# Architecting for Genomic Data Security and Compliance in AWS

## Working with Controlled-Access Datasets from dbGaP, GWAS, and other Individual-Level Genomic Research Repositories

*Angel Pizarro*
*Chris Whalley*

*December 2014*

# Table of Contents

# Overview

Researchers who plan to work with genomic sequence data on Amazon Web Services (AWS) often have questions about security and compliance; specifically about how to meet guidelines and best practices set by government and grant funding agencies such as the National Institutes of Health. In this whitepaper, we review the current set of guidelines, and discuss which services from AWS you can use to meet particular requirements and how to go about evaluating those services.

# Scope

This whitepaper focuses on common issues raised by Amazon Web Services (AWS) customers about security best practices for human genomic data and controlled access datasets, such as those from National Institutes of Health (NIH) repositories like Database of Genotypes and Phenotypes (dbGaP) and genome-wide association studies (GWAS). Our intention is to provide you with helpful guidance that you can use to address common privacy and security requirements. However, we caution you not to rely on this whitepaper as legal advice for your specific use of AWS. We strongly encourage you to obtain appropriate compliance advice about your specific data privacy and security requirements, as well as applicable laws relevant to your human research projects and datasets.

# Considerations for Genomic Data Privacy and Security in Human Research

Research involving individual-level genotype and phenotype data and de-identified controlled access datasets continues to increase. The data has grown so fast in volume and utility that the availability of adequate data processing, storage, and security technologies has become a critical constraint on genomic research. The global research community is recognizing the practical benefits of the AWS cloud, and scientific investigators, institutional signing officials, IT directors, ethics committees, and data access committees must answer privacy and security questions as they evaluate the use of AWS in connection with individual-level genomic data and other controlled access datasets. Some common questions include: Are data protected on secure servers? Where are data located? How is access to data controlled? Are data protections appropriate for the Data Use Certification?

These considerations are not new and are not cloud-specific. Whether data reside in an investigator lab, an institutional network, an agency-hosted data repository or within the AWS cloud, the essential considerations for human genomic data are the same. You must correctly implement data protection and security controls in the system by first defining the system requirements and then architecting the system security controls to meet those requirements, particularly the shared responsibilities amongst the parties who use and maintain the system.

# AWS Approach to Shared Security Responsibilities

AWS delivers a robust web services platform with features that enable research teams around the world to create and control their own private area in the AWS cloud so they can quickly build, install, and use their data analysis applications and data stores without having to purchase or maintain the necessary hardware and facilities. As a researcher, you can create your private AWS environment yourself using a self-service signup process that establishes a unique AWS account ID, creates a root user account and account ID, and provides you with access to the AWS Management Console and Application Programming Interfaces (APIs), allowing control and management of the private AWS environment.

Because AWS does not access or manage your private AWS environment or the data in it, you retain responsibility and accountability for the configuration and security controls you implement in your AWS account. This *customer accountability* for your private AWS environment is fundamental to understanding the respective roles of AWS and our customers in the context of data protections and security practices for human genomic data. Figure 1 depicts the AWS Shared Responsibility Model.

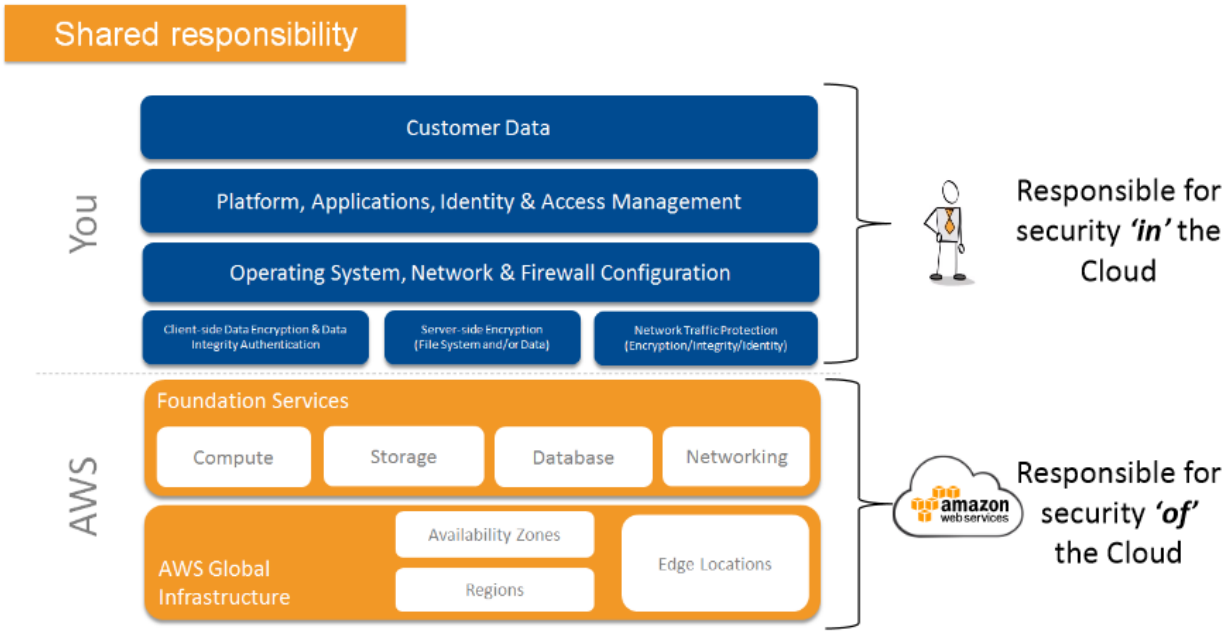



*Figure 1 - Shared Responsibility Model*

In order to deliver and maintain the features available within every customer's private AWS environment, AWS works vigorously to enhance the security features of the platform and ensure that the feature delivery operations are secure and of high quality. AWS defines quality and security as *confidentiality*, *integrity*, and *availability* of our services, and AWS seeks to provide researchers with visibility and assurance of our quality and security practices in four important ways.

First, AWS infrastructure is designed and managed in alignment with a set of internationally recognized security and quality accreditations, standards and best-practices, including industry standards ISO 27001, ISO 9001, AT 801 and 101 (formerly SSAE 16), as well as government standards NIST, FISMA, and FedRAMP. Independent third parties perform accreditation assessments of AWS. These third parties are auditing experts in cloud computing environments and each brings a unique perspective from their compliance backgrounds in a wide range of industries including healthcare, life sciences, financial services, government and defense, and others. Because each accreditation carries a unique audit schedule, including continuous monitoring, AWS security and quality controls are constantly audited and improved for the benefit of all AWS customers including those with dbGaP, HIPAA, and other health data protection requirements.

Second, AWS provides transparency by making these ISO, SOC, FedRAMP and other compliance reports available to customers upon request. Customers can use these reports to evaluate AWS for their particular needs. You can request AWS compliance reports at https://aws.amazon.com/compliance/contact, and you can find more information on AWS compliance certifications, customer case studies, and alignment with best practices and standards at the AWS compliance website, http://aws.amazon.com/compliance/.

Third, as a controlled U.S. subsidiary of Amazon.com Inc., Amazon Web Services Inc. participates in the Safe Harbor program developed by the U.S. Department of Commerce, the European Union, and Switzerland, respectively. Amazon.com and its controlled U.S. subsidiaries have certified that they adhere to the Safe Harbor Privacy Principles agreed upon by the U.S., the E.U. and Switzerland, respectively. You can view the Safe Harbor certification for Amazon.com and its controlled U.S. subsidiaries on the U.S. Department of Commerce's Safe Harbor website. The Safe Harbor Principles require Amazon and its controlled U.S. subsidiaries to take reasonable precautions to protect the personal information that our customers give us in order to create their account. This certification is an illustration of our dedication to security, privacy, and customer trust.

Lastly, AWS respects the rights of our customers to have a choice in their use of the AWS platform. The AWS Account Management Console and Customer Agreement are designed to ensure that every customer can stop using the AWS platform and export all their data at any time and for any reason. This not only helps customers maintain control of their private AWS environment from creation to deletion, but it also ensures that AWS must continuously work to earn and keep the trust of our customers.

# Architecting for Compliance with dbGaP Security Best Practices in AWS

A primary principle of the dbGaP security best practices is that researchers should download data to a secure computer or server and not to unsecured network drives or servers.[1] The remainder of the dbGaP security best practices can be broken into a set of three IT security control domains that you must address to ensure that you meet the primary principle:

[1] http://www.ncbi.nlm.nih.gov/projects/gap/pdf/dbgap_2b_security_procedures.pdf

- **Physical Security** refers to both physical access to resources, whether they are located in a data center or in your desk drawer, and to remote administrative access to the underlying computational resources.
- **Electronic Security** refers to configuration and use of networks, servers, operating systems, and application-level resources that hold and analyze dbGaP data.
- **Data Access Security** refers to managing user authentication and authorization of access to the data, how copies of the data are tracked and managed, and having policies and processes in place to manage the data lifecycle.

Within each of these control domains are a number of control areas, which are summarized in Table 1.

*Table 1 - Summary of dbGaP Security Best Practices*

| Control Domain | Control Areas |
|---|---|
| Physical Security | Deployment Model<br>Data Location<br>Physical Server Access<br>Portable Storage Media |
| Electronic Security | User Accounts, Passwords, and Access Control Lists<br>Internet, Networking, and Data Transfers<br>Data Encryption<br>File Systems and Storage Volumes<br>Operating Systems and Applications<br>Auditing, Logging And Monitoring |
| Data Access Security | Authorizing Access to Data<br>Cleaning Up Data and Retaining Results |

The remainder of this paper focuses on the control areas involved in architecting for security and compliance in AWS.

# Deployment Model

A basic architectural consideration for dbGaP compliance in AWS is determining whether the system will run entirely on AWS or as a *hybrid deployment* with a mix of AWS and non-AWS resources. This paper focuses on the control areas for the AWS resources. If you are architecting for hybrid deployments, you must also account for your non-AWS resources, such as the local workstations you might download data to and from your AWS environment, any institutional or external networks you connect to your AWS environment, or any third-party applications you purchase and install in your AWS environment.

# Data Location

The AWS cloud is a globally available platform in which you can choose the geographic region in which your data is located. AWS data centers are built in clusters in various global regions. AWS calls these data center clusters *Availability zones* (AZs). As of December 2014, AWS

maintains 28 AZs organized into 11 regions globally. As an AWS customer you can choose to use one region, all regions, or any combination of regions using built-in features available within the AWS Management Console.

AWS regions and Availability Zones ensure that if you have location-specific requirements or regional data privacy policies, you can establish and maintain your private AWS environment in the appropriate location. You can choose to replicate and back up content in more than one region, but you can be assured that AWS does not move customer data outside the region(s) you configure.

## Physical Server Access

Unlike traditional laboratory or institutional server systems where researchers install and control their applications and data directly on a specific physical server, the applications and data in a private AWS account are decoupled from a specific physical server. This decoupling occurs through the built-in features of the AWS Foundation Services layer (see Figure 1 - Shared Responsibility Model) and is a key attribute that differentiates the AWS cloud from traditional server systems or even traditional server virtualization. Practically, this means that every resource (virtual servers, firewalls, databases, genomic data, etc.) within your private AWS environment is reduced to a single set of software files that are orchestrated by the Foundational Services layer across multiple physical servers. Even if a physical server fails, your private AWS resources and data maintain confidentiality, integrity, and availability. This attribute of the AWS cloud also adds a significant measure of security, because even if someone were to gain access to a single physical server, they would not have access to all the files needed to recreate the genomic data within the your private AWS account.

AWS owns and operates its physical servers and network hardware in highly-secure, state-of-the-art data centers that are included in the scope of independent third-party security assessments of AWS for ISO 27001, Service Organization Controls 2 (SOC 2), NIST's federal information system security standards, and other security accreditations. Physical access to AWS data centers and hardware is based on the least privilege principle, and access is authorized only for essential personnel who have experience in cloud computing operating environments and who are required to maintain the physical environment. When individuals are authorized to access a data center they are not given logical access to the servers within the data center. When anyone with data center access no longer has a legitimate need for it, access is immediately revoked even if they remain an employee of Amazon or Amazon Web Services.

Physical entry into AWS data centers is controlled at the building perimeter and ingress points by professional security staff who use video surveillance, intrusion detection systems, and other electronic means. Authorized staff must pass two-factor authentication a minimum of two times to enter data center floors, and all physical access to AWS data centers is logged, monitored, and audited routinely.

## Portable Storage Media

The decision to run entirely on AWS or in a hybrid deployment model has an impact on your system security plans for portable storage media. Whenever data are downloaded to a portable device, such as a laptop or smartphone, the data should be encrypted and hardcopy printouts controlled. When genomic data are stored or processed in AWS, customers can encrypt their

data, but there is no portable storage media to consider because all AWS customer data resides on controlled storage media covered under AWS's accredited security practices. When controlled storage media reach the end of their useful life, AWS procedures include a decommissioning and media sanitization process that is designed to prevent customer data from being exposed to unauthorized individuals. AWS uses the techniques detailed in DoD 5220.22-M ("National Industrial Security Program Operating Manual") or NIST 800-88 ("Guidelines for Media Sanitization") to destroy data as part of the decommissioning process. All decommissioned magnetic storage devices are degaussed and physically destroyed in accordance with industry-standard practices.

For more information, see *Overview of Security Processes*.[2]

## User Accounts, Passwords, and Access Control Lists

Managing user access under dbGaP requirements relies on a principle of least privilege to ensure that individuals and/or processes are granted only the rights and permissions to perform their assigned tasks and functions, but no more.[3] When you use AWS, there are two types of user accounts that you must address:

- Accounts with direct access to AWS resources, and
- Accounts at the operating system or application level.

Managing user accounts with direct access to AWS resources is centralized in a service called AWS Identity and Access Management (IAM). After you establish your root AWS account using the self-service signup process, you can use IAM to create and manage additional users and groups within your private AWS environment. In adherence to the least privilege principle, new users and groups have no permissions by default until you associate them with an IAM policy. IAM policies allow access to AWS resources and support fine-grained permissions allowing operation-specific access to AWS resources. For example, you can define an IAM policy that restricts an Amazon S3 bucket to read-only access by specific IAM users coming from specific IP addresses. In addition to the users you define within your private AWS environment, you can define IAM roles to grant temporary credentials for use by externally authenticated users or applications running on Amazon EC2 servers.

Within IAM, you can assign users individual credentials such as passwords or access keys. Multi-factor authentication (MFA) provides an extra level of user account security by prompting users to enter an additional authentication code each time they log in to AWS. dbGaP also requires that users not share their passwords and recommends that researchers communicate a written password policy to any users with permissions to controlled access data. Additionally, dbGaP recommends certain password complexity rules for file access. IAM provides robust features to manage password complexity, reuse, and reset rules.

How you manage user accounts at the operating system or application level depends largely on which operating systems and applications you choose. For example, applications developed specifically for the AWS cloud might leverage IAM users and groups, whereas you'll need to assess and plan the compatibility of third-party applications and operating systems with IAM on a case-by-case basis. You should always configure password-enabled screen savers on any

[2] http://media.amazonwebservices.com/pdf/AWS_Security_Whitepaper.pdf
[3] http://www.ncbi.nlm.nih.gov/projects/gap/pdf/dbgap_2b_security_procedures.pdf

local workstations that you use to access your private AWS environment, and configure virtual server instances within the AWS cloud environment with OS-level password-enabled screen savers to provide an additional layer of protection.

More information on IAM is available in the IAM documentation and IAM Best Practices guide, as well as on the Multi-Factor Authentication page.

# Internet, Networking, and Data Transfers

The AWS cloud is a set of web services delivered over the Internet, but data within each customer's private AWS account is not exposed directly to the Internet unless you specifically configure your security features to allow it. This is a critical element of compliance with dbGaP security best practices, and the AWS cloud has a number of built-in features that prevent direct Internet exposure of genomic data.

Processing genomic data in AWS typically involves the Amazon Elastic Compute Cloud (Amazon EC2). Amazon EC2 is a service you can use to create virtual server instances that run operating systems like Linux and Microsoft Windows. When you create new Amazon EC2 instances for downloading and processing genomic data, by default those instances are accessible only by authorized users within the private AWS account. The instances are not discoverable or directly accessible on the Internet unless you configure them otherwise. Additionally, genomic data within an Amazon EC2 instance resides in the operating system's file directory, which requires that you set OS-specific configurations before any data can be accessible outside of the instance. When you need clusters of Amazon EC2 instances to process large volumes of data, a Hadoop framework service called Amazon Elastic MapReduce (Amazon EMR) allows you to create multiple, identical Amazon EC2 instances that follow the same basic rule of least privilege unless you change the configuration otherwise.

Storing genomic data in AWS typically involves object stores and file systems like Amazon Simple Storage Service (Amazon S3) and Amazon Elastic Block Store (Amazon EBS), as well as database stores like Amazon Relational Database Service (Amazon RDS), Amazon Redshift, Amazon DynamoDB, and Amazon ElastiCache. Like Amazon EC2, all of these storage and databases services default to least privilege access and are not discoverable or directly accessible from the Internet unless you configure them to be so.

Individual compute instances and storage volumes are the basic building blocks that researchers use to architect and build genomic data processing systems in AWS. Individually these building blocks are private by default and networking them together within the AWS environment can provide additional layers of security and data protections. Using Amazon Virtual Private Cloud (Amazon VPC), you can create private, isolated networks within the AWS cloud where you retain complete control over the virtual network environment, including definition of the IP address range, creation of subnets, and configuration of network route tables and network gateways. Amazon VPC also offers stateless firewall capabilities through the use of Network Access Control Lists (NACLs) that control the source and destination network traffic endpoints and ports, giving you robust security controls that are independent of the computational resources launched within Amazon VPC subnets. In addition to the stateless firewalling capabilities of Amazon VPC NACLs, Amazon EC2 instances and some services are launched within the context of AWS Security Groups. Security groups define network-level stateful firewall rules to protect computational resources at the Amazon EC2 instance or service

layer level. Using security groups, you can lock down compute, storage, or application services to strict subsets of resources running within an Amazon VPC subnet, adhering to the principal of least privilege.

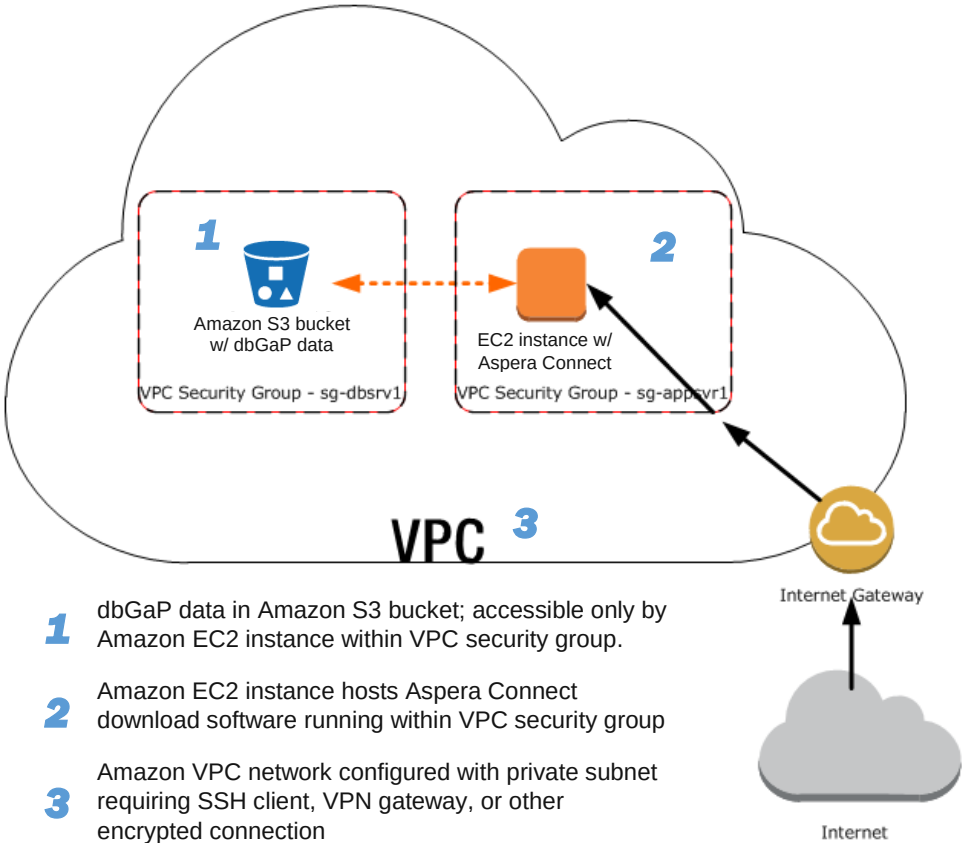



*Figure 2 - Protecting data from direct Internet access using Amazon VPC*

In addition to networking and securing the virtual infrastructure within the AWS cloud, Amazon VPC provides several options for connecting to your AWS resources. The first and simplest option is providing secure public endpoints to access resources, such as SSH bastion servers. A second option is to create a secure Virtual Private Network (VPN) connection that uses Internet Protocol Security (IPSec) by defining a *virtual private gateway* into the Amazon VPC. You can use the connection to establish encrypted network connectivity over the Internet between an Amazon VPC and your institutional network.

Lastly, research institutions can establish a dedicated and private network connection to AWS using AWS Direct Connect. AWS Direct Connect lets you establish a dedicated, high-bandwidth (1 Gbps to 10 Gbps) network connection between your network and one of the AWS Direct Connect locations. Using industry standard 802.1q VLANs, this dedicated connection can be partitioned into multiple virtual interfaces, allowing you to use the same connection to access public resources such as objects stored in Amazon S3 using public IP address space, and private resources such as Amazon EC2 instances running within an Amazon Virtual Private Cloud (Amazon VPC) using private IP space, while maintaining network separation between the public and private environments. You can reconfigure virtual interfaces at any time to meet your changing needs.

Using a combination of hosted and self-managed services, you can take advantage of secure, robust networking services within a VPC and secure connectivity with another trusted network. To learn more about the finer details, see our Amazon VPC whitepaper, the Amazon VPC documentation, and the Amazon VPC Connectivity Options Whitepaper.

# Data Encryption

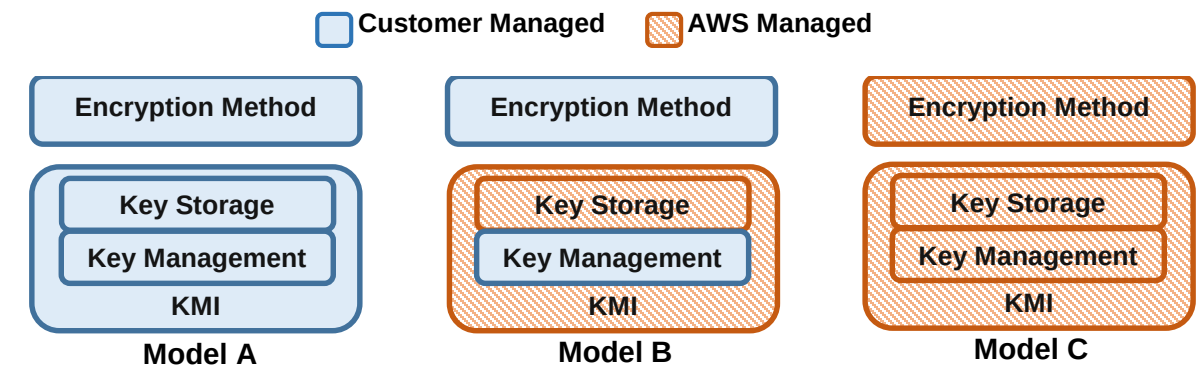
Encrypting data in-transit and at rest is one of the most common methods of securing controlled access datasets. As an Internet-based service provider, AWS understands that many institutional IT security policies consider the Internet to be an insecure communications medium and, consequently, AWS has invested considerable effort in the security and encryption features you need in order to use the AWS cloud platform for highly sensitive data, including protected health information under HIPAA and controlled access genomic datasets from the National Institutes of Health (NIH). AWS uses encryption in three areas:

- Service management traffic
- Data within AWS services
- Hardware security modules

As an AWS customer, you use the AWS Management Console to manage and configure your private environment. Each time you use the AWS Management Console an SSL/TLS[4] connection is made between your web browser and the console endpoints. Service management traffic is encrypted, data integrity is authenticated, and the client browser authenticates the identity of the console service endpoint using an X.509 certificate. After this encrypted connection is established, all subsequent HTTP traffic, including data in transit over the Internet, is protected within the SSL/TLS session. Each AWS service is also enabled with application programming interfaces (APIs) that you can use to manage services either directly from applications or third-party tools, or via Software Development Kits (SDK), or via AWS command line tools. AWS APIs are web services over HTTPS and protect commands within an SSL/TLS encrypted session.

Within AWS there are several options for encrypting genomic data, ranging from completely automated AWS encryption solutions (server-side) to manual, client-side options. Your decision to use a particular encryption model may be based on a variety of factors, including the AWS service(s) being used, your institutional policies, your technical capability, specific requirements of the data use certificate, and other factors. As you architect your systems for controlled access datasets, it's important to identify each AWS service and encryption model you will use with the genomic data. There are three different models for how you and/or AWS provide the encryption method and work with the key management infrastructure (KMI), as illustrated in Figure 3.

[4] Secure Sockets Layer (SSL)/Transport Layer Security (TLS)




*Researcher manages the encryption method and entire KMI.*

*Researcher manages the encryption method; AWS provides storage component of KMI while researcher provides management layer of KMI.*

*AWS manages the encryption method and the entire KMI.*

*Figure 2 - Encryption Models in AWS*

In addition to the client-side and server-side encryption features built-in to many AWS services, another common way to protect keys in a KMI is to use a dedicated storage and data processing device that performs cryptographic operations using keys on the devices. These devices, called hardware security modules (HSMs), typically provide tamper evidence or resistance to protect keys from unauthorized use. For researchers who choose to use AWS encryption capabilities for your controlled access datasets, the AWS CloudHSM service is another encryption option within your AWS environment, giving you use of HSMs that are designed and validated to government standards (NIST FIPS 140-2) for secure key management.

If you want to manage the keys that control encryption of data in Amazon S3 and Amazon EBS volumes, but don't want to manage the needed KMI resources either within or external to AWS, you can leverage the AWS Key Management Service (AWS KMS). AWS Key Management Service is a managed service that makes it easy for you to create and control the encryption keys used to encrypt your data, and uses HSMs to protect the security of your keys. AWS Key Management Service is integrated with other AWS services including Amazon EBS, Amazon S3, and Amazon Redshift. AWS Key Management Service is also integrated with AWS CloudTrail, discussed later, to provide you with logs of all key usage to help meet your regulatory and compliance needs. AWS KMS also allows you to implement key creation, rotation, and usage policies. AWS KMS is designed so that no one has access to your master keys. The service is built on systems that are designed to protect your master keys with extensive hardening techniques such as never storing plaintext master keys on disk, not persisting them in memory, and limiting which systems can connect to the device. All access to update software on the service is controlled by a multi-level approval process that is audited and reviewed by an independent group within Amazon.

As mentioned in the Internet, Network, and Data Transfer section of this paper, you can protect data transfers to and from your AWS environment to an external network with a number of encryption-ready security features, such as VPN.

For more information about encryption options within the AWS environment, see Securing Data at Rest with Encryption, as well as the AWS CloudHSM product details page. To learn more about how AWS KMS works you can read the AWS Key Management Service whitepaper[5].

# File Systems and Storage Volumes

Analyzing and securing large datasets like whole genome sequences requires a variety of storage capabilities that allow you to make use of that data. Within your private AWS account, you can configure your storage services and security features to limit access to authorized users. Additionally, when research collaborators are authorized to access the data, you can configure your access controls to safely share data between your private AWS account and your collaborator's private AWS account.

When saving and securing data within your private AWS account, you have several options. Amazon Web Services offers two flexible and powerful storage options. The first is Amazon Simple Storage Service (Amazon S3), a highly scalable web-based object store. Amazon S3 provides HTTP/HTTPS REST endpoints to upload and download data objects in an Amazon S3 bucket. Individual Amazon S3 objects can range from 1 byte to 5 terabytes. Amazon S3 is designed for 99.99% availability and 99.999999999% object durability, thus Amazon S3 provides a highly durable storage infrastructure designed for mission-critical and primary data storage. The service redundantly stores data in multiple data centers within the Region you designate, and Amazon S3 calculates checksums on all network traffic to detect corruption of data packets when storing or retrieving data. Unlike traditional systems, which can require laborious data verification and manual repair, Amazon S3 performs regular, systematic data integrity checks and is built to be automatically self-healing.

Amazon S3 provides a base level of security, whereby default-only bucket and object owners have access to the Amazon S3 resources they create. In addition, you can write security policies to further restrict access to Amazon S3 objects. For example, dbGaP recommendations call for all data to be encrypted while the data are in flight. With an Amazon S3 bucket policy, you can restrict an Amazon S3 bucket so that it only accepts requests using the secure HTTPS protocol, which fulfills this requirement. Amazon S3 bucket policies are best utilized to define broad permissions across sets of objects within a single bucket. The previous examples for restricting the allowed protocols or source IP ranges are indicative of best practices. For data that need more variable permissions based on whom is trying to access data, IAM user policies are more appropriate. As discussed previously, IAM enables organizations with multiple employees to create and manage multiple users under a single AWS account. With IAM user policies, you can grant these IAM users fine-grained control to your Amazon S3 bucket or data objects contained within.

Amazon S3 is a great tool for genomics analysis and is well suited for analytical applications that are purpose-built for the cloud. However, many legacy genomic algorithms and applications cannot work directly with files stored in a HTTP-based object store like Amazon S3, but rather need a traditional file system. In contrast to the Amazon S3 object-based storage approach,

[5] https://d0.awsstatic.com/whitepapers/KMS-Cryptographic-Details.pdf

Amazon Elastic Block Store (Amazon EBS) provides network-attached storage volumes that can be formatted with traditional file systems. This means that a legacy application running in an Amazon EC2 instance can access genomic data in an Amazon EBS volume as if that data were stored locally in the Amazon EC2 instance. Additionally, Amazon EBS offers whole-volume encryption without the need for you to build, maintain, and secure your own key management infrastructure. When you create an encrypted Amazon EBS volume and attach it to a supported instance type, data stored at rest on the volume, disk I/O, and 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 Amazon EC2 instances to Amazon EBS storage. Amazon EBS encryption uses AWS Key Management Service (AWS KMS) Customer Master Keys (CMKs) when creating encrypted volumes and any snapshots created from your encrypted volumes. The first time you create an encrypted Amazon EBS volume in a region, a default CMK is created for you automatically. This key is used for Amazon EBS encryption unless you select a CMK that you created separately using AWS Key Management Service. Creating your own CMK gives you more flexibility, including the ability to create, rotate, disable, define access controls, and audit the encryption keys used to protect your data. For more information, see the AWS Key Management Service Developer Guide.

There are three options for Amazon EBS volumes:

- **Magnetic volumes** are backed by magnetic drives and are ideal for workloads where data are accessed infrequently, and scenarios where the lowest storage cost is important.
- **General Purpose (SSD) volumes** are backed by Solid-State Drives (SSDs) and are suitable for a broad range of workloads, including small- to medium-sized databases, development and test environments, and boot volumes.
- **Provisioned IOPS (SSD) volumes** are also backed by SSDs and are designed for applications with I/O-intensive workloads such as databases. Provisioned IOPs offer storage with consistent and low-latency performance, and support up to 30 IOPS per GB, which enables you to provision 4,000 IOPS on a volume as small as 134 GB. You can also achieve up to 128MBps of throughput per volume with as little as 500 provisioned IOPS. Additionally, you can stripe multiple volumes together to achieve up to 48,000 IOPS or 800MBps when attached to larger Amazon EC2 instances.

While general-purpose Amazon EBS volumes represent a great value in terms of performance and cost, and can support a diverse set of genomics applications, you should choose which Amazon EBS volume type to use based on the particular algorithm you're going to run. A benefit of scalable on-demand infrastructure is that you can provision a diverse set of resources, each tuned to a particular workload.

For more information on the security features available in Amazon S3, see the Access Control and Using Data Encryption topics in the *Amazon S3 Developer Guide*. For an overview on security on AWS, including Amazon S3, see *Amazon Web Services: Overview of Security Processes*. For more information about Amazon EBS security features, see Amazon EBS Encryption and Amazon Elastic Block Store (Amazon EBS).

# Operating Systems and Applications

Recipients of controlled-access data need their operating systems and applications to follow predefined configuration standards. Operating systems should align with standards, such as

NIST 800-53, dbGaP Security Best Practices Appendix A, or other regionally accepted criteria. Software should also be configured according to application-specific best practices, and OS and software patches should be kept up-to-date. When you run operating systems and applications in AWS, you are responsible for configuring and maintaining your operating systems and applications, as well as the feature configurations in the associated AWS services such as Amazon EC2 and Amazon S3.

As a concrete example, imagine that a security vulnerability in the standard SSL/TLS shared library is discovered. In this scenario, AWS will review and remediate the vulnerability in the foundation services (see Figure 1), and you will review and remediate the operating systems and applications, as well as any service configuration updates needed for hybrid deployments.

You must also take care to properly configure the OS and applications to restrict remote access to the instances and applications. Examples include locking down security groups to only allow SSH or RDP from certain IP ranges, ensuring strong password or other authentication policies, and restricting user administrative rights on OS and applications.

## Auditing, Logging, and Monitoring

Researchers who manage controlled access data are required to report any inadvertent data release in accordance with the terms in the Data Use Certification, breach of data security, or other data management incidents contrary to the terms of data access.

The dbGaP security recommendations recommend use of security auditing and intrusion detection software that regularly scans and detects potential data intrusions. Within the AWS ecosystem, you have the option to use built-in monitoring tools, such as Amazon CloudWatch, as well as a rich partner ecosystem of security and monitoring software specifically built for AWS cloud services. The AWS Partner Network lists a variety of system integrators and software vendors that can help you meet security and compliance requirements. For more information, see the AWS Life Science Partner webpage[6].

Amazon CloudWatch is a monitoring service for AWS cloud resources and the applications you run on AWS. You can use Amazon CloudWatch to collect and track metrics, collect and monitor log files, and set alarms. Amazon CloudWatch provides performance metrics on the individual resource level, such as Amazon EC2 instance CPU load, and network IO, and sets up thresholds on these metrics to raise alarms when the threshold is passed. For example, you can set an alarm to detect unusual spikes in network traffic from an Amazon EC2 instance that may be an indication of a compromised system. CloudWatch alarms can integrate with other AWS services to send the alerts simultaneously to multiple destinations. Example methods and destinations might include a message queue in Amazon Simple Queuing Service (Amazon SQS) which is continuously monitored by watchdog processes that will automatically quarantine a system; a mobile text message to security and operations staff that need to react to immediate threats; an email to security and compliance teams who audit the event and take action as needed.

Within Amazon CloudWatch you can also define custom metrics and populate these with whatever information is useful, even outside of a security and compliance requirement. For instance, an Amazon CloudWatch metric can monitor the size of a data ingest queue to trigger

[6] http://aws.amazon.com/partners/competencies/life-sciences/

the scaling up (or down) of computational resources that process data to handle variable rates of data acquisition.

AWS CloudTrail and AWS Config are two services that enable you to monitor and audit all of the operations against the AWS product API's. AWS CloudTrail is a web service that records AWS API calls for your account and delivers log files to you. The recorded information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by the AWS service. With AWS CloudTrail, you can get a history of AWS API calls for your account, including API calls made via the AWS Management Console, AWS SDKs, command line tools, and higher-level AWS services (such as AWS CloudFormation). The AWS API call history produced by AWS CloudTrail enables security analysis, resource change tracking, and compliance auditing.

AWS Config builds upon the functionality of AWS CloudTrail, and provides you with an AWS resource inventory, configuration history, and configuration change notifications to enable security and governance. With AWS Config you can discover existing AWS resources, export a complete inventory of your AWS resources with all configuration details, and determine how a resource was configured at any point in time. These capabilities enable compliance auditing, security analysis, resource change tracking, and troubleshooting.

Lastly, AWS has implemented various methods of external communication to support all customers in the event of security or operational issues that may impact our customers. Mechanisms are in place to allow the customer support team to be notified of operational and security issues that impact each customer's account. The AWS incident management team employs industry-standard diagnostic procedures to drive resolution during business-impacting events within the AWS cloud platform. The operational systems that support the platform are extensively instrumented to monitor key operational metrics, and alarms are configured to automatically notify operations and management personnel when early warning thresholds are crossed on those key metrics. Staff operators provide 24 x 7 x 365 coverage to detect incidents and to manage their impact and resolution. An on-call schedule is used so that personnel are always available to respond to operational issues.

## Authorizing Access to Data

Researchers using AWS in connection with controlled access datasets must only allow authorized users to access the data. Authorization is typically obtained either by approval from the Data Access Committee (DAC) or within the terms of the researcher's existing Data Use Certification (DUC).

Once access is authorized, you can grant that access in one or more ways, depending on where the data reside and where the collaborator requiring access is located. The scenarios below cover the situations that typically arise:

- Provide the collaborator access within an AWS account via an IAM user (see User Accounts, Passwords and Access Control Lists)
- Provide the collaborator access to their own AWS accounts (see File Systems, Storage Volumes, and Databases)
- Open access to the AWS environment to an external network (see Internet, Networking, and Data Transfers)

## Cleaning Up Data and Retaining Results

Controlled-access datasets for closed research projects should be deleted upon project close-out, and only encrypted copies of the minimum data needed to comply with institutional policies should be retained. In AWS, deletion and retention operations on data are under the complete control of a researcher. You might opt to replicate archived data to one or more AWS regions for disaster recovery or high-availability purposes, but you are in complete control of that process.

As it is for on-premises infrastructure, data provenance[7] is the sole responsibility of the researcher. Through a combination of data encryption and other standard operating procedures, such as resource monitoring and security audits, you can comply with dbGaP security recommendations in AWS.

With respect to AWS storage services, after Amazon S3 data objects or Amazon EBS volumes are deleted, removal of the mapping from the public name to the object starts immediately, and is generally processed across the distributed system within several seconds. After the mapping is removed, there is no remote access to the deleted object. The underlying storage area is then reclaimed for use by the system.

# Conclusion

The AWS cloud platform provides a number of important benefits and advantages to genomic researchers and enables them to satisfy the NIH security best practices for controlled access datasets. While AWS delivers these benefits and advantages through our services and features, researchers are still responsible for properly building, using, and maintaining the private AWS environment to help ensure the confidentiality, integrity, and availability of the controlled access datasets they manage. Using the practices in this whitepaper, we encourage you to build a set of security policies and processes for your organization so you can deploy applications using controlled access data quickly and securely.



[7] The process of tracing and recording the origins of data and its movement between databases.