AWS provides additional security options that enable AWS customers to further protect their AWS Account and control access: AWS Identity and Access Management (AWS IAM), Multi-Factor Authentication (MFA) and Key Rotation.	AWS customers are responsible for managing access to all AWS services that are included in their CDE. IAM can be used to manage resource management and AWS configuration roles and permissions. Customers are responsible for configuring AWS account and session controls to meet requirements. Customers must be aware of AWS guidelines for credentials and access control for AWS resource management.
Requirement 9: Restrict physical access to cardholder data.	AWS maintains the physical security and media handling controls for the services included in the assessment.	No customer responsibilities.

Requirement	AWS Responsibility	Customer Responsibility
Requirement 10: Track and monitor all access to network resources and cardholder data.	AWS maintains audit logs for the AWS Management Environment.	AWS customers are responsible for logging within all OS instances. AWS customers are responsible for configuration of logging within AWS services. User activity logs of resource management activities via the console and command line are available to users via Amazon CloudTrail. Amazon CloudTrail must be used to record and monitor AWS resource management activities.
Requirement 11: Regularly test security systems and processes.	AWS conducts wireless rogue access point detection, vulnerability and penetration testing, intrusion detection and file integrity monitoring for the AWS Management Environment and the identified services. AWS implements and monitors IDS/IPS on networks that implement AWS services.	AWS customers are responsible for internal and external scanning and pen testing of their instances and virtual networks. Customers must follow AWS processes for scanning and pen testing. AWS customers are responsible for implementing IDS functionality, typically using host-based IDS (HIDS) network segments they implement and manage.
Requirement 12: Maintain a policy that addresses information security for employees and contractors.	AWS maintains security policies and procedures, security awareness training, security incident response plan, and human resource processes that align with PCI requirements.	AWS customers are responsible for all policies and procedures. AWS customers should include AWS as an infrastructure provider for Req 12.8. Alerts from AWS should be part of the IRP for Req. 12.9.
Requirement A: Shared hosting providers must protect the cardholder data environment.	AWS customer instances and data are protected by instance isolation and other security measures in the AWS Management Environment.	AWS customers may also be considered a shared hosting provider, if they run applications or store data for their customers. In this case, customers are responsible for protecting their customer's data within AWS services.

APPENDIX B: RESOURCES CITED

The following table summarizes all links referenced throughout this technical workbook.

Section	Resource	Link
1.2.3	PCI DSS version 3.2	https://www.pcisecuritystandards.org/security_standards/documents.php
1.2.3	Manage AWS Environments	http://aws.amazon.com/getting-started/
1.2.3	PCI Cloud Computing Guidelines	https://www.pcisecuritystandards.org/pdfs/PCI_DSS_v2_Cloud_Guidelines.pdf
2	AWS PCI Level 1 FAQ	http://aws.amazon.com/compliance/pci-dss-level-1-faqs
2.3.1	Request copy of AWS AOC	http://aws.amazon.com/compliance/contact/
3.3	EBS Encryption	http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSEncryption.html#EBSEncryption_supported_instances
3.3	Amazon S3 Server Side Encryption	http://docs.aws.amazon.com/AmazonS3/latest/dev/serv-side-encryption.html
3.3	Amazon S3 Upload Objects	http://docs.aws.amazon.com/AmazonS3/latest/UG/UploadingObjectsintoAmazonS3.html
3.3	AWS KMS Cryptographic Details	https://d0.awsstatic.com/whitepapers/KMS-Cryptographic-Details.pdf
3.3	AWS KMS API Key Rotation	http://docs.aws.amazon.com/kms/latest/APIReference/API_EnableKeyRotation.html
3.3	AWS KMS Manual Key Creation	http://docs.aws.amazon.com/kms/latest/developerguide/rotate-keys.html
3.3	Logging using CloudTrail	http://docs.aws.amazon.com/kms/latest/developerguide/logging-using-cloudtrail.html
3.4	ELB Security Policies Table	http://docs.aws.amazon.com/ElasticLoadBalancing/latest/DeveloperGuide/elb-security-policy-table.html
3.4	VPC FAQ	http://aws.amazon.com/vpc/faqs/
3.6	AWS Linux Security Center	https://alas.aws.amazon.com/
3.7	Admin Guide Directory Management	http://docs.aws.amazon.com/directoryservice/latest/adminguide/directory_management.html

Section	Resource	Link
3.8	Directory Service – Create a Directory	http://docs.aws.amazon.com/directoryservice/latest/adminguide/create_directory.html
3.10	CloudTrail Event Reference Record	http://docs.aws.amazon.com/awscloudtrail/latest/userguide/event_reference_record_body.html
3.10	Amazon S3 Lifecycle Configuration	http://docs.aws.amazon.com/AmazonS3/latest/UG/LifecycleConfiguration.html
3.11	AWS Penetration Testing Information	http://aws.amazon.com/security/penetration-testing
4.1.4.6	Amazon VPC User Guide	http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/vpc-ip-addressing.html#subnet-public-ip
5.1	AWS Technical Support	https://aws.amazon.com/premiumsupport/
5.1	Anitian Website	www.anitian.com
App. A	AWS Pen Test Guidelines	http://aws.amazon.com/security/penetration-testing/