DoD-Compliant Implementations in the AWS Cloud

Reference Architectures

Paul Bockelman

Andrew McDermott

April 2015

Contents

Abstract

This whitepaper is intended for existing and potential DoD mission owners who are designing the security infrastructure and configuration for applications running in Amazon Web Services (AWS). It provides security best practices that will help you properly design and deploy compliant DoD applications and protect your data and assets in the AWS Cloud. The paper is targeted at IT decision makers and security personnel and assumes that mission owners are familiar with basic security concepts in the areas of networking, operating systems, data encryption, and operational controls.

AWS provides a secure hosting environment for mission owners to field their applications, but does not relieve the mission owners of their responsibility to securely deploy, manage and monitor their applications in accordance with DoD security and compliance policy. When operating an application on AWS, the mission owner is responsible for the guest operating system, management of users, and the configuration of the AWS-provided networking functionality.

Introduction

On 26 June 2012, the Department of Defense (DoD) Chief Information Officer (CIO) released a memo[i] specifying that the Defense Information Systems Agency (DISA) will perform cloud brokerage functions. The purpose and mission of DISA in the cloud brokerage role is to aid the DoD in *"achieving IT efficiencies, reliability, interoperability, and improve security and end-to-end performance by using cloud service offerings."* In support of this mission, DISA developed a Cloud Security Model (CSM) to establish security guidelines for hosting DoD data and mission applications in a cloud environment. AWS has achieved three Provisional Authorizations to Operate for mission systems designated as CSM Level 1-2 (covering all AWS regions in the contiguous United States (CONUS)) and CSM Level 3-5 (covering only the US GovCloud region).

In January 2015, DISA issued the DoD Cloud Computing (CC) Security Requirements Guide (SRG), which has revised the guidance for Cloud Service Providers and for DoD mission owners. The DoD CC SRG supersedes the CSM, and is now the primary guidance for cloud computing within the DoD community. AWS is currently transitioning to the CC SRG in accordance with DoD guidance, and will attain updated DoD authorizations based upon the CC SRG in the near future. Visit the AWS DoD compliance page for the latest information on AWS DOD Provisional Authorizations[1].

This whitepaper is a high-level guide for DoD mission owners and partners in the design of mission solutions that are both CC SRG-compliant and operating in the AWS Cloud at

[1] http://aws.amazon.com/compliance/dod-csm-faqs/

levels 2 and 4-5. Although there are many design permutations that will meet CC SRG requirements on AWS, this document presents two reference architectures that will address many of the common use cases for levels 2 and 4-5.

Getting Started

When considering a move to the AWS Cloud, mission owners must first make sure that their IT plans align with their organization's business model. Having a solid understanding of the core competencies of your organization will help you identify the areas that are best served through an external infrastructure, such as the AWS Cloud.

Next, mission owners have to think through key technology questions. The list will vary depending upon the project and mission, but usually includes the following questions:

- Do you have legacy applications that need greater scalability, reliability, or security than you can afford to maintain in your own environment?
- What are your hardware and bandwidth capacity requirements?
- How will you be prepared to scale up (and down) following deployment?
- How can the cloud advance your IT and business objectives?

As you answer each question, apply the lenses of *flexibility*, *cost effectiveness*, *scalability*, *elasticity*, and *security*. Taking advantage of Amazon Web Services will allow you to focus on your core competencies and leverage the resources and experience AWS provides.

Shared Responsibilities and Governance

Because mission owners are building systems on top of the AWS Cloud infrastructure, many security measures will be *shared*: mission owners will provide security for their software components, and AWS will provide security for its infrastructure. Mission owners will also be able to leverage security controls from AWS's security, meaning that mission owners won't have to provide those controls for their components because AWS is already providing them.

Shared Responsibility Environment

Moving IT infrastructure to services in AWS creates a model of shared responsibility between the mission owner and AWS. This shared model can help relieve the operational burden of mission owner because AWS operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the service operates.

The mission owner assumes responsibility for and management of the guest operating system (including updates and security patches), other associated application software, and the configuration of the AWS-provided security group firewall. Mission owners should carefully consider the services they choose because their responsibilities vary depending on the services used, the integration of those services into their IT environment, and applicable laws and regulations.

It is possible for mission owners to enhance security and/or meet their more stringent compliance requirements by leveraging AWS technology such as host-based firewalls, host-based intrusion detection/prevention, encryption, and key management. The nature of this shared responsibility also provides the flexibility and mission owner control that permits the deployment of solutions that meet industry-specific certification requirements.

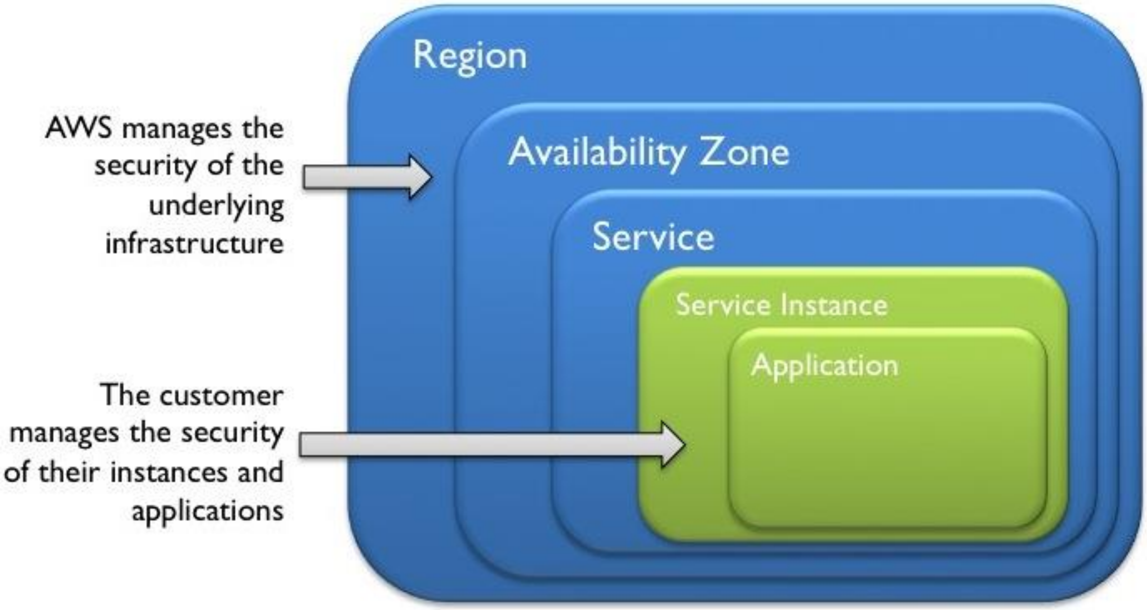
Region
Availability Zone
Service
Service Instance
Application
AWS manages the security of the underlying infrastructure
The customer manages the security of their instances and applications

Figure 1: Control Responsibilities

This mission owner/AWS shared responsibility model also extends to IT controls. Just as the responsibility to operate the IT environment is shared between AWS and its mission owners, so is the management, operation, and verification of shared IT controls. AWS can manage those controls associated with the physical infrastructure deployed in the AWS environment that might previously have been managed by the mission owner. Mission owners can then use the AWS control and compliance documentation available to them (described in the AWS Certifications and Third-party Attestations section of this document) to perform their control evaluation and verification procedures as required.

Mission Owner Responsibilities

Service Instance Management

Mission owners are responsible for managing their instantiations of storage buckets and objects, compute instances, and virtual private cloud (VPC) network environments. This includes mission-owner-installed operating systems and applications that are outside of the authorization boundary. Mission owners are also responsible for managing specific controls relating to shared touch points within the security authorization boundary, such as establishing customized security control solutions. Examples include, but are not limited to, configuration and patch management, vulnerability scanning, disaster recovery, protecting data in transit and at rest, host firewall management, managing credentials, identity and access management, and managing VPC network configurations.

Mission owners provision and configure their implementation of storage, virtual machines, and VPCs using API calls to the AWS API endpoints. This allows the mission owner to launch and terminate cloud instances, change firewall parameters, and perform other management functions.

Application Management

Major Applications (MAs) that run on AWS services are entirely the responsibility of each mission owner to configure and maintain. Mission owners should develop a separate System Security Plan that addresses the controls relevant to each application.

Operating System Maintenance

AWS-provided images (Amazon Linux AMIs) for the guest operating system are patched to a point in time. The installation will be the minimum installation of the operating system. Amazon Machine Images (AMIs) are also provided containing standard releases from OS vendors such as Windows Server, RHEL, SUSE Linux, and Ubuntu Linux with no additional configuration applied to the image. AWS does not perform patch management or system hardening, and does not provide any application support within the image. DoD mission owners are responsible for properly hardening and patching their AMIs in accordance with DoD Security Technical Implementation Guides and the Information Assurance Vulnerability Management process.

At mission owner instantiation of an AMI, AWS makes no warranties as to the patch level or configuration settings. Mission owner responsibility includes updating any Amazon Elastic Compute Cloud (EC2) instance to a recent patch level and configuring to suit specific needs. Upon deployment of virtual machines, the mission owner assumes full administrator access and is responsible for performing additional configuration, patching, security hardening, vulnerability scanning, and application installation, as necessary. AWS will not maintain administrator access to mission owners' AMIs.

An AMI provides the information required to launch an EC2 instance, which is a virtual server in the cloud. The mission owner specifies the AMI used to launch an instance (e.g., Windows Server, Ubuntu, RHEL, etc.), and the mission owner can launch as many instances from the AMI as needed. An AMI includes the following:

- A template for the root volume for the instance (for example, an operating system, an application server, and applications). The root volume of an instance is either an Amazon Elastic Block Store (EBS) volume or an instance store volume.
- Launch permissions that control which AWS accounts can use the AMI to launch instances.
- A block device mapping that specifies the volumes to attach to the instance when it's launched.

Mission owners can customize the instance launched from a public AMI and then save that configuration as a custom AMI for the mission owner's own use. After mission owners create and register an AMI, they can use it to launch new instances. Instances launched from this customized AMI then use all the customizations the mission owner has made. The mission owner can deregister the AMI when finished. After the AMI is deregistered, mission owners cannot use it to launch new instances.

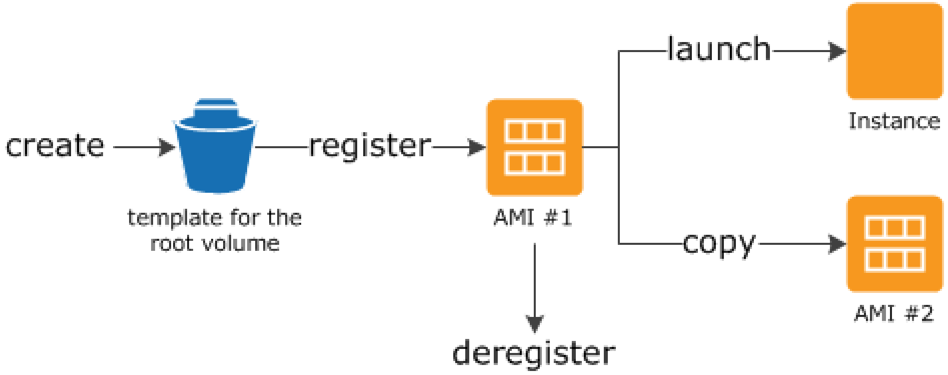

Figure 2: Using an AMI

Configuration of Security Groups

AWS mission owners are responsible for properly configuring their security groups in accordance with their pertinent networking policies, and ensuring that they regularly review their security group configuration and instance assignment in order to maintain a secure baseline. Security groups are not a solution that can be deployed using a one-size-fits-all approach. They should be carefully tailored to the intended functionality of each class of instance deployed within the mission owner's AWS environment.

VPC Configuration

Amazon Virtual Private Cloud (VPC) provides enhanced capabilities that AWS mission owners can use to further secure their AWS environment through the deployment of traditional networking concepts, such as DMZs and subnets that are segregated by functionality. A VPC's network ACLs provide stateless filtering that can be used in a fashion similar to a screening firewall or router to easily defend against malicious traffic at the subnet level. This adds another layer of network security on top of the mission owner's security group implementation.

Backups

Mission owners are responsible for establishing an effective backup strategy using AWS or third-party functionality that meets the retention goals identified for their application. Through effective use of Amazon EBS snapshots, mission owners can ensure their data is backed up to Amazon Simple Storage Service (S3) on a regular basis. Mission owners are responsible for setting and maintaining proper access permissions to their EBS volumes and Amazon S3 buckets and objects.

Host-Based Security Tools

Mission owners should install and manage anti-malware and host-based intrusion systems in accordance with their organization's policies. Host-based security tools can be included within the mission owner's AMIs, or installed via bootstrapping services upon launching an instance.

Vulnerability Scanning and Penetration Testing

Mission owners are responsible for conducting regular vulnerability scanning and penetration testing of their systems, as mandated by their organization's policies. All vulnerability and penetration testing must be properly coordinated with AWS Security, in accordance with AWS policy. For more information, go to http://aws.amazon.com/security/penetration-testing/

Identity and Access Management

AWS mission owners are responsible for properly managing their root AWS account, as well as any IAM users, groups and roles they associated with their account. Through the proper use of AWS Identity and Access Management (IAM), mission owners can implement an access schema that properly separates users by their identified roles and responsibilities, thereby establishing least privilege and ensuring that users have only the permissions necessary to perform their assigned tasks.

Identity Federation

AWS supports identify federation with on-premises authentication stores such as Lightweight Directory Access Protocol/ Active Directory (LDAP/AD), which allows mission owners to effectively mirror their existing account structures within AWS without having to recreate them.

Multi-Factor Authentication

At a minimum, AWS mission owners should implement multi-factor authentication (MFA) for their root AWS account, as well as any privileged IAM accounts associated with it. MFA can be used to add an additional layer of security in Amazon S3, through activation of the MFA delete feature.

Bastion Hosts

Mission owners should implement a bastion host for administering their AWS environment. Bastion hosts are hardened instances used for administrative tasks within the AWS environment. Rather than opening up all instances to SSH access from across the entirety of the Internet, access can be restricted to a single instance, thereby limiting the attack surface for possible compromises. Access to the bastion host should be limited to known good IP addresses within the mission owner's organization, require valid SSH keys, and require two-factor authentication. Auditing on the bastion host should be configured to record all administrative activity.

Auditing

Mission owners are responsible for properly configuring their AMIs to ensure that required audit logs are being generated. Audit logs should be forwarded to a dedicated log server instance located in the mission owner's VPC management subnet, or written to a properly secured Amazon S3 bucket to ensure that sensitive data is properly protected. Additionally, the mission owner should enable the use of AWS CloudTrail, if it is available, which will provide detailed activity logs listing API calls made by their AWS account and any associated IAM users.

Disaster Recovery

Mission owner applications built using AWS are designed, constructed, and operated by the mission owner, and the responsibility for recovery, restoration, and system contingency planning is incumbent upon the mission owner. Therefore, mission owners should architect their AWS usage to take advantage of multiple regions and Availability Zones. Distributing applications across multiple Availability Zones provides the ability to remain resilient in the face of most failure modes including natural disasters or system failures.

Data Spills

To provide protection against spills, all mission owner data stored within Amazon EBS and Amazon S3 must be encrypted using AES 256 in accordance with DoD guidance. The mission owner is responsible for implementing FIPS 140-2 validated encryption for data at rest, with customer-managed keys, in accordance with DoD policy. The combination of the mission owner's encryption and AWS's automated wipe functionality ensures that any spilled data is ciphertext, sufficiently limiting the risk of accidental disclosure. AWS degausses and destroys all decommissioned media in accordance with NIST and NSA standards.

Intrusion Detection

AWS and its mission owners each bear responsibility for ensuring that the AWS infrastructure and all systems hosted on it, respectively, are properly monitored for unauthorized activity. Mission owners are responsible for properly implementing host-based intrusion detection on their instances, as well as any required network-based intrusion detection, whether it is implemented using a hardware-based solution housed within a colocation facility, or a fully virtualized system within their AWS environment. Mission owners are responsible for coordinating deployment of their intrusion detection capabilities with their CNDSP.

Compliance Governance

AWS mission owners are required to continue to maintain adequate governance over the entire IT control environment regardless of whether it is deployed in a traditional data center or within the cloud. Leading practices include an understanding of required compliance objectives and requirements (from relevant sources), the establishment of a control environment that meets those objectives and requirements, an understanding of the validation required based on the organization’s risk tolerance, and the verification of the operating effectiveness of the control environment. Deployment in the AWS Cloud gives organizations options to apply various types of controls and various verification methods.

Strong mission owner compliance and governance might include the following basic steps:

1. Review information from AWS and other sources to understand the entire IT environment.
2. Document all compliance requirements.
3. Design and implement control objectives to meet the organization’s compliance requirements.
4. Identify and document controls owned by outside parties.
5. Verify that all control objectives are met and all key controls are designed and operating effectively.

Approaching compliance governance in this manner will help mission owners gain a better understanding of their control environment and will help clearly delineate the verification activities to be performed. More information regarding governance within the cloud is available in our Security at Scale: Governance in AWS[2] whitepaper.

[2] http://media.amazonwebservices.com/AWS_Security_at_Scale_Governance_in_AWS.pdf

What Is FedRAMP?

On February 8, 2011, the Office of Management and Budget (OMB) released *The Federal Cloud Computing Strategy*, which established guidance for all federal agencies to adopt cloud technologies across the federal government. This strategy was followed by a federal requirement released in December 2011 establishing the Federal Risk and Authorization Management Program (FedRAMP). FedRAMP is mandatory for federal agency cloud deployments and service models at the low and moderate risk impact levels.

The DoD CC SRG uses the FedRAMP program to establish a standardized approach for DoD entities that are acquiring cloud services. AWS has been assessed and approved under FedRAMP, and has been issued two Agency Authority to Operate (ATO) authorizations covering all CONUS regions[3], namely AWS GovCloud (US), US East, and US West. For more information on the compliance of AWS with FedRAMP visit our FedRAMP FAQ[4] page. All cloud service providers (CSPs) must achieve compliance with FedRAMP before they can be considered for a provisional authorization under the CC SRG by DoD.

What Is the Cloud Computing SRG?

The Department of Defense (DoD) CC SRG provides a formalized assessment and authorization process for CSPs to gain a DoD Provisional Authorization (PA), which can subsequently be leveraged by DoD mission owners. AWS's provisional authorizations under the CSM provide a reusable certification that attests to the compliance of AWS with DoD standards, reducing the time necessary for a DoD mission owner to assess and authorize one of their systems for operation on AWS.

The CC SRG supports the overall federal goal to increase the utilization of commercial cloud computing, and it provides a means for the DoD to support this goal. The CC SRG requires the categorization of mission systems and their workloads at one of four impact levels (formerly six under the DoD CSM). The level is assigned based on a determination of the data sensitivity of a particular system and the controls required to protect it, starting at level 2 (lowest) through level 6 (highest). The following table summarizes the impact levels with a description of a typical workload, connectivity restrictions, Border Cloud Access Point (BCAP) requirements, and Computer and Network Defense (CND) requirements.

[3] http://aws.amazon.com/about-aws/global-infrastructure/

[4] http://aws.amazon.com/compliance/fedramp-faqs/

SRG Level	Workload Description	Public Access	BCAP	Computer and Network Defense
2	Publicly releasable; as well as data with limited controlled access	Yes	Optional	Cloud-based or Managed CND Suite
4	CUI	No	Required	BCAP and Border CND suite
5	CUI that requires higher levels of protection, as well as NSS	No	Required	BCAP and Border CND suite
6	Classified, not permissible			

To begin planning for the deployment of a DoD mission system in AWS, it is critical that the CC SRG impact level categorization be made in advance. Systems designated at impact level 2 can begin deployments relatively quickly. Conversely, a designation at impact level 4 or 5 requires that the mission application on AWS be connected to the Nonsecure Internet Protocol Router Network (NIPRNet) by means of AWS Direct Connect, Internet Protocol Security (IPsec) virtual private network (VPN), or both. This NIPRNet connection also requires that the traversal of all ingress and egress traffic to and from the Amazon Virtual Private Cloud (VPC) be routed through a Border Cloud Access Point (BCAP) and its associated Computer and Network Defense suite. The provisioning of circuits for an AWS Direct Connect-to-NIPRNet connection typically has a substantial lead-time, so mission owners should plan accordingly.

For more information regarding the Department of Defense CC SRG, please refer to the DISA Information Assurance Support Environment (IASE) website for the latest Cloud Security announcements and requirements[5] or the latest CC SRG v1.1 document[6].

[5] http://iase.disa.mil/cloud_security/

[6] http://iase.disa.mil/cloud_security/Documents/u-cloud_computing_srg_v1r1_final.pdf

FedRAMP + CC SRG Compliance = the Path to AWS

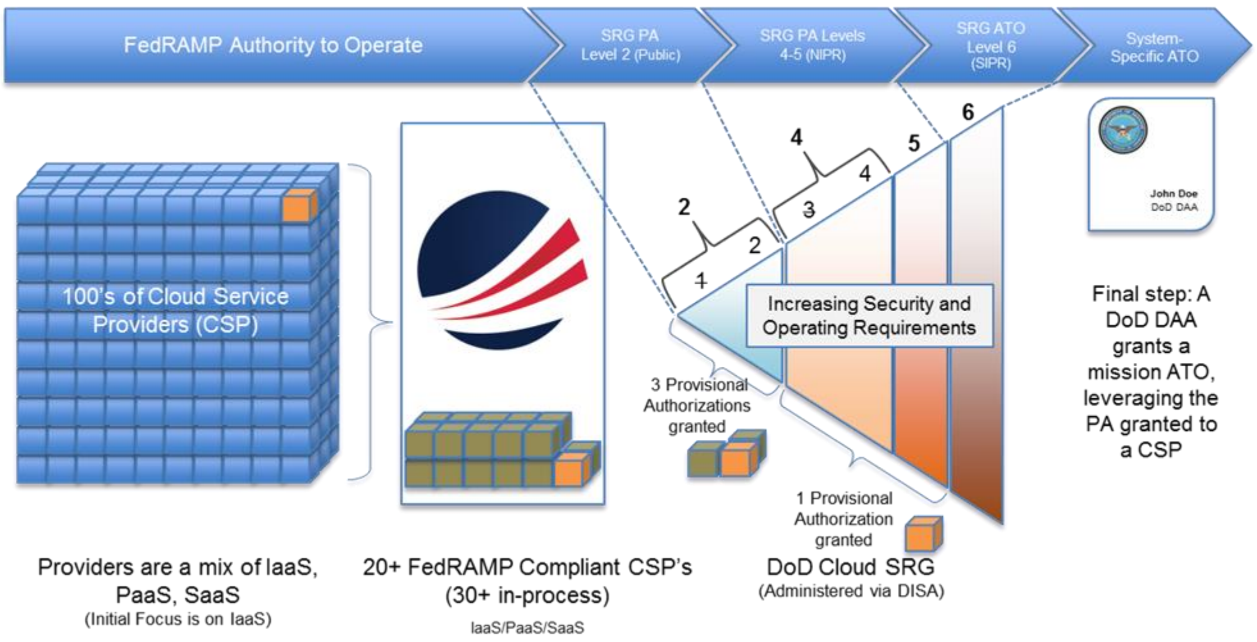
FedRAMP Authority to Operate
SRG PA Level 2 (Public)
SRG PA Levels 4-5 (NIPR)
SRG ATO Level 6 (SIPR)
System-Specific ATO
100's of Cloud Service Providers (CSP)
1
2
2
3
4
4
5
6
Increasing Security and Operating Requirements
3 Provisional Authorizations granted
1 Provisional Authorization granted
John Doe
DoD DAA
Final step: A DoD DAA grants a mission ATO, leveraging the PA granted to a CSP
Providers are a mix of IaaS, PaaS, SaaS
(Initial Focus is on IaaS)
20+ FedRAMP Compliant CSP's
(30+ in-process)
IaaS/PaaS/SaaS
DoD Cloud SRG
(Administered via DISA)

Figure 3: The FedRAMP and CC SRG Roadmap

For DoD application owners to gain a mission ATO for their cloud-deployed applications from their approving authority, they must select a cloud service provider (CSP) that has obtained a provisional authorization from the DoD. Gaining authorization under FedRAMP is the first step toward gaining authorization from DoD. There are four paths into the FedRAMP repository, with the Joint Authorization Board (JAB) and Agency ATO paths being used most often to date. If a CSP wants to go beyond FedRAMP and become a DoD CSP, the CSP must go further through the DoD CC SRG assessment process. Currently, attaining a FedRAMP Moderate authorization enables a CSP to be considered for Level 2 of the CC SRG, while all CSPs are required to undergo additional assessment against the FedRAMP+ controls of Levels 4 and 5 prior to being granted a PA at those levels.

Regardless of whether the Designated Accrediting Authority (DAA) is using the DoD Information Assurance and Certification Accreditation Process (DIACAP) or the Risk Management Framework (RMF) process, the DAA has the ability to leverage and inherit the Provisional Authorization package(s) as part of its assessment toward a final ATO which only it grants (not the Defense Information Systems Agency (DISA)). The RMF process has been formally adopted by the DoD, and its use will become more prevalent in 2015.

AWS Global Infrastructure

AWS provides facilities and hardware in support of mission owners, with security features controlled by AWS at the infrastructure level. In the infrastructure as a service (IaaS) model, AWS is responsible for applicable service delivery layers including: infrastructure (hardware and software that comprise the infrastructure) and service management processes (the operation and management of the infrastructure and the system and software engineering lifecycles). Mission owners use AWS to manage the cloud infrastructure including the network, data storage, system resources, data centers, security, reliability, and supporting hardware and software.

Across the globe, the infrastructure of AWS is organized into regions. Each region contains Availability Zones, which are located within a particular geographic area that allows for low-latency communication between the zones. Customer data resides within a particular region, and is moved between regions at the direction of the customer. Currently, there are four regions available within CONUS that are permitted for use by the DoD.

Each Availability Zone has an identical IaaS cloud services platform, offering computing power (Amazon EC2), storage (Amazon EBS, Amazon S3), secure communications (Amazon VPC), and other functionality that enables mission owners to deploy applications and services with great flexibility, scalability, and reliability. AWS provides mission owners with the option to choose only the services they require and the ability to provision or release them as needed.

AWS Services

Compute

Amazon Elastic Compute Cloud (EC2)

Amazon EC2 is a web service that provides virtual server instances DoD customers can use to build and host software systems. Amazon EC2 facilitates web-scale computing by enabling mission owners to deploy virtual machines on demand. The simple web service interface allows mission owners to obtain and configure capacity with minimal friction, and it provides complete control over computing resources. Amazon EC2 changes the economics of computing by allowing organizations to avoid large capital expenditures and instead pay only for capacity that is actually used.

Amazon EC2 functionality and features:

- **Elastic:** Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing mission owners to quickly scale capacity, both up and down, as computing requirements change.

- **Flexible:** The mission owner can choose among various options for number of CPUs, memory size, and storage size. A highly reliable and fault-tolerant system can be built using multiple EC2 instances. EC2 instances are very similar to traditional hardware servers. EC2 instances use operating systems such as Linux, Windows, or OpenSolaris. They can accommodate most software that can run on those operating systems. EC2 instances have IP addresses, so the usual methods of interacting with a remote machine, such as Secure Shell (SSH) and Remote Desktop Protocol (RDP), can be used.
- **Amazon Machine Image (AMI):** AMI templates are used to define an EC2 server instance. Each AMI contains a software configuration, including operating system, application server, and applications applied to an instance type. Instance types in Amazon EC2 are essentially hardware archetypes matched to the amount of memory (RAM) and computing power (number of CPUs) needed for the application.

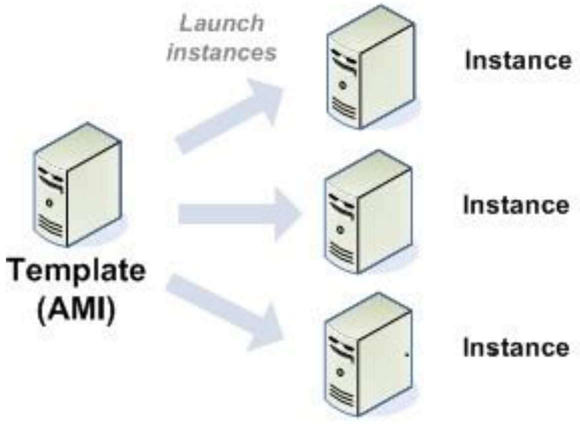

Figure 4: AMI Template

- **Custom AMI:** The first step toward building applications in AWS is to create a library of customized AMIs. Starting an application then becomes a matter of launching the AMI. For example, if an application is a website or web service, the AMI should be configured with a web server (e.g., Apache or Microsoft Internet Information Server), the associated static content, and the code for all dynamic pages. Alternatively, the AMI could be configured to install all required software components and content by running a bootstrap script as soon as the instance is launched. As a result, after launching the AMI, the web server will start and the application can begin accepting requests. After an AMI has been created, replacing a failing instance is very simple; a replacement instance can easily be launched that uses the same AMI as its template.

- **EC2 local instance store volumes:** These volumes provide temporary block-level storage for EC2 instances. When an EC2 instance is created from an AMI, in most cases, it comes with a preconfigured block of pre-attached disk storage. Unlike Amazon EBS volumes, data on instance store volumes persists only during the life of the associated EC2 instance, and they are not intended to be used as durable disk storage. Data on EC2 local instance store volumes is persistent across orderly instance reboots, but not in situations where the EC2 instance terminates or goes through a failure/restart cycle. *Local instance store volumes should not be used for any data that must persist over time, such as permanent file or database storage.* Although local instance store volumes are not persistent, the data can be persisted by periodically copying or backing it up to Amazon EBS or Amazon S3.
- **Mission-owner controlled:** Mission owners have complete control of their instances. They have root access to each one and can interact with them as they would any machine. Mission owners can stop their instance while retaining the data on a boot partition and then subsequently restart the same instance using web service APIs. Instances can be rebooted remotely using web service APIs. Mission owners also have access to the AWS Management Console to view and control their instances.
- **API Management:** Managing instances can be done through an API call, scriptable command line tools, or the AWS Management Console. Being able to quickly launch replacement instances based on a custom AMI is a critical first step towards fault tolerance. The next step is storing persistent data that these server instances use.
- **Multiple Availability Zones:** Amazon EC2 provides the ability to place instances in multiple Availability Zones. Availability Zones are distinct locations that are engineered to be insulated from failures in other Availability Zones. They provide inexpensive, low latency network connectivity to other zones in the same region. By launching instances in separate Availability Zones, mission owners can protect their applications from failure of a single location. Regions consist of one or more Availability Zones.
- **Reliable:** The Amazon EC2 Service Level Agreement (SLA) commitment is 99.95% availability for each EC2 region.
- **Elastic IP Addresses:** Elastic IP addresses are static IP addresses designed for dynamic cloud computing. An Elastic IP address is associated with a mission owner account and not with a particular instance. So mission owners control that address until they choose to explicitly release it. Unlike traditional static IP addresses, however, Elastic IP addresses can be programmatically remapped to any instance in their account in the event of an instance or Availability Zone failure. Mission owners don’t need to wait for a data technician to reconfigure or replace a host, or wait for the Domain Name System (DNS) to propagate. In addition, mission owners can optionally configure the reverse DNS record of any of their Elastic IP addresses. Note: Impact level 4 and 5 systems are prohibited from accessing the Internet

directly from an AWS VPC. As such, an Elastic IP address is only permitted for CC SRG Impact level 2 systems.

Scalability (Durability)

Auto Scaling is a web service that enables AWS users to automatically launch or terminate Amazon EC2 instances based on user-defined policies, health status checks, and schedules. Amazon EC2 instances are servers in the cloud. For applications configured to run on a cloud infrastructure, scaling is an important part of cost control and resource management. Scaling is the ability to increase or decrease the compute capacity of an application by either changing the number of servers (horizontal scaling) or changing the size of the servers (vertical scaling).

In a typical business situation, when the web application starts to get more traffic, the mission owner either adds more servers or increases the size of existing servers to handle the additional load. Similarly, if the traffic to the web application starts to slow down, the under-utilized servers can be terminated or the size of the existing servers can be decreased. Depending on the infrastructure involved, vertical scaling might involve changes to server configurations every time the application scales. With horizontal scaling, AWS simply increases or decreases the number of servers according to the application's demands. The decision when to scale vertically and when to scale horizontally depends on factors such as the mission owner’s use case, cost, performance, and infrastructure.

When using Auto Scaling, mission owners can increase the number of servers in use automatically when the user demand goes up to ensure that performance is maintained, and can decrease the number of servers when demand goes down to minimize costs. Auto Scaling helps make efficient use of compute resources by automatically doing the work of scaling for the mission owner. This automatic scaling is the core value of the Auto Scaling service.

Auto Scaling is well suited for applications that experience hourly, daily, or weekly variability in usage and need to automatically scale horizontally to keep up with changes in usage. Auto Scaling frees users from having to predict traffic spikes accurately and plan for provisioning resources in advance of them. With Auto Scaling, mission owners can build a fully scalable and affordable infrastructure in the cloud.

Networking

Amazon Virtual Private Cloud (VPC)

Network isolation and the ability to demonstrate separation of infrastructure and data is applicable at levels 2, 4 and 5, and it is a key requirement of the CC SRG for levels 4-5. AWS enables a mission owner to create the equivalent of a “Virtual Private Enclave” with the Amazon VPC service. Amazon VPC is used to provision a logically isolated section of the AWS Cloud where a customer can launch AWS resources in a virtual network that is defined by the mission owner.

This logically separate space within AWS is used to contain compute and storage resources that can be connected to a mission owner's existing infrastructure through a virtual private network (VPN) connection, AWS Direct Connect (private) connection, and/or the Internet. With Amazon VPC, it is then possible to extend existing DoD directory services, management tools, monitoring/security scanning solutions, and inspection capabilities, thus maintaining a consistent means of protecting information whether it is residing on internal DoD IT resources or in AWS. *Note: The CC SRG requires level 4-5 DoD applications to be connected to NIPRNet without direct Internet access from the VPC.*

AWS mission owners have complete control over the definition of the virtual networking environment within their VPC, including the selection of a private (RFC 1918) address range of their choice (e.g., 10.0.0.0/16), the creation of subnets, the configuration of route tables, and the inclusion or exclusion of network gateways. Further, mission owners can define the subnets within their VPC in a way that enables them to group similar kinds of instances based on IP address range.

Mission owners can use VPC functionality and features in the following ways:

- Mission owners can define a VPC on scalable infrastructure, and specify its private IP address range from any range they choose.
- Mission owners can sub-divide a VPC's private IP address further into one or more public or private subnets according to application requirements and security best practices. This can facilitate running applications and services in a customer's VPC.
- Mission owners define inbound and outbound access to and from individual subnets using network access control lists.
- Data can be stored in Amazon S3 with set permissions ensuring that the data can only be accessed from within a mission owner's VPC.
- An Elastic IP address can be attached to any instance in a mission owner's VPC so it can be reached directly from the Internet (CSM Impact level 2 only).
- A mission partner's VPC can be bridged with their onsite DoD IT infrastructure (encapsulated in an encrypted VPN connection) to extend existing security and management policies to the VPC instances as if they were running within the mission partner's physical infrastructure.

Amazon VPC provides advanced security features, such as security groups and network access control lists, to enable inbound and outbound filtering at the instance level and subnet level. When building a VPC, mission owners must define the subnets, routing rules, security groups, and network access control lists (ACLs) that comply with the networking and security requirements of the DoD and their parent organization.

Subnets

VPCs can span multiple Availability Zones. After creating a VPC, mission owners can add one or more subnets in each Availability Zone. Each subnet must reside entirely within one Availability Zone, cannot span zones, and is assigned a unique ID by AWS.

Routing

By design, each subnet must be associated with a route table that specifies the allowed routes for outbound traffic leaving the subnet. Every subnet is automatically associated with the main route table for the VPC. By updating the association, mission owners can change the contents of the main route table.

Mission owners should know the following basic things about VPC route tables:

- The VPC has an implicit router.
- The VPC comes with a main route table that mission owners can modify.
- Mission owners can create additional custom route tables for their VPC.
- Each subnet must be associated with a route table, which controls the routing for the subnet. If a mission owner does not associate a subnet with a particular route table, the subnet uses the main route table.
- Mission owners can replace the main route table with a custom table that they have created (this table becomes the default table each new subnet is associated with).
- Each route in a table specifies a destination Classless Inter-Domain Routing (CIDR) block and a target (for example, traffic destined for 172.16.0.0/12 is targeted for the virtual private gateway). Amazon VPC uses the most specific route that matches the traffic to determine how to route the traffic.

Security Groups and Network ACLs

AWS provides two features that mission owners can use to increase security in their VPC: security groups and network access control lists (ACLs). Both features enable mission owners to control the inbound and outbound traffic for their instances. Security groups work at the instance level, and network ACLs work at the subnet level. Security groups default to deny all, and must be configured by the mission owner to permit traffic.

Security groups provide stateful filtering at the instance level and can meet the network security needs of many AWS mission owners. However, VPC users can choose to use both security groups and network ACLs to take advantage of the additional layer of security that network ACLs provide.

An ACL is an optional layer of security that acts as a firewall for controlling traffic in and out of a subnet. Mission owners can set up network ACLs with rules similar to those implemented in security groups in order to add a layer of stateless filtering to their VPC.

Mission owners should know the following basic things about network ACLs:

- A network ACL is a numbered list of rules that is evaluated in order, starting with the lowest numbered rule, to determine whether traffic is allowed in or out of any subnet associated with the network ACL. The highest rule number available for use is 32766. We suggest that mission owners start by creating rules with rule numbers that are multiples of 100, so that new rules can be inserted later on.
- A network ACL has separate inbound and outbound rules, and each rule can either allow or deny traffic.
- Each VPC automatically comes with a modifiable default network ACL; by default, it allows all inbound and outbound traffic.
- Each subnet must be associated with a network ACL; if mission owners don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.
- Mission owners can create custom network ACLs; each custom network ACL starts out closed (permits no traffic) until the mission owner adds a rule.
- Network ACLs are stateless; responses to allow inbound traffic are subject to the rules for outbound traffic (and vice versa).

The following table summarizes the basic differences between security groups and network ACLs. Inbound traffic will first be processed according to the rules of the network ACL applied to a subnet, and subsequently by the security group applied at the instance level.

Security Group	Network ACL
Operates at the instance level (first layer of defense)	Operates at the subnet level (additional layer of defense)
Supports allow rules only	Supports allow rules and deny rules
Is stateful: Return traffic is automatically allowed, regardless of any rules	Is stateless: Return traffic must be explicitly allowed by rules
We evaluate all rules before deciding whether to allow traffic	We process rules in number order when deciding whether to allow traffic
Applies to an instance only if someone specifies the security group when launching the instance, or associates the security group with the instance later on	Automatically applies to all instances in the subnets it's associated with (backup layer of defense, so you don't have to rely on someone specifying the security group)

The following diagram illustrates the layers of security provided by security groups and network ACLs. For example, traffic from an Internet gateway is routed to the appropriate subnet using the routes in the routing table. The rules of the network ACL associated with the subnet control which traffic is allowed to the subnet. The rules of the security group associated with an instance control which traffic is allowed to the instance.

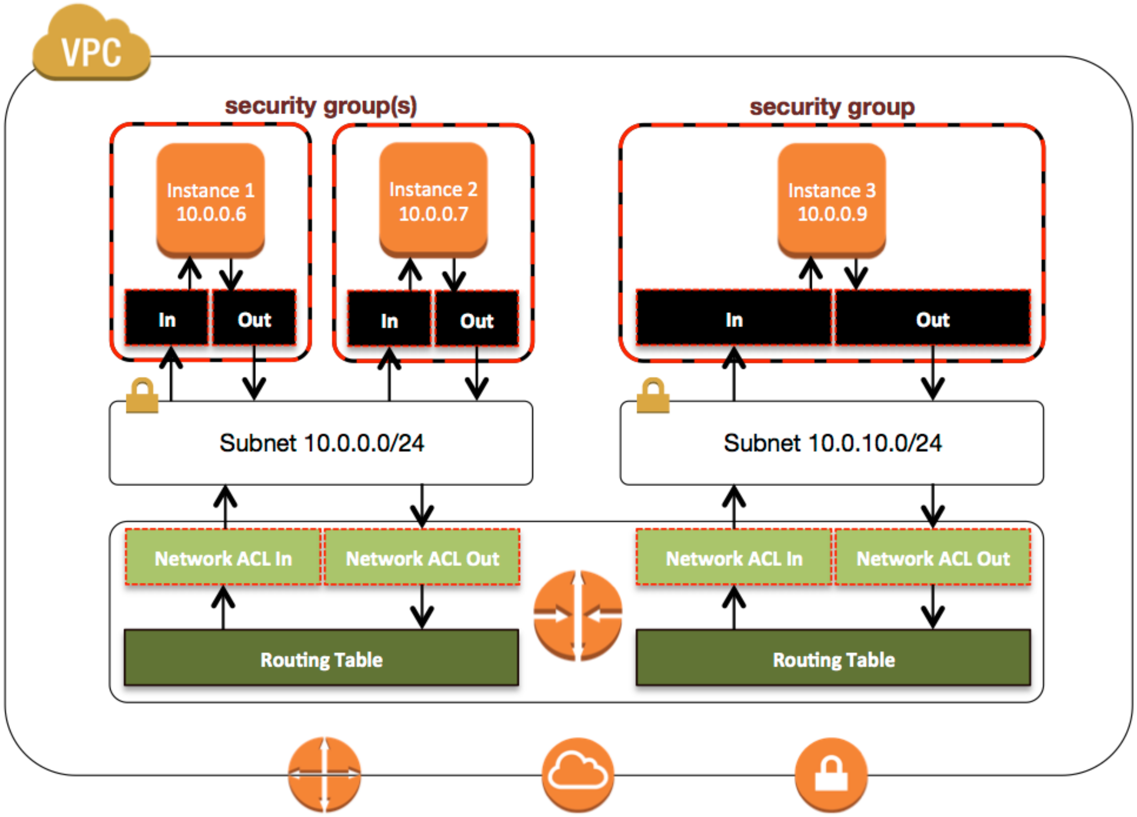

Figure 5: Layers of Security Using Security Groups and ACLs

Storage

Options

There are three different storage options for instances and/or resources that can be utilized in conjunction with a system hosted within an Amazon VPC. The three storage types are Amazon Simple Storage Service (S3), Amazon Elastic Block Store (EBS), and instance storage, each of which has distinct use cases.

Amazon S3

Amazon S3 is a highly durable repository designed for mission-critical and primary data storage for mission-owner data. It enables mission owners to store and retrieve any amount of data, at any time, from within Amazon EC2 or anywhere on the web.

Amazon S3 stores data objects redundantly on multiple devices across multiple facilities and allows concurrent read or write access to these data objects by many separate clients or application threads. S3 is designed to protect data and allow access to it even in the case of a failure of a data center. Additionally, mission owners can use the redundant data stored in Amazon S3 to recover quickly and reliably from instance or application failures. The Amazon S3 versioning feature allows the retention of prior versions of objects stored in S3 and also protects against accidental deletions initiated by staff or software error. Versioning can be enabled on any Amazon S3 bucket.

Mission owners should know the following basic things about Amazon S3 functionality and features:

- Mission owners can write, read, and delete objects containing from 1 byte to 5 terabytes of data each. The number of objects mission owners can store is unlimited.
- Each object is stored in an Amazon S3 bucket and retrieved via a unique, developer-assigned key.
- Objects stored in an AWS region never leave unless the mission owner transfers them out.
- Authentication mechanisms are provided to ensure that data is kept secure from unauthorized access. Objects can be made private or public, and rights can be granted to specific users.
- Options for secure data upload and download and encryption of data at rest are provided for additional data protection.
- Amazon S3 uses standards-based REST and SOAP interfaces designed to work with any Internet-development toolkit.
- Amazon S3 is built to be flexible so that protocol or functional layers can easily be added.
- Amazon S3 includes options for performing recurring and high volume deletions. For recurring deletions, rules can be defined to remove sets of objects after a pre-defined time period. For efficient one-time deletions, up to 1,000 objects can be deleted with a single request.

For more information on these Amazon S3 features, consult the Amazon Simple Storage Service (S3) documentation.[7]

Amazon Elastic Block Store

Amazon EBS provides block-level storage volumes for use with Amazon EC2 instances. Amazon EBS volumes are highly available and reliable storage volumes that can be

[7] http://aws.amazon.com/documentation/s3/

attached to any running instance that is in the same Availability Zone. Amazon EBS volumes that are attached to an Amazon EC2 instance are exposed as storage volumes that persist independently from the life of the instance. With Amazon EBS, users pay only for what they use.

Amazon EBS is recommended when data changes frequently and requires long-term persistence. Amazon EBS volumes are particularly well-suited for use as the primary storage for file systems, databases, or for any applications that require fine granular updates and access to raw, unformatted, block-level storage. Amazon EBS is particularly helpful for database-style applications that frequently encounter many random reads and writes across the data set.

Mission owners can attach multiple volumes to the same instance within the limits specified by their AWS account. Currently, an AWS account is limited to 5,000 Amazon EBS volumes, as well as 20 TiB of total storage.

Amazon EBS volumes store data redundantly, making them more durable than a typical hard drive. The annual failure rate for an Amazon EBS volume is 0.1% to 0.5%, compared to 4% for a commodity hard drive.

Amazon EBS and Amazon EC2 are often used in conjunction with one another when building an application on the AWS platform. Any data that needs to persist can be stored on EBS volumes, not on the temporary storage associated with each EC2 instance. If the EC2 instance fails and needs to be replaced, the EBS volume can simply be attached to the new EC2 instance. Because this new instance is a duplicate of the original, there is no loss of data or functionality.

EBS volumes are highly reliable, but to further mitigate the possibility of a failure, backups of these volumes can be created using a feature called snapshots. A robust backup strategy will include an interval between backups, a retention period, and a recovery plan. Snapshots are stored for high-durability in Amazon S3. Snapshots can be used to create new EBS volumes, which are an exact copy of the original volume at the time the snapshot was taken. These EBS operations can be performed through API calls.

Mission owners should know the following basic things about Amazon EBS functionality and features:

- Amazon EBS allows mission owners to create storage volumes from 1 GB to 16 TB that can be mounted as devices by EC2 instances. Multiple volumes can be mounted to the same instance.
- Storage volumes behave like raw, unformatted block devices, with user supplied device names and a block device interface. Mission owners can create a file system

on top of EBS volumes, or use them in any other way they would use a block device (like a hard drive).

- Amazon EBS volumes are placed in a specific Availability Zone, and can then be attached to instances also in that same Availability Zone.
- Each storage volume is automatically replicated within the same Availability Zone. This prevents data loss due to failure of any single hardware component.
- Amazon EBS also provides the ability to create point-in-time snapshots of volumes, which are persisted to Amazon S3. These snapshots can be used as the starting point for new EBS volumes, and protect data for long-term durability. The same snapshot can be used to instantiate as many volumes as desired.
- For more information on these Amazon EBS features, please consult the Amazon Elastic Block Store (EBS) documentation.[8]

Instance Storage

An instance store provides volatile, temporary block-level storage for use with an EC2 instance, and consists of one or more instance store volumes. Instance store volumes must be configured using block device mapping at launch time and mounted on the running instance before they can be used. Instances launched from an instance store-backed AMI have a mounted instance store volume for the virtual machine's root device volume, and can have other mounted instances store volumes, depending on the instance type.

The data in an instance store is temporary and only persists during the lifetime of its associated instance. If an instance reboots (intentionally or unintentionally), data in the instance store persists. However, data on instance store volumes is lost under the following circumstances:

- Failure of an underlying drive
- Stopping an Amazon EBS-backed instance
- Terminating an instance

Therefore, AWS mission owners should not rely on instance store volumes for important, long-term data. Instead, keep data safe by using a replication strategy across multiple instances storing data in Amazon S3, or using Amazon EBS volumes.

Encryption

AWS supports multiple encryption mechanisms for data stored within a mission owner's VPC. The following is a summary of the encryption methods:

[8] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html

Amazon EBS encryption: For EBS volumes, encryption is managed by OS-level encryption (e.g., BitLocker or Encrypted File System (EFS)), by third-party products, or by Amazon EBS encryption. For EBS encryption, when customers create an encrypted EBS volume and attach it to a supported instance type, the data stored at rest on the volume, the disk I/O, and the snapshots created from the volume are all encrypted. The encryption occurs on the servers that host Amazon EC2 instances, providing encryption of data in transit from EC2 instances to EBS storage.

There are AWS partner solutions that can help automate the process of encrypting Amazon EBS volumes, as well as supply and protect the necessary keys. AWS partners provide products that encrypt Amazon EBS volumes and include a key management infrastructure (KMI). These products are available on the AWS Marketplace[9].

Amazon S3 encryption: Provides added security for object data stored in buckets in Amazon S3. Mission owners can encrypt data on the client side and upload the encrypted data to Amazon S3. In this case, mission owners manage the encryption process, the encryption keys, and related tools. Optionally, mission owners can use the Amazon S3 server-side encryption feature. Amazon S3 encrypts object data before saving it on disks in its data centers, and it decrypts the object data when objects are downloaded, freeing mission owners from the tasks of managing encryption, encryption keys, and related tools. Mission owners can also use their own encryption keys with the Amazon S3 server-side encryption feature.

AWS CloudHSM: The AWS CloudHSM service provides tamper-resistant hardware security module (HSM) appliances that are designed to comply with international (Common Criteria EAL4+) and U.S. Government (NIST FIPS 140-2) standards. This encryption service utilizes SafeNet Luna hardware devices that reside within AWS data centers, providing an encryption management capability to mission owners that can reside within their VPC. If used, mission owners retain full control of encryption keys and cryptographic operations on the HSM, while Amazon manages and maintains the hardware without having access to mission owner keys.

Management

AWS Identity and Access Management (IAM)

IAM is a web service that enables mission owners to manage users and user permissions in AWS. The service is targeted at organizations with multiple users or systems that use products such as Amazon EC2, Amazon Relational Database Service (RDS), and the AWS Management Console. With IAM, mission owners can centrally manage users, security credentials such as access keys, and permissions that control which AWS resources users can access.

[9] https://aws.amazon.com/marketplace

Without IAM, organizations with multiple users and systems must either create multiple AWS accounts, each with its own billing and subscriptions to AWS products, or employees must all share the security credentials of a single AWS account. Also, without IAM, mission owners have no control over the tasks a particular user or system can do and what AWS resources they might use. IAM addresses this issue by enabling organizations to create multiple users (each user is a person, system, or application) who can use AWS products, each with individual security credentials, all controlled by and billed to a single AWS account. With IAM, each user is allowed to do only what they need to do as part of the user's job.

IAM includes the following features:

- *Central control of users and security credentials*—Mission owners control creation, rotation, and revocation of each user's AWS security credentials (such as access keys).
- *Central control of user access*—Mission owners control what data users can access and how they access it.
- *Shared resources*—Users can share data for collaborative projects.
- *Permissions based on organizational groups*—Mission owners can restrict users' AWS access based on their job duties (for example, admin, developer, etc.) or departments. When users move inside the organization, mission owners can easily update their AWS access to reflect the change in their role.
- *Central control of AWS resources*—A mission owner’s organization maintains central control of the data the users create, with no breaks in continuity or lost data as users move around within or leave the organization.
- *Control over resource creation*—Mission owners can help make sure that users create data only in sanctioned places.
- *Networking controls*—Mission owners can restrict user access to AWS resources to only from within the organization's corporate network, using SSL.

Level 2 Sample Reference Architecture

SRG impact level 2 systems are appropriate for hosting public or limited access information. Level 2 systems are not required to be fully segregated from Internet traffic, and they can connect directly to the Internet. The following is a sample reference architecture for an impact level 2 system with a recovery time objective (RTO) of greater than or equal to 1 day.

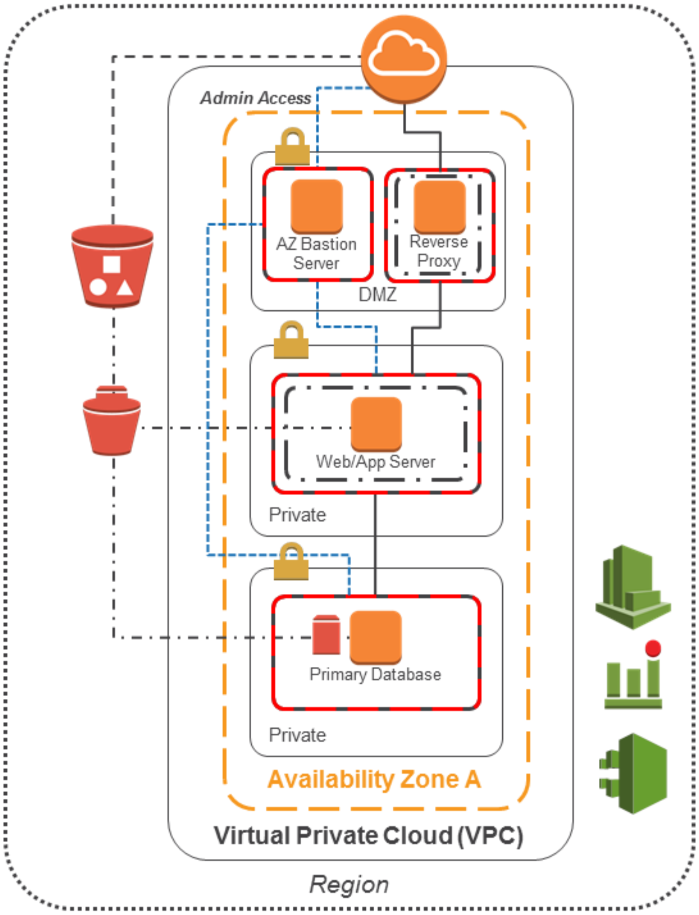

Figure 6: Sample Impact Level 2 Architecture with RTO >= 1 day(s)

The following is a sample reference architecture for an impact level 2 system with a recovery time objective (RTO) of greater than or equal to 1 hour.

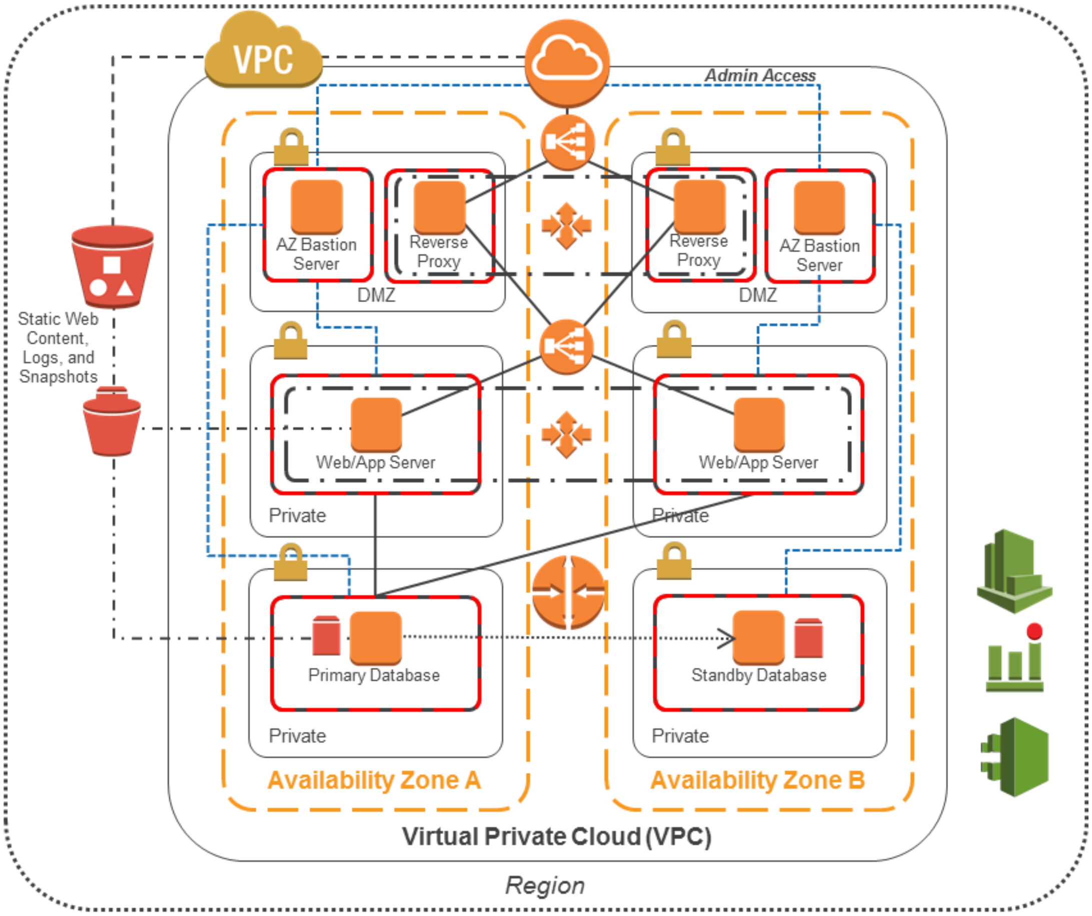

Figure 7: Sample Impact Level 2 Architecture with RTO >= 1 hour

Key Attributes:

- Access to/from the Internet traverses an Internet Gateway (IGW).
- A layer 7 reverse web proxy (residing in a DMZ) is included for protection against application-level attacks targeting web infrastructures.
- Each impact level 2 infrastructure should be adequately stratified to limit access to the web/application and database assets to either authorized traffic (by strata), or to administrative traffic initiated from an authorized bastion host contained within the infrastructure.
- Static web-addressable content is stored in secured Amazon S3 buckets (using bucket policies) and directly addressable from the Internet.
- Infrastructure backups, images, and volume snapshots are securely stored in the Amazon S3 infrastructure, but they are not publically addressable from the Internet.

By default, the AWS infrastructure operates in a “zero trust” security model. Access to an instance, regardless of the strata on which it resides, must be implicitly allowed. The enforcement of this model is enabled through the use of security groups (SG), which are addressable by other security groups. For administrative access to any instance in the infrastructure, the use of a bastion host is defined as the only host instance that is authorized to access infrastructure assets within a designated infrastructure. These hosts are typically, Windows Server instances (RDP via port 3389), Remote Desktop Gateway servers, and/or Linux instances for SSH access to Linux hosts. Any instance designated as a bastion host should be included in a bastion security group. This should be the only security group granted access to the reverse web proxy, web/application instances, and database instances (via ports 22 and/or 3389). Additionally, to further bolster the defensive posture of the infrastructure, the bastion host(s) should be powered off when administration activities are not being performed.

The following table is a sample summary of security group behavior by traffic flow:

Traffic From (SG)	**Traffic To (SG)**	**Security Group Action**
Internet	Reverse web proxy (*reverse-proxy-SG*)	Allow 80/443 from Internet (all)
Reverse web proxy (*reverse-proxy-SG*)	Web/application server(s) (*web-server-SG*)	Allow 80/443 from *reverse-proxy-SG*
Web/application server(s) (*web-server-SG*)	Database server(s) (db-server-SG)	Allow appropriate database port
Administrator (Internet, trusted admin IP)	Bastion host (*bastion-host-SG*)	Allow 3389/22 from trusted remote administration host (host IP address range)
Bastion host (*bastion-host-SG*)	Proxy, web, application, database instances (*reverse-proxy-SG, web-server-SG, db-server-SG*)	Allow 3389/22 from *bastion-host-SG*

For systems requiring an RTO of less than or equal to 1 hour, it is recommended to deploy the application in a highly available infrastructure design. The following reference architecture is an example of how to both meet application RTO requirements and maintain CC SRG compliance.

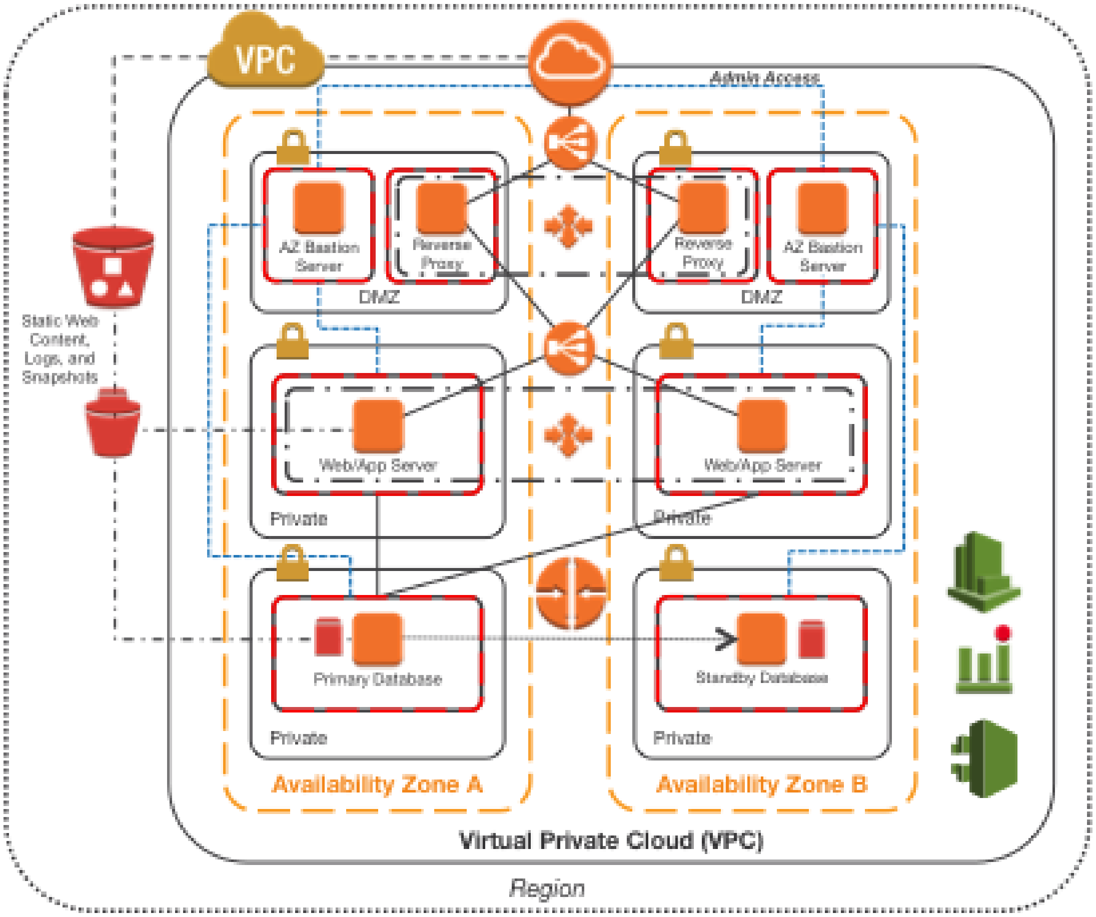

Figure 8: Sample Impact Level 2 Architecture with RTO <= 1 hour

Key Attributes:

- Access to/from the Internet traverses an Internet Gateway (IGW).
- The layer 7 reverse web proxy (residing in a DMZ) is deployed in an Auto Scaling group across multiple Availability Zones.
- Web/application instances are deployed in an Auto Scaling group across multiple Availability Zones.
- Application database is replicated to a read-only instance residing in a secondary Availability Zone.
- Static web-addressable content is stored in secured Amazon S3 buckets (using bucket policies) and is directly addressable from the Internet.
- Infrastructure backups, images, and volume snapshots are securely stored in the Amazon S3 infrastructure, but they are not publically addressable from the Internet.

Level 4-5 Sample Reference Architecture

DoD systems hosting data categorized at levels 4 and 5 of the CC SRG must attain complete separation from systems hosting non-DoD data, and route traffic entirely through dedicated connections to the DoD information networks (DoDIN) through a VPN or an AWS Direct Connect connection. To achieve full separation of network traffic, the current approved DoD reference architecture is to establish an AWS Direct Connect connection from DoDIN to AWS, including BCAP with a Computer Network Defense (CND) Suite hosted in a colocation facility associated with AWS.

The following illustration is a sample reference architecture for an impact level 4-5 system.

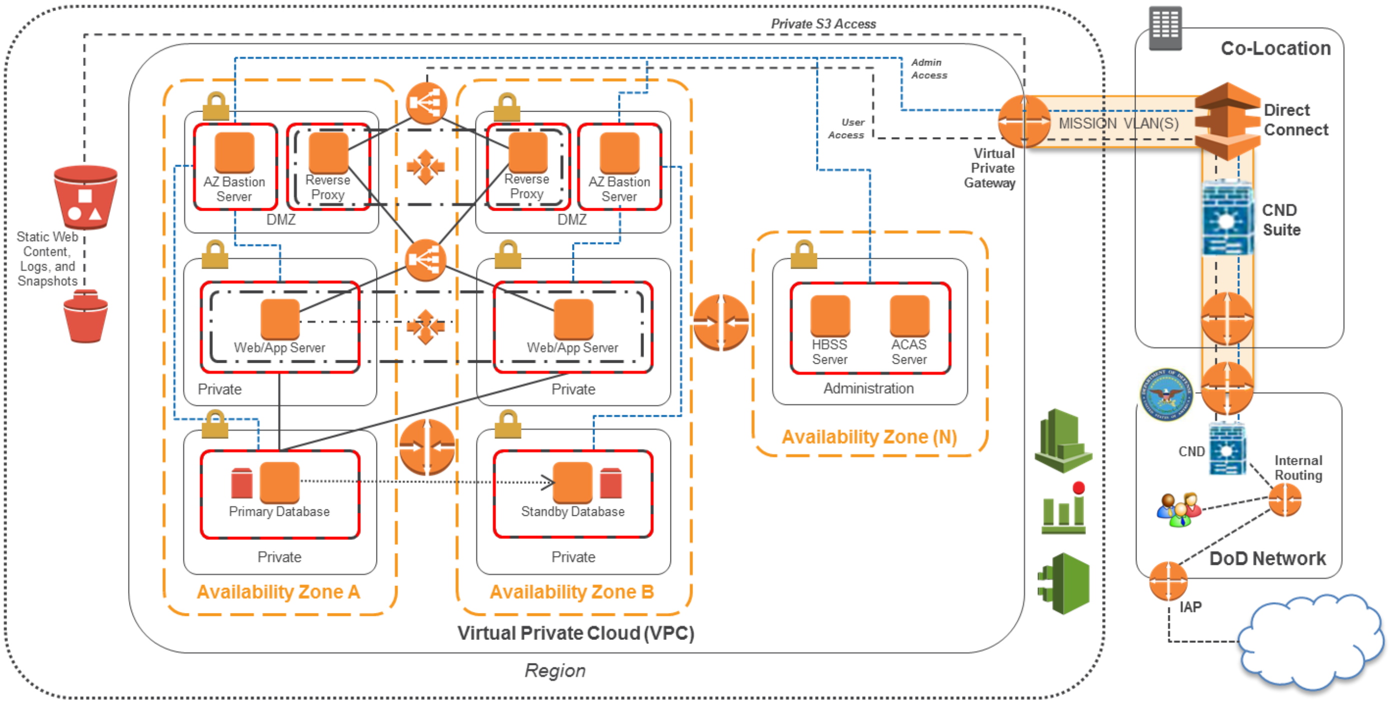

Figure 9: Sample Impact Level 4-5 Architecture

The following list contains the CSM Reference Architecture requirements for impact levels 4-5 that are added to those already defined for impact level 2:

- *No direct access to/from the public Internet*–All traffic in/out of AWS must traverse the DoDIN through a Virtual Private Gateway.
- *Connection to the DoDIN*–This can be accomplished through the use of AWS Direct Connect, IPsec VPN, and/or a combination of the two. All traffic traversing between DoDIN and the DoD application must use a BCAP.

- *Access to Amazon S3 is restricted to AWS Direct Connect*–Access to Amazon S3 is, while Internet addressable by default, only accessible through a private route introduced as part of the AWS Direct Connect service.
- *All traffic to/from the VPC is scanned on DoDIN*–All traffic entering and/or exiting the Amazon VPC is required to pass through a hardware-based Computer Network Defense suite of tools. This infrastructure is both owned and operated by the government (or on behalf of the government by a Mission Partner organization).
- *Host Based Security System servers (HBSS) are deployed in the VPC*–All DoD EC2 instances will have HBSS installed, and they will communicate with a McAfee ePolicy Orchestrator (ePo) server hosted in the management subnet, or on-premises.
- *Assured Compliance Assessment Solution (ACAS) tool is deployed in the VPC*–All DoD instances will be scanned by an ACAS tool that is located in the management subnet, with full access to the subnets of the VPC.

Conclusion

AWS provides a number of important benefits to DoD mission owners, including flexibility, elasticity, utility billing, and reduced time-to-market. It provides a range of security services and features that that you can use to manage the security of your assets and data in the AWS. Although AWS provides an excellent service management layer for infrastructure or platform services, mission owners are still responsible for protecting the confidentiality, integrity, and availability of their data in the cloud, and for meeting specific mission requirements for information protection.

Conventional security and compliance concepts still apply in the cloud. Using the various best practices highlighted in this whitepaper, we encourage you to build a set of security policies and processes for your organization so you can deploy applications and data.

Further Reading

For additional help, please consult the following sources:

- Amazon Elastic Compute Cloud (EC2) Documentation [10]
- Amazon Virtual Private Cloud (VPC) Documentation [11]
- Amazon Simple Storage Service (S3) Documentation [12]
- Amazon Elastic Block Store (EBS) Documentation [13]

[10] http://aws.amazon.com/documentation/ec2/

[11] http://aws.amazon.com/documentation/vpc/

[12] http://aws.amazon.com/documentation/s3/

- AWS Identity and Access Management (IAM)Documentation[14]
- AWS Direct Connect Documentation[15]
- Amazon Security Center[16]

[13] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html

[14] http://aws.amazon.com/documentation/iam/

[15] http://aws.amazon.com/documentation/direct-connect/

[16] http://aws.amazon.com/security/