

# HIPAA on AWS

Thursday, June 23, 2016 12:27 PM

## Introduction

The purpose of this page is to describe a HIPAA-compliant data system built on AWS including operational guidelines. A comparable effort will -- we hope -- come about with Azure and other public cloud providers.

## Terminology

Health Insurance Portability and Accountability Act

- [https://en.wikipedia.org/wiki/Health\\_Insurance\\_Portability\\_and\\_Accountability\\_Act](https://en.wikipedia.org/wiki/Health_Insurance_Portability_and_Accountability_Act)

HC = HIPAA Compliant

Herein:

- Tools do not come in contact with PHI. 'triggers and orchestration'
- Services may come in contact with PHI. 'data and compute'

The Nine Technologies under the AWS BAA that are HIPAA-aligned:

- S3
- EC2
- RDS
- EBS
- DynamoDB
- EMR
- ELB
- Glacier
- Redshift

Non-Nine but still want to use

- Lambda
- VPC

## Questions, Social Rules

Does data on an encrypted drive that moves through an encrypted link to another encrypted drive count as 'ok'? The alternative is to encrypt it first (that is, the bytes of the file are also encrypted) so that if it were to take a wrong turn and land on a public un-encrypted drive it would still be encrypted.

Filenames may not include PHI. Hence there is an obligation on the MRs to follow this and/or build it into file generation.

What about gov-cloud?

Cris Ewell is the UW IT Security person.

## Story

Data under custody of a medical researcher is present in some native system ('on premise'). It may reside on an encrypted drive and/or it may be encrypted itself. See the Questions section to notice that this poses our first open question.

We would like to move this data to the cloud, notice it, process it, and return PHI back to the medical researcher. We would like this to happen in a fully automated manner; but the initiation of the process and the completion of the process are described here as a manual narrative. To wit:

The research machine connects through an encrypted connection to an S3 bucket on AWS. The data are

pushed across as a set of files with file names that are anonymized. The AWS system 'notifies' the arrival of this data and a chain reaction is initiated: Virtual machines start up inside a Virtual Private Cloud. Data are pulled from S3 into these machines and processed. Notifications are sent on initiation and at termination. Results are written in a different S3 bucket. A notification goes back to the medical researcher. Everything shuts down so that no residual data are present in the VPC (outside of the S3 buckets).

To make this happen we must first build the system. From a technical perspective it would be helpful to hear this story re-told in terms of that preparation phase.

Kilroy diagram needed

## Technical Story

- I have a home system called Homer. On here I install encryption software, e.g. 'PGP'. This produces a key file called HomerKey.
- I go to AWS and create a VPC
- On the VPC I create a public-facing Bastion Server BS
  - This has an Elastic ip address which will persist
- Inside the VPC I create a public and a private subnet using a Routing Table.
- I set up a NAT Gateway with Routing Table.
- I place on the private subnet a small Dedicated EC2 instance called Edgar.
- The BS has only port 22 open (ssh)
- The BS uses Secure Groups on AWS to limit access to certain URLs.
- Using ssh / WinSCP I copy HomerKey to the BS. From there again I move HomerKey to Edgar.
- I create an AMI to do processing. Call this BusyBee.
- I create a Role that allows an entity (a BusyBee) to read from and write to S3.
- I set up an Ansible-assisted process for configuring and running jobs on EC2 instances.
- I set up S3 buckets for input and output.
  - The input buckets only accept http PUT. No GET or LIST.
  - The S3 buckets have a VPC Endpoint included: Terminates inside our VPC.
- I set up a DynamoDB table: It keeps track of the names of uploaded files.
- I set up a Lambda service
  - Triggered by 'new object in bucket' in the S3 input bucket
  - The Lambda is managed using a role
- Kilroy a longer version of this story includes a database.
- I wait.
- The Medical Researcher (MR) pushes data to S3.
  - File names must not contain PHI. This goes to social rules, see below.
  - Uses a 3rd party app such as Cloudberry
  - Uses the AWS CLI
  - Uses an API call (this is a programmatic way to keep PHI out of file names)
- Lambda notices this, appends the file names to DynamoDB and pings the BS.
- The BS tries to run a new processing job
  - Launch five Dedicated EC2 BusyBees
    - Assign them S3 access role
    - BusyBees have encrypted volumes
    - These volumes come with the pipeline pre-installed
      - However the pipeline may be update...
        - ◆ ...and the AMI could subsequently be updated as well
    - These EBS volumes come pre-loaded with reference data (non-PHI) that the pipeline uses
    - Here are Sheena's notes on this topic
      - Bastion Server Setup
        1. Get worker public key from S3 (which allows us to ssh from bastion into worker if need be)

2. Create SQS queue of all objects (samples) listed in S3 bucket
  3. Kick off a worker instance for each message in the queue
- Worker Setup and Execution
    1. Grab latest pipeline code, install
    2. Create EBS Genomes volume from snapshot
    3. Grab message from SQS which is really a file for analysis in S3
    4. Grab fastq files from S3.
    5. Run analysis
    6. Write message to done queue.
    7. If last instance running, grab certain analysis files from S3 to create run level output
    8. SNS topic notifies me when last instance shuts down.
  - Run Ansible script to configure BusyBees (patch, get data file names from DynamoDB table, etcetera)
  - Get BusyBees the HomerKey from Edgar
  - The BusyBees send an Alert through the NAT gateway to Simple Notification Service (SNS) which uses something called SES to send an email 'system is working on PHI data'.
  - BusyBees pull data from S3 using VPC Endpoint; thanks to the Route table
  - BusyBees decrypt data using HomerKey
  - BusyBees process their data into result files: Encrypted EBS volume.
  - Optionally the result files are encrypted in place in the EBS volume.
  - Through the VPC Endpoint the results are moved to S3.
  - The BusyBees send an Alert through the NAT gateway to Simple Notification Service (SNS) which uses something called SES to send another email: 'We're done now.'
  - BusyBees evaporate completely leaving no trace.
  - BS returns to quiescent state.
  - The MR can go to the results S3 bucket using a secure pipe (e.g. 3rd party application 'Cloudberry') to pull results.
    - If the data are encrypted for the return trip then there is another decryption step.
  - The system could also accommodate some housekeeping mechanisms

## In Practice

### Start a VPC

Let's start on the AWS console, choose VPC and start one up.

Kilroy left off here on a editing pass Sep 15 2016.

## Networking



Let's do this.

AWS

Services

Edit

VPC Dashboard

Filter by VPC:

None

Virtual Private Cloud

Your VPCs

Subnets

Create VPC

Actions

Search VPCs and their props

	Name	VPC ID	State
<input type="checkbox"/>	dbaccess	vpc-bf5737db	available
<input type="checkbox"/>		vpc-d4f8e4b1	available

Create!

Create VPC

A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. Use the Classless Inter-Domain Routing (CIDR) block format to specify your VPC's contiguous IP address range, for example, 10.0.0.0/16. You cannot create a VPC larger than /16.

Name tag

czarhipaa

CIDR block

10.0.0.0/16

Tenancy

Dedicated

Cancel

Yes, Create

czarhipaa should be unique

CIDR as shown is typical

Dedicated Instance means: Nobody else allowed here. Kilroy elaborate what this means please.

<input type="checkbox"/>	Name	VPC ID	State	VPC CIDR	DHCP options set	Route table	Network ACL	Tenancy	Default VPC
<input type="checkbox"/>	dbaccess	vpc-bf5737db	available	10.0.0.0/16	dopt-4c890e29	rtb-e915448d  ...	acl-0a443a6e	Default	No
<input checked="" type="checkbox"/>	czarhipaa	vpc-f98cf79d	available	10.0.0.0/16	dopt-4c890e29	rtb-553a0f31	acl-d487d4b0	Dedicated	No

Now it exists; time to fill it with stuff.

### Pro Tip

**You can be more cost-effective by not making this Dedicated but then your PHI-Using instances will have to be launched Dedicated. This is carte blanche Dedicated and so is more expensive. We do not consider this option in this tutorial because we are erring on the side of caution rather than cost.**

Next: Create a subnet

## VPC Dashboard

Filter by VPC:

None

### Virtual Private Cloud

Your VPCs

Subnets

Create Subnet

Search Subn

Name

Database pri

Database pri

## Create Subnet

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

Name tag public

VPC vpc-f98cf79d (10.0.0.0/16) | czarhipaa

Availability Zone us-west-2a

CIDR block 10.0.0.0/24

Cancel Yes, Create

Public subnet addresses will be of the form: 10.0.0.2, .3, .4, ... .254  
Take note of the AZ:

public	subnet-2592fa41	available	vpc-f98cf79d (10.0.0.0/16)   czarh...	10.0.0.0/24	251	us-west-2a	rtb-553a0f31	acl-d487d4b0	No
--------	-----------------	-----------	---------------------------------------	-------------	-----	------------	--------------	--------------	----

We could do multiple public sub-nets by creating more than one in multiple Azs; that is a big-time concept (kilroy).

Kilroy it would behoove us to have a short primer or pointer or both on ip address formatting.

Now the private one:

## Create Subnet



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

Name tag

VPC

Availability Zone

CIDR block

Cancel

Yes, Create

<input checked="" type="checkbox"/>	private	subnet-4f93fb2b	available	vpc-f98cf79d (10.0.0.0/16)   czarh...	10.0.1.0/24	251	us-west-2a	rtb-553a0f31	acl-d487d4b0	No
<input type="checkbox"/>	public	subnet-2592fa41	available	vpc-f98cf79d (10.0.0.0/16)   czarh...	10.0.0.0/24	251	us-west-2a	rtb-553a0f31	acl-d487d4b0	No

## VPC Dashboard

Filter by VPC:

## Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Create Internet Gateway

☐ Name



## Create Internet Gateway



An Internet gateway is a virtual router that connects a VPC to the Internet.

Name tag

Cancel

Yes, Create

Attach it to the VPC:

Create Internet GatewayDeleteAttach to VPCDetach from VPC

Search Internet Gateways and X

<input type="checkbox"/>	Name	ID	State	VPC
<input type="checkbox"/>		igw-e7befd82	attached	vpc-d4f8e4b1 (172.31.0.0/16)
<input type="checkbox"/>		igw-2cd1cb49	attached	vpc-bf5737db (10.0.0.0/16)   dbac...
<input checked="" type="checkbox"/>	czarhipaa_igw	igw-2afc174e	detached	

Create Internet GatewayDeleteAttach to VPCDetach from VPC

Search Internet Gateways and X

<input type="checkbox"/>	Name	ID	State	VPC
<input type="checkbox"/>		igw-e7befd82	attached	vpc-d4f8e4b1 (172.31.0.0/16)
<input type="checkbox"/>		igw-2cd1cb49	attached	vpc-bf5737db (10.0.0.0/16)   dbac...
<input checked="" type="checkbox"/>	czarhipaa_igw	igw-2afc174e	attached	vpc-f98cf79d (10.0.0.0/16)   czarh...

Now for the NAT Gateway

VPC Dashboard

Filter by VPC:  
None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Create NAT Gateway

Filter by attributes or search

Choose the public one in czarhipaa

## Create a NAT Gateway

Create a NAT gateway and assign it an Elastic IP address. [Learn more](#)

Subnet\*

Search subnets by ID or name or VPC e.g. 'subnet-1a2b3c' ⓘ

Elastic IP Allocation ID\*

Subnet	VPC
subnet-0cb12868 (10.0.1.0/24)   Database privat...	vpc-bf5737db (10.0.0.0/16)   dbaccess
subnet-30a91946 (10.0.2.0/24)   Database privat...	vpc-bf5737db (10.0.0.0/16)   dbaccess
subnet-c8d030ac (172.31.16.0/20)	vpc-d4f8e4b1 (172.31.0.0/16)
subnet-4f93fb2b (10.0.1.0/24)   private	vpc-f98cf79d (10.0.0.0/16)   czarhipaa
subnet-2592fa41 (10.0.0.0/24)   public	vpc-f98cf79d (10.0.0.0/16)   czarhipaa
subnet-6702503e (172.31.0.0/20)	vpc-d4f8e4b1 (172.31.0.0/16)
subnet-0db12869 (10.0.0.0/24)   Database access	vpc-bf5737db (10.0.0.0/16)   dbaccess
subnet-14959063 (172.31.32.0/20)	vpc-d4f8e4b1 (172.31.0.0/16)

## Create a NAT Gateway

Create a NAT gateway and assign it an Elastic IP address. [Learn more](#)

Subnet\*

subnet-2592fa41 ⓘ

Elastic IP Allocation ID\*

eipalloc-ae7d40ca ⓘ

Create New EIP ⓘ


New EIP (52.32.64.42) creation successful.

Cancel

Create a NAT Gateway

Notice we Created a New EIP

## Create a NAT Gateway



**Your NAT gateway has been created.**

**Note:** In order to use your NAT gateway, ensure that you edit your route tables to include a route with a target of 'nat-06c541083712a04f7'.

[Find out more.](#)

View NAT Gateways

Edit Route Tables

Now the Route Table



## VPC Dashboard

Filter by VPC:

None

## Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Create Route Table

Delete

Search Route Tables

Name

Public Route Table

### Create Route Table

×

A route table specifies how packets are forwarded between the subnets within your VPC, the Internet, and your VPN connection.

Name tag czarhipaa\_public\_routetable

i

VPC vpc-f98cf79d (10.0.0.0/16) | czarhipaa

i

Cancel

Yes, Create

### Create Route Table

×

A route table specifies how packets are forwarded between the subnets within your VPC, the Internet, and your VPN connection.

Name tag czarhipaa\_private\_routetable

i

VPC vpc-f98cf79d (10.0.0.0/16) | czarhipaa

i

Cancel

Yes, Create

Here they are (including the default main one):

Create Route Table

Delete Route Table

Set As Main Table

<input type="checkbox"/>	Name	Route Table ID	Explicitly Associat	Main	VPC
<input type="checkbox"/>		rtb-553a0f31	0 Subnets	Yes	vpc-f98cf79d (10.0.0.0/16)   czarh...
<input type="checkbox"/>	czarhipaa_public_routet	rtb-00261364	0 Subnets	No	vpc-f98cf79d (10.0.0.0/16)   czarh...
<input checked="" type="checkbox"/>	czarhipaa_private_route	rtb-e0261384	0 Subnets	No	vpc-f98cf79d (10.0.0.0/16)   czarh...

## rtb-00261364 | czarhipaa\_public\_routetable

Summary

**Routes**

Subnet Associations

Route Propagation

Tags

Edit

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Edit and modify as shown:

rtb-00261364 | czarhipaa\_public\_routetable

Summary

**Routes**

Subnet Associations

Route Propagation

Tags

Cancel

Save

Destination	Target	Status	Propagated	Remove
10.0.0.0/16	local	Active	No	
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="igw-2afc174e"/>		No	<input type="button" value="x"/>

Add another route

Save...

And then under Subnet Associations tab: Edit:

**rtb-00261364 | czarhipaa\_public\_routetable**

Summary Routes **Subnet Associations** Route Propagation Tags

[Cancel](#) [Save](#)

Associate	Subnet	CIDR	Current Route Table
<input checked="" type="checkbox"/>	<a href="#">subnet-2592fa41 (10.0.0.0/24)   public</a>	10.0.0.0/24	Main
<input type="checkbox"/>	<a href="#">subnet-4f93fb2b (10.0.1.0/24)   private</a>	10.0.1.0/24	Main

Now let's go back to the Route Table selector...

Subnets

Route Tables

<input checked="" type="checkbox"/>	<a href="#">czarhipaa_private_route</a>	<a href="#">rtb-e0261384</a>	0 Subnets	No	<a href="#">vpc-f98cf79d (10.0.0.0/16)   czarh...</a>
-------------------------------------	---	------------------------------	-----------	----	---

**rtb-e0261384 | czarhipaa\_private\_routetable**

Summary **Routes** Subnet Associations Route Propagation 1

[Cancel](#) [Save](#)

Destination	Target	Status	Propagated	Remove
10.0.0.0/16	local	Active	No	
<input type="text" value="0.0.0.0/0"/>	<input type="text"/>		No	

[Add another route](#)

[igw-2afc174e | czarhipaa\\_igw](#)  
[nat-06c541083712a04f7](#)

[Cancel](#) [Save](#)

Destination	Target	Status	Propagated	Remove
10.0.0.0/16	local	Active	No	
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="nat-06c541083712a04f7"/>		No	

[Add another route](#)

And now Subnet Associations tab:

Summary

Routes

Subnet Associations

Route Propagation

Tags

Edit

Subnet

CIDR

You do not have any subnet associations.

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Subnet

CIDR

[subnet-4f93fb2b \(10.0.1.0/24\) | private](#) 10.0.1.0/24

Cancel

Save

Associate

Subnet

CIDR

Current Route Table

[subnet-2592fa41 \(10.0.0.0/24\) | public](#)

10.0.0.0/24

[rtb-00261364 | czarhipaa\\_public\\_routetable](#)[subnet-4f93fb2b \(10.0.1.0/24\) | private](#)

10.0.1.0/24

Main

Now for the Endpoint

VPC Dashboard

Create Endpoint

Actions ▾

Filter by VPC:

None ▾



Filter by attributes or search by keyword

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

Notice that this has Full Access; we will restrict access at a later step:

## Step 1: Configure Endpoint

A VPC Endpoint allows you to securely connect your Amazon VPC to another AWS service.

**VPC\*** vpc-f98cf79d (10.0.0.0/16) | czarhipaa 

**Service** com.amazonaws.us-west-2.s3 

**Policy\*** ☒ Full Access - Allow access by any user or service within the VPC using credentials from any AWS accounts to any S3 resources. All policies — IAM user policies, VPC endpoint policies, and AWS service-specific policies (e.g. Amazon S3 bucket policies, any S3 ACL policies) — must grant the necessary permissions for access to succeed.  
☐ Custom

Use the [policy creation tool](#) to generate a policy, then paste the generated policy below.

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": "*"
    }
  ]
}
```



[Cancel and Exit](#)

[Next Step](#)

Now we end up here... but we