AWS CloudHSM

User Guide

AWS CloudHSM: User Guide

Table of Contents

What Is AWS CloudHSM?

A hardware security module (HSM) is a hardware appliance that provides secure key storage and cryptographic operations within a tamper-resistant hardware module. HSMs are designed to securely store cryptographic key material and use the key material without exposing it outside the cryptographic boundary of the appliance.

AWS CloudHSM helps you meet corporate, contractual and regulatory compliance requirements for data security by using dedicated HSM appliances within the AWS cloud. AWS and AWS Marketplace partners offer a variety of solutions for protecting sensitive data within the AWS platform, but additional protection is necessary for some applications and data that are subject to strict contractual or regulatory requirements for managing cryptographic keys.

Until now, your only options were to maintain the sensitive data or the encryption keys protecting the sensitive data in your on-premises data centers. However, those options either prevented you from migrating these applications to the cloud or significantly slowed application performance. AWS CloudHSM allows you to protect your encryption keys within HSMs that are designed and validated to government standards for secure key management. You can securely generate, store, and manage the cryptographic keys used for data encryption in a way that ensures that only you have access to the keys. AWS CloudHSM helps you comply with strict key management requirements within the AWS cloud without sacrificing application performance.

Payment Card Industry (PCI) Data Security Standard (DSS) Compliance

AWS CloudHSM supports the processing, storage, and transmission of credit card data by a merchant or service provider, and has been validated as being compliant with Payment Card Industry (PCI) Data Security Standard (DSS). For more information about PCI DSS, including how to request a copy of the AWS PCI Compliance Package, see PCI DSS Level 1.

Pricing

For more information about AWS CloudHSM pricing, go to AWS CloudHSM Pricing. If you want to try the AWS CloudHSM service for free, you can request a two week trial. For more information about the free trial, go to Free Trial.

Related Services

AWS CloudHSM works with Amazon Virtual Private Cloud (Amazon VPC). HSM appliances are provisioned inside your VPC with an IP address that you specify, providing simple and private network connectivity to your EC2 instances. Placing HSM appliances near your EC2 instances decreases network latency, which can improve application performance. Your HSM appliances are dedicated exclusively to you and are isolated from other AWS customers. Available in multiple regions and Availability Zones, AWS CloudHSM can be used to build highly available and durable applications.

For more information about Amazon VPC, see What Is VPC? in the *Amazon VPC User Guide*.

Where to Get Additional Help

We recommend that you take advantage of the AWS Discussion Forums. These are community-based forums for users to discuss technical questions related to AWS services. For the AWS CloudHSM forum, go to https://forums.aws.amazon.com/forum.jspa?forumID=156.

You can also get help if you subscribe to AWS Premium Support, a one-on-one, fast-response support channel (for more information, go to http://aws.amazon.com/premiumsupport).

About Amazon Web Services

Amazon Web Services (AWS) is a collection of digital infrastructure services that developers can leverage when developing their applications. The services include computing, storage, database, and application synchronization(messaging and queuing). AWS uses a pay-as-you-go service model. You are charged only for the services that you—or your applications—use. Also, to make AWS more approachable as a platform for prototyping and experimentation, AWS offers a free usage tier. On this tier, services are free below a certain level of usage. For more information about AWS costs and the Free Tier, see Test-Driving AWS in the Free Usage Tier. To obtain an AWS account, open the AWS home page and then click Sign Up.

Setting Up AWS CloudHSM

Before you can use AWS CloudHSM, you must have an AWS account, and have a specific environment in which your HSM appliances are provisioned.

Topics

Create an AWS Account

Your AWS account gives you access to all services, but you are charged only for the resources that you use.

If you do not have an AWS account, use the following procedure to create one.

To sign up for AWS

1. Open http://aws.amazon.com/ and choose **Create an AWS Account**.
2. Follow the online instructions.

Your root account credentials identify you to services in AWS and grant you unlimited use of your AWS resources. To allow other users to manage AWS CloudHSM resources without sharing your security credentials, use AWS Identity and Access Management (IAM). We recommend that everyone work as an IAM user, even the account owner. You should create an IAM user for yourself, give that IAM user administrative privileges, and use it for all your work. For more information, see Controlling Access to AWS CloudHSM Resources (p. 5).

Create an IAM User

The AWS CloudHSM API and CLI tools require your access keys so that the service can determine whether you have permission to access its resources. You can create access keys for your AWS account to access the API and CLI. However, we recommend that you avoid accessing AWS using your root AWS account access keys; instead, we recommend that you use AWS Identity and Access Management (IAM) to create an IAM user and add the IAM user to an IAM group with administrative permissions. This grants the IAM user administrative permissions. You then use the access keys for the IAM user with the AWS CloudHSM API and CLI.

If you signed up for AWS but have not created an IAM user for yourself, you can create one using the IAM console.

To create an IAM user for yourself and add the user to an Administrators group

1. Sign in to the Identity and Access Management (IAM) console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users**, and then choose **Add user**.
3. For **User name**, type a user name, such as `Administrator`. The name can consist of letters, digits, and the following characters: plus (+), equal (=), comma (,), period (.), at (@), underscore (_), and hyphen (-). The name is not case sensitive and can be a maximum of 64 characters in length.
4. Select the check box next to **AWS Management Console access**, select **Custom password**, and then type the new user's password in the text box. You can optionally select **Require password reset** to force the user to select a new password the next time the user signs in.
5. Choose **Next: Permissions**.
6. On the **Set permissions for user** page, choose **Add user to group**.
7. Choose **Create group**.
8. In the **Create group** dialog box, type the name for the new group. The name can consist of letters, digits, and the following characters: plus (+), equal (=), comma (,), period (.), at (@), underscore (_), and hyphen (-). The name is not case sensitive and can be a maximum of 128 characters in length.
9. For **Filter**, choose **Job function**.
10. In the policy list, select the check box for **AdministratorAccess**. Then choose **Create group**.
11. Back in the list of groups, select the check box for your new group. Choose **Refresh** if necessary to see the group in the list.
12. Choose **Next: Review** to see the list of group memberships to be added to the new user. When you are ready to proceed, choose Add permissions.

You can use this same process to create more groups and users, and to give your users access to your AWS account resources. To learn about using policies to restrict users' permissions to specific AWS resources, go to Access Management and Example Policies for Administering AWS Resources.

To sign in as this new IAM user, sign out of the AWS Management Console, then use the following URL, where *<your_aws_account_id>* is your AWS account number without the hyphens (for example, if your AWS account number is `1234-5678-9012`, your AWS account ID is `123456789012`):

```
https://<your_aws_account_id>.signin.aws.amazon.com/console/
```

Enter the IAM user name and password that you just created. When you're signed in, the navigation bar displays "*<your_user_name>* @ *<your_aws_account_id>*".

If you don't want the URL for your sign-in page to contain your AWS account ID, you can create an account alias. From the IAM dashboard, click **Customize** and enter an alias, such as your company name. To sign in after you create an account alias, use the following URL:

```
https://<your_account_alias>.signin.aws.amazon.com/console/
```

For more information about using IAM policies to control access to your AWS Directory Service resources, see Controlling Access to AWS CloudHSM Resources (p. 5).

Controlling Access to AWS CloudHSM Resources

By default, IAM users don't have permission to AWS CloudHSM operations. To allow IAM users to manage AWS CloudHSM operations, you must create an IAM policy that explicitly grants IAM users permission to use certain AWS CloudHSM operations, and attach the policy to the IAM users or groups that require those permissions. For more information about IAM policies, see Permissions and Policies in the *IAM User Guide* guide.

The following policy statement grants a user or group permission to use all AWS CloudHSM operations.

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : "cloudhsm:*",
      "Resource" : "*"
    }
  ]
}
```

The following policy statement grants a user or group permissions to use the operations that read AWS CloudHSM resources.

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "cloudhsm:Get*",
        "cloudhsm:List*",
        "cloudhsm:Describe*"
      ],
      "Resource" : "*"
    }
  ]
}
```

For more information about IAM, see the following:

- Identity and Access Management (IAM)
- IAM User Guide

Set up the AWS CloudHSM Environment

AWS CloudHSM requires the following environment before an HSM appliance can be provisioned.

- A virtual private cloud (VPC) in the region where you want the AWS CloudHSM service. For more information about Amazon VPC, see What Is VPC? in the *Amazon VPC User Guide*.
- One private subnet (a subnet with no Internet gateway) in the VPC. The HSM appliance is provisioned into this subnet.
- One public subnet (a subnet with an Internet gateway attached). The control instances are attached to this subnet.
- An AWS Identity and Access Management (IAM) role that delegates access to your AWS resources to AWS CloudHSM. This is needed so AWS CloudHSM can create and configure AWS resources, such as elastic network interfaces, on your behalf. For more information about IAM roles, see Roles in the *IAM User Guide* guide.
- An EC2 instance, in the same VPC as the HSM appliance, that has the SafeNet client software installed. This instance is referred to as the control instance and is used to connect to and manage the HSM appliance.
- A security group that has port 22 (for SSH) or port 3389 (for RDP) open to your network. This security group is attached to your control instances so you can access them remotely. For more information, see Authorizing Inbound Traffic for Your Instances in the *Amazon EC2 User Guide for Linux Instances*.

You can either use AWS CloudFormation to set up your AWS CloudHSM environment, or set up your environment manually.

- Automatically Setting Up Your AWS CloudHSM Environment Using AWS CloudFormation (p. 6)
- Manually Setting Up Your AWS CloudHSM Environment (p. 9)

Automatically Setting Up Your AWS CloudHSM Environment Using AWS CloudFormation

You can use an AWS CloudFormation template from AWS CloudHSM to automatically set up your AWS environment for AWS CloudHSM.

Topics

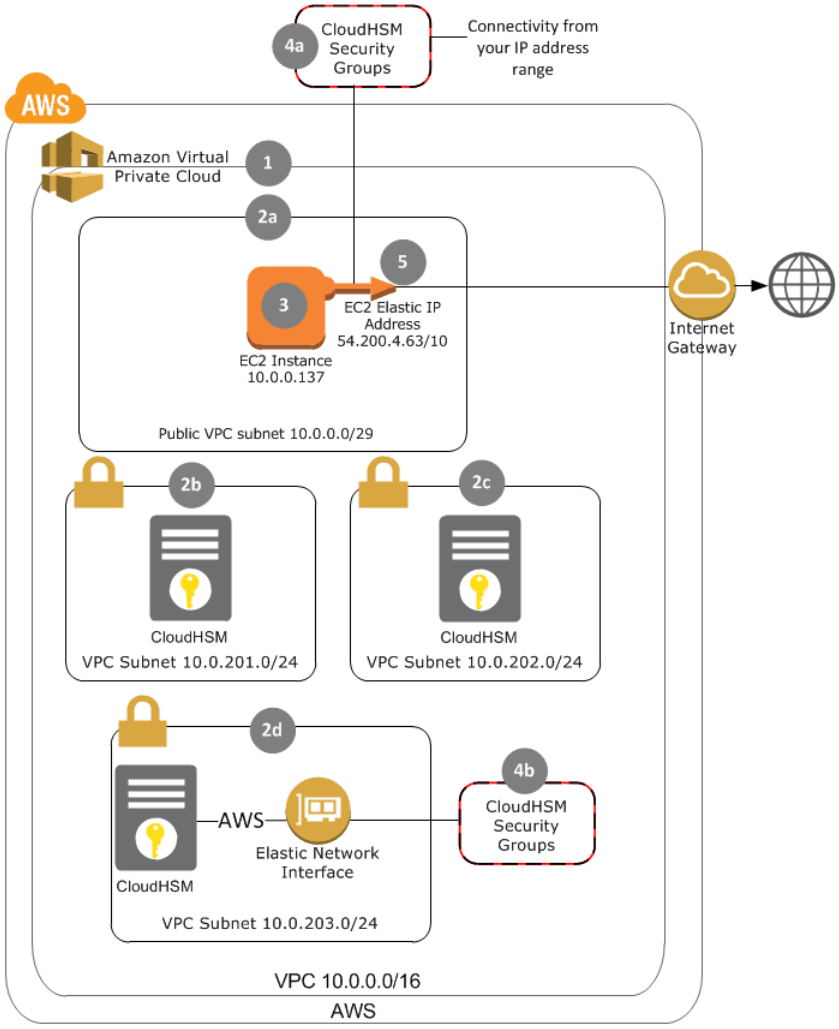
Prerequisites

Before you can start this process, you need the following:

- Your AWS account must have one VPC available to be created in the selected region. For the number of VPCs allowed per AWS region, see Amazon VPC Limits in the *Amazon VPC User Guide*.
- An Amazon EC2 key pair. You will use this key pair to access the client instance that AWS CloudFormation creates. You must create this key pair in the same AWS region where you will set up your AWS CloudHSM environment. For more information, see Creating Your Key Pair Using Amazon EC2 in the *Amazon EC2 User Guide for Linux Instances*.

AWS CloudHSM Environment Details

The following diagram shows the AWS environment that the AWS CloudFormation template automatically sets up for you to use with AWS CloudHSM.



AWS CloudFormation creates and configures the following resources for you:

1. A virtual private cloud (VPC).
2. Subnets, including one subnet that is publicly accessible and a private subnet for each Availability Zone. Consider these examples:
   - For regions that have three Availability Zones, four subnets are created: one subnet that is publicly accessible (2a) and three private subnets (2b, 2c, and 2d).
   - For regions that have two Availability Zones, three subnets are created: one subnet that is publicly accessible (2a) and two private subnets (2c and 2d).

     Note
     AWS CloudHSM provisions each HSM appliance into a private subnet to isolate it from the Internet.
3. An Amazon Elastic Compute Cloud (Amazon EC2) instance (m3.medium running Amazon Linux x86 64-bit) in the public subnet, with the SafeNet client software already installed. This instance is referred to as the *client instance*. To authenticate your identity with the client instance, you use the key pair that you specify during the creation of the AWS CloudFormation stack.

4. Security groups that allow SSH connections into the public subnet from the Internet (4a) as well as SSH and NTLS connections into the private subnet from the public subnet (4b).
5. An Elastic IP address for the client instance.
6. An IAM role that allows AWS CloudHSM to access your AWS resources. (Not shown in the preceding diagram.)
7. The necessary IAM credentials to send an Amazon Simple Notification Service (Amazon SNS) notification of your stack's configuration to AWS CloudHSM. (Not shown in the diagram.)

Setting Up Your AWS CloudHSM Environment Using AWS CloudFormation

Complete the following steps to have AWS CloudFormation set up your AWS CloudHSM environment from the `cloudhsm-quickstart` template.

To use AWS CloudFormation to set up your AWS CloudHSM environment automatically

1. Sign in to the AWS Management Console and open the AWS CloudFormation console at https://console.aws.amazon.com/cloudformation/.
2. On the navigation bar, use the region selector to choose one of the AWS regions where AWS CloudHSM is currently supported:
   - **US East (N. Virginia)**
   - **US East (Ohio)**
   - **US West (N. California)**
   - **US West (Oregon)**
   - **Canada (Central)**
   - **EU (Ireland)**
   - **EU (Frankfurt)**
   - **Asia Pacific (Tokyo)**
   - **Asia Pacific (Singapore)**
   - **Asia Pacific (Sydney)**
3. Choose **Create Stack** or **Create New Stack**.
4. Choose **Specify an Amazon S3 template URL**, then type the following URL: `https://cloudhsm.s3.amazonaws.com/cloudhsm-quickstart.json`

   Choose **Next**.
5. For **Stack name**, type an identifiable name for the stack such as **`CloudHSM-Environment`**. For **KeyName**, choose a key pair to use when connecting to your HSM client instance. Choose **Next**.
6. (Optional) On the **Options** page, add any tags you want to apply to the stack. When you are finished, choose **Next**.
7. On the **Review** page, review your settings and then select the **I acknowledge that this template might cause AWS CloudFormation to create IAM resources.** check box. This acknowledges that you understand that AWS CloudFormation will create an IAM role in your account and will use the IAM role to create the other AWS resources described in the preceding section (AWS CloudHSM Environment Details (p. 7)).

   Choose **Create**.

After the stack is created, the status changes to **CREATE_COMPLETE**. If an error occurs, the stack is rolled back and the status eventually changes to **ROLLBACK_COMPLETE**. You can use the **Events** tab in the AWS CloudFormation console to help determine why the failure occurred.

For more information about AWS CloudFormation stacks, see Viewing AWS CloudFormation Stack Data and Resources on the AWS Management Console in the *AWS CloudFormation User Guide*.

Preparing To Provision Your HSMs

Collect the following information. This information is required to provision your HSMs. This information is available in the **Outputs** tab of the AWS CloudFormation console when your AWS CloudFormation stack is complete.

- The IAM role ARN
- The private subnet IDs
- The client IP address

After collecting this information, proceed to Generating an SSH Key (p. 13).

Manually Setting Up Your AWS CloudHSM Environment

Use the following procedures to manually set up your AWS environment for use with AWS CloudHSM. If you prefer, you can instead use an AWS CloudFormation template provided by AWS CloudHSM to set up your environment automatically. For more information, see Automatically Setting Up Your AWS CloudHSM Environment Using AWS CloudFormation (p. 6).

To set up your environment manually, complete the steps in each of the following topics.

Topics

Create a Virtual Private Cloud (VPC)

Use Amazon Virtual Private Cloud (Amazon VPC) to create a new VPC.

To create a VPC

1. Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. On the navigation bar, use the region selector to choose one of the AWS regions where AWS CloudHSM is currently supported:

   - **US East (N. Virginia)**
   - **US East (Ohio)**
   - **US West (N. California)**
   - **US West (Oregon)**
   - **Canada (Central)**
   - **EU (Ireland)**
   - **EU (Frankfurt)**

- **Asia Pacific (Tokyo)**
- **Asia Pacific (Singapore)**
- **Asia Pacific (Sydney)**

3. Choose **Start VPC Wizard**.
4. Choose the first option, **VPC with a Single Public Subnet**, and then choose **Select**.
5. For **VPC name:**, type an identifiable name such as **`CloudHSM`**. For **Subnet name:**, type an identifiable name such as **`CloudHSM public subnet`**. Leave all other options set to their defaults, and choose **Create VPC**.

Create the Private Subnets

Create a private subnet (a subnet with no Internet gateway attached) for each Availability Zone in the region. This provides the most flexibility in choosing the subnet for your HSMs. Provisioning HSMs in different Availability Zones provides the most robust configuration for high availability.

To create the private subnets in your HSM VPC

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. On the navigation pane, choose **Subnets** and then choose **Create Subnet**.
3. On the **Create Subnet** dialog box, do the following:

   a. For **Name tag**, type an identifiable name such as **`CloudHSM private subnet`**.
   b. For **VPC**, choose the CloudHSM VPC that you created previously.
   c. For **Availability Zone**, choose the first Availability Zone in the list.
   d. For **CIDR block**, type the CIDR block to use for the subnet.

      For more information about choosing a subnet CIDR block, see Subnet Sizing in the *Amazon VPC User Guide*.

   Choose **Yes, Create**.
4. Repeat steps 3 and 4 for each remaining Availability Zone in the region.

Create the Security Group

Create a security group for use with AWS CloudHSM, and then add the required inbound rules to your security group.

> **Note**
> The security group rules provided here are the minimum rules that you need to get started with AWS CloudHSM. For production deployments, you should define appropriate rules to constrain network traffic according to your security policies and best practices.

To create your security group for use with AWS CloudHSM

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. On the navigation pane, choose **Security Groups** and then choose **Create Security Group**.
3. In the **Create Security Group** dialog box, do the following:

   a. For **Name tag**, type an identifiable name such as **`CloudHSM_SG`**.
   b. For **Group name**, type an identifiable name. You can type the same name you used for **Name tag**.
   c. For **Description**, type a description such as **`SG for CloudHSM instances`**.

   d. For **VPC**, choose the CloudHSM VPC that you created previously.
4. Choose **Yes, Create**.

To add the required inbound rules to your security group

1. With your security group selected in the list, choose the **Inbound Rules** tab and then choose **Edit**.
2. Do the following to add an inbound rule that allows traffic on port 22 (SSH) from your network:
   a. For **Type**, choose **SSH (22)**.
   b. For **Source**, type the CIDR block for your network.
   c. Choose **Add another rule**.
3. Do the following to add an inbound rule that allows traffic on port 22 (SSH) from your VPC:
   a. For **Type**, choose **SSH (22)**.
   b. For **Source**, type or choose the security group ID of the CloudHSM security group that you created previously. For example, **`sg-0123abcd`**.
   c. Choose **Add another rule**.
4. Do the following to add an inbound rule that allows traffic on port 3389 (RDP) from your network:
   a. For **Type**, choose **RDP (3389)**.
   b. For **Source**, type the CIDR block for your network.
   c. Choose **Add another rule**.
5. Do the following to add an inbound rule that allows traffic on port 1792 from your VPC:
   a. For **Type**, choose **Custom TCP Rule**.
   b. For **Port Range**, type **`1792`**.
   c. For **Source**, type or choose the security group ID of the CloudHSM security group that you created previously. For example, **`sg-0123abcd`**.
6. Choose **Save** to add all four of the inbound rules to your security group.

Create an IAM Role

You must give AWS CloudHSM permission to perform certain actions on your behalf, such as listing your VPCs and creating elastic network interfaces (ENIs) to attach to your HSMs. To do this, you create an IAM role that AWS CloudHSM is allowed to assume and that gives these permissions.

To create the IAM role for AWS CloudHSM

1. Open the Identity and Access Management (IAM) console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Roles** and then choose **Create New Role**.
3. For **Role Name**, type an identifiable name such as **`CloudHSM_Role`** and then choose **Next Step**.
4. Select **AWS Service Roles**. Scroll down to find **AWS CloudHSM**. On the **AWS CloudHSM** row, choose **Select**.
5. Select the check box next to **AWSCloudHSMRole** and then choose **Next Step**.
6. Choose **Create Role**.

Launch a Client Instance

Create a *client instance* from which to access the HSM. To access the HSM from your client instance, you must install the HSM client software. AWS CloudHSM offers a custom Amazon Machine Image

(AMI) that you can use to launch an Amazon Elastic Compute Cloud (Amazon EC2) instance that is preconfigured with the HSM client software. Alternatively, you can manually install the HSM client software on an instance you already have. For more information about manually installing the HSM client software, see Configuring a Linux HSM Client (p. 27) or Configuring a Windows HSM Client (p. 29).

To launch a preconfigured EC2 instance from the AWS CloudHSM AMI

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. Choose **Launch Instance**.
3. Choose the **Community AMIs** tab, and then use **Search community AMIs** to find **`CloudHSM 5.4`**.
4. Locate the row for the **CloudHSM 5.4 Client AMI**. Use the following table of AMI IDs to ensure that you choose the correct AMI for your AWS region. Then choose **Select**.

AWS Region	AMI ID
US East (N. Virginia)	ami-85a975ee
US East (Ohio)	ami-58f6ad3d
US West (N. California)	ami-e5561a85
US West (Oregon)	ami-cd717dfd
Canada (Central)	ami-309d2f54
EU (Ireland)	ami-1e501b69
EU (Frankfurt)	ami-c22326df
Asia Pacific (Tokyo)	ami-8c09be8c
Asia Pacific (Singapore)	ami-00cecc52
Asia Pacific (Sydney)	ami-87d492bd

5. Select the instance type that you want to launch, and then choose **Next: Configure Instance Details**.
6. For **Step 3: Configure Instance Details**, do the following:
   a. For **Network**, choose the CloudHSM VPC that you created previously.
   b. For **Subnet**, choose the CloudHSM public subnet you created previously.
   c. For **Auto-assign Public IP**, choose **Enable**.
   d. (Optional) Change the remaining instance details as desired.
7. Choose **Next: Add Storage**, and then optionally change the storage settings as desired.
8. Choose **Next: Tag Instance**, and then optionally add tags as desired.
9. Choose **Next: Configure Security Group**.
10. Choose **Select an existing security group** and then select the check box next to the CloudHSM security group that you created previously. Choose **Review and Launch**.
11. Review your instance details, and then choose **Launch**.
12. Choose whether to launch your instance with an existing key pair or to create a new key pair in the **Select an existing key pair or create a new key pair** dialog box.
    - To use an existing key pair, do the following:
      1. Choose **Choose an existing key pair**.

2. For **Select a key pair**, choose the key pair to use.
3. Select the check box next to **I acknowledge that I have access to the selected private key file (*`private key file name`*.pem), and that without this file, I won't be able to log into my instance.**

- To create a new key pair, do the following:
  1. Choose **Create a new key pair**.
  2. For **Key pair name**, type an identifiable key pair name such as **`CloudHSM client key`**.
  3. Choose **Download Key Pair** and save the private key file in a secure and accessible location.

     Warning
     You will not be able to download the private key file again after this point. If you do not download the private key file now, you will be unable to access the client instance.

13. Choose **Launch Instances**.

When the control instance is running, you can connect to it using SSH. For more information, see Connect to Your Instance in the *Amazon EC2 User Guide for Linux Instances*.

Preparing to Provision Your HSMs

Collect the following information. This information is required when you provision your HSMs.

- The subnet IDs of the private subnets that you created previously for CloudHSM. This information is available in the Amazon VPC console at https://console.aws.amazon.com/vpc/home#subnets:.
- The group ID of the security group that you created previously for CloudHSM. This information is available in the Amazon VPC console at https://console.aws.amazon.com/vpc/home#securityGroups:.
- The Amazon Resource Name (ARN) of the IAM role that you created previously for CloudHSM. This information is available in the IAM console at https://console.aws.amazon.com/iam/home#roles. Choose the name of the role for CloudHSM, and then note the **Role ARN**.

Proceed to Generating an SSH Key (p. 13).

Generating an SSH Key

AWS CloudHSM uses an SSH key pair to authenticate the manager account when logging in to the HSM appliance. When you sign up for AWS CloudHSM, you supply the public key to AWS. It is important that you only send the public key information to AWS. The public key is installed on the HSM appliance during provisioning. The private key must be available to any instance you use to connect to the HSM appliance.

You can generate the key pair on any machine, but you need to copy the private key to any instances that will be used to connect to the HSM appliance. If you generate the key pair on the same instance that you will use to connect to the HSM appliance, you don't have to copy the private key file. You can use an existing SSH key pair or generate a new one. There are many key pair generators available, but on Linux, a common generator is the **ssh-keygen** command. On Windows, you can use the PuTTYgen utility.

You should include a passphrase with the private key to prevent unauthorized persons from logging in to your HSM appliance. When you include a passphrase, you have to enter the passphrase whenever you log in to the HSM appliance.

Topics

Generating an SSH Key on Linux

To generate an SSH key on a Linux machine, you can use the **ssh-keygen** command as shown in the following example:

```
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
df:c4:49:e9:fe:8e:7b:eb:28:d5:1f:72:82:fb:f2:69
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|              .  |
|            o    |
|           + .   |
|        S   *.   |
|         . =.o.o |
|          ..+ +..|
|          .o Eo .|
|           .OO=. |
+-----------------+
$
```

Generating an SSH Key on Windows

To generate an SSH key on a Windows machine, you can use the PuTTYgen utility. For more information about using the PuTTYgen utility to create a key pair, go to http://www.howtoforge.com/ssh_key_based_logins_putty.

PuTTYgen stores its private keys in a proprietary format that is only used by PuTTY. If you need to use the private key with an SSH client other than PuTTY, you can use PuTTYgen to convert the private key to OpenSSH format by clicking on **Conversion** in the PuTTYgen menu and selecting **Export OpenSSH key**.

The public key that is used by the HSM appliance must be in SSH format. In PuTTYgen, copy the contents of the **Public key for pasting into OpenSSH authorized keys file** field and save this to a file. This is your public key file.

Example SSH Public Key

The following example shows an SSH public key that was generated using the **ssh-keygen** command on Linux. The public key that you provide to AWS should look similar to the following. The line breaks in the following example are only for readability; your SSH public key should be one continuous line.

```
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEA6bUsFjDSFcPC/BZbIAv8cAR5syJMB
```

```
GiEqzFOIEHbm0fPkkQ0U6KppzuXvVlc2u7w0mgPMhnkEfV6j0YBITu0Rs8rNHZFJs
CYXpdoPxMMgmCf/FaOiKrb7+1xk21q2VwZyj13GPUsCxQhRW7dNidaaYTf14sbd9A
qMUH4UOUjs27MhO37q8/WjV3wVWpFqexm3f4HPyMLAAEeExT7UziHyoMLJBHDKMN7
1Ok2kV24wwn+t9P/Va/6OR6LyCmyCrFyiNbbCDtQ9JvCj5RVBla5q4uEkFRl0t6m9
XZg+qT67sDDoystq3XEfNUmDYDL4kq1xPM66KFk3OS5qeIN2kcSnQ==
```

Copying the Private Key

You must copy the private key to all instances that will be used to connect to the HSM appliance. These instances are referred to as `control instances`.

Topics

- Copy the Private Key to a Linux Instance (p. 15)
- Copy the Private Key to a Windows Instance (p. 15)

Copy the Private Key to a Linux Instance

Perform the following steps if your control instance is a Linux instance.

1. If the key was created using PuTTYgen, use PuTTYgen to convert the private key to OpenSSH format. For more information, see Generating an SSH Key on Windows (p. 14).
2. Copy the private key file from the machine it is stored on to the `~/.ssh/` directory on the control instance.
3. Connect to the control instance over SSH. The remaining steps in this procedure are performed from the control instance.
4. In the control instance, modify the permissions for the private key file.

   ```
   $ chmod 600 ~/.ssh/[private_key_file]
   ```

5. Use **ssh-add** to add the private key to the authentication agent. The **ssh-add** command prompts you for the passphrase that was used to secure the private key when it was generated.

   ```
   $ ssh-add ~/.ssh/[private_key_file]
   ```

   When you connect to the HSM appliance, this key is now used for authentication. You have to repeat this command every time you reconnect to the control instance. As an alternative, you can specify which private key file **ssh** and **scp** should use with the **-i** option.

Copy the Private Key to a Windows Instance

Perform the following steps if your control instance is a Windows instance.

1. Copy the private key file from the machine it was stored on to the directory on the control instance where your PuTTY keys are stored.
2. Connect to the control instance over RDP. The remaining steps in this procedure are performed from the control instance.
3. If the private key is not a PuTTY private key file, perform the following steps:

   a. In the control instance, use PuTTYgen to import the private key file that was copied by clicking on **Conversion** in the PuTTYgen menu, selecting **Import key**, and selecting the private key file. You are prompted for the passphrase for the key.

   b. In PuTTYgen, save the private key as a PuTTY private key file by selecting **Save private key**.

When you connect to the HSM appliance using PuTTY, you use this private key file for authentication. To prevent you from having to enter your passphrase every time you log in, you can use Pageant. Pageant is an SSH authentication agent that is used with PuTTY. It holds your private keys in memory, already decoded, so that you can use them often without needing to type a passphrase. For more information, go to Using Pageant for authentication.

Setting Up the AWS CloudHSM CLI Tools

The AWS CloudHSM CLI tools simplify and centralize your HSM administration. The tools make it easy for you to create and initialize your HSM appliances and configure your HSMs into a high availability configuration without having to log into each HSM and run Luna shell commands.

Topics

Installing the CLI Tools

To set up and configure an instance for use with the AWS CloudHSM CLI tools, perform the following steps.

To set up an instance for the AWS CloudHSM CLI

1. Launch a Linux instance in the same VPC that contains your HSM appliances.
2. Configure your VPC security groups as follows:
   - The security group that is assigned to the instance must have port 22 (SSH) open to incoming traffic from your network. This allows you to connect to the instance using SSH.
   - The security group that is assigned to the HSM appliance must have port 22 (SSH) open to incoming traffic from your VPC. This allows the instance to communicate with the HSM appliance.
3. Install Python 2.7 and pip on the Linux instance as described in Installing Python 2.7 (p. 16) and Installing pip (p. 17).
4. Download and install the AWS CloudHSM CLI tools as described in Installing the AWS CloudHSM CLI Tools (p. 17).
5. Copy the private key files for all of your HSMs to the instance. These are the private portions of the keys that were installed on your HSMs when they were provisioned. These are needed for many commands, such as `initialize-hsm (p. 82)` and `add-hsm-to-hapg (p. 57)`. For more information, see Copying the Private Key (p. 15).

Topics

Installing Python 2.7

You can determine if Python 2.7 is already installed by running the following command on the instance:

```
$ python2.7 -V
```

If the response is that the `python2.7` command is not found, install Python 2.7 by running one of the following commands on the instance.

- On RHEL systems and RHEL derivatives, including Amazon Linux, use the following command:

```
$ sudo yum install python27
```

- On Debian systems and Debian derivatives, such as Ubuntu, use the following command:

```
$ sudo apt-get install python27
```

Installing pip

You can determine if pip is already installed by running the following command on the instance:

```
$ pip -V
```

If the response is that the `pip` command is not found, download pip by running the following command on the instance:

```
$ curl -O https://bootstrap.pypa.io/get-pip.py
```

Then install pip by running the following command on the instance:

```
$ sudo python2.7 get-pip.py
```

Installing the AWS CloudHSM CLI Tools

After installing Python 2.7 (p. 16) and installing pip (p. 17), install the AWS CloudHSM CLI Tools by running one of the following commands on the instance.

- On Amazon Linux, use the following command:

```
$ sudo yum install aws-cloudhsm-cli
```

- On all other operating systems, use the following command:

```
$ pip install aws-cloudhsm-cli
```

  You might need to run the preceding `pip` command with `sudo`, depending on your operating system configuration.

Use the following command to verify that you have the AWS CloudHSM CLI tools correctly installed.

```
$ cloudhsm version
```

Setting the Necessary File and Directory Ownership

If your instance has the SafeNet client software installed, the AWS CloudHSM CLI tools require that the user running the commands be the owner of certain files and directories.

To set the owner of the files and directories

1. Use the following commands on the instance that is running the AWS CloudHSM CLI tools to set the owner and write permission on the `Chrystoki.conf` file.

   ```
   $ sudo chown <owner> /etc/Chrystoki.conf

   $ sudo chmod +w /etc/Chrystoki.conf
   ```

   The *<owner>* can be either the user or a group that the user belongs to.
2. Use the following command on the instance that is running the AWS CloudHSM CLI tools to set the owner of the Luna client directory.

   ```
   $ sudo chown <owner> -R /usr/safenet/lunaclient/
   ```

   <owner> can be either the user or a group that the user belongs to.

Configuring the AWS CloudHSM CLI Tools

The following topics explain how to configure and use the AWS CloudHSM CLI tools.

Topics

Authentication

The AWS CloudHSM CLI tools use your AWS access key ID and secret access key credentials to identify and authenticate you with the service. For more information about these keys, see AWS Security Credentials and Best Practices for Managing AWS Access Keys in the *Amazon Web Services General Reference*.

You can provide your credentials to the AWS CloudHSM CLI tools in the following ways:

- You can set your credentials in the `aws_access_key_id` and `aws_secret_access_key` settings in a configuration file that you specify with the `--conf_file` parameter to each command. For more information, see Configuration Files (p. 19). This is the recommended method of providing your credentials to the CLI.
- You can set your credentials to be used by all AWS CloudHSM commands by adding them to a boto config file on the instance as shown in the following example.

  ```
  [Credentials]
  aws_access_key_id = access_key_id
  ```

```
aws_secret_access_key = secret_access_key
```

You can set this boto config file for all users on the system or just for the current user. To have these credentials apply to all users, save the file as `/etc/boto.cfg`. To have these credential apply to only the current user, save the file as `~/.boto`.

- You can provide your credentials as arguments to each command with the `--aws-access-key-id` and `--aws-secret-access-key` parameters to each command. All AWS CloudHSM CLI commands accept these arguments. We do not recommend this method because it can lead to inadvertent exposure of your credentials if a script is shared with others.

SSH Connections

Many of the AWS CloudHSM CLI commands, such as `initialize-hsm (p. 82)`, must communicate with your HSM appliances using the SSH protocol. To facilitate this, you must enable the command `ssh <hsm_ip_address>`, with no other parameters, to connect to each of your HSMs. There are several methods for accomplishing this. We recommend adding an entry similar to the following in your `~/.ssh/config` file.

```
Host <hsm_ip_address>
User manager
IdentityFile <private_key_file>
```

Replace `<hsm_ip_address>` with the IP address of your HSM appliance, and replace `<private_key_file>` with the private key file that corresponds to the public key that was installed on the HSM appliance during provisioning. If your private key is protected with a passphrase, you can use ssh-agent to unlock the private key and pass it to the ssh process so that the command `ssh <hsm_ip_address>`, with no other parameters or inputs, will connect to your HSM.

Note
Password-based SSH authentication is not supported. You must use a key pair to authenticate to your HSM. For more information, see Generating an SSH Key (p. 13).

Passwords

We recommend that you print out a copy of the Password Worksheet (p. 33), use it to record your AWS CloudHSM passwords, and store it in a secure place. We also recommended that you store at least one copy of this worksheet in secure off-site storage. AWS does not have the ability to recover your key material from an HSM for which you do not have the proper HSM security officer credentials.

Configuration Files

Many of the AWS CloudHSM CLI commands require common parameters, such as the AWS region or authentication credentials. Rather than pass these as command line options, you can instead set them in a configuration file and pass the configuration file with the `--conf_file` parameter.

The following example shows the format of the configuration file:

```
[cloudhsmcli]
aws_access_key_id=<value>
aws_secret_access_key=<value>
aws_region=<value>
hapg_arns=
  <value1>
  <value2>
  <value...>
```

```
so_password=<value>
```

Client Certificates

Every CloudHSM client requires a private key and a certificate to authenticate it with the HSM partition or partition group that it is associated with. You create an HSM client with the `create-client (p. 64)` command, passing the certificate to the command.

The certificate is a file that contains a base64-encoded X.509 v3 PEM certificate. The PEM certificate must be in the following format:

```
-----BEGIN CERTIFICATE-----
<certificate contents>
-----END CERTIFICATE-----
```

The private key must reside in the LunaSA client certificate directory on the client. This directory is created when the LunaSA client software is installed on the client. For more information about the LunaSA client software, see Configuring a Linux HSM Client (p. 27) or Configuring a Windows HSM Client (p. 29). The location of the LunaSA client certificate directory varies depending on your client operating system.

Linux clients
`/usr/safenet/lunaclient/cert`

Windows clients
`%ProgramFiles%\SafeNet\LunaClient\cert`

There are many ways to create this certificate. Two of the most common are to use the LunaSA `vtl createCert` command, or to use the OpenSSL toolkit.

Topics

- LunaSA Command (p. 20)
- OpenSSL Toolkit (p. 21)

LunaSA Command

To create a private key and certificate with the LunaSA `vtl` command, you must have the LunaSA client software installed on your client. For more information, see Configuring a Linux HSM Client or Configuring a Windows HSM Client in the *AWS CloudHSM User Guide*.

To create a client certificate with the `vtl` command on a Linux client, issue the following command.

```
$ sudo vtl createCert -n <client_name>
```

To create a client certificate with the `vtl` command on a Windows client, issue the following command.

```
C:\> vtl createCert -n <client_name>
```

The *<client_name>* can be any name that is unique and does not contain any spaces or special characters. You must use this same name for the `--label` parameter in the `create-client (p. 64)` command.

The output of the `vtl createCert` command will be similar to the following.

```
Private Key created and written to:
```

```
<luna_client_cert_dir>/<client_name>Key.pem
Certificate created and written to:
<luna_client_cert_dir>/<client_name>.pem
```

<luna_client_cert_dir> is the LunaSA client certificate directory on the client.

You pass the *<client_name>*`.pem` file for the `--certificate-filename` parameter in the `create-client` (p. 64) command.

OpenSSL Toolkit

You can use the OpenSSL toolkit to create your private key and certificate by issuing the following commands.

Create the private key.

```
openssl genrsa -out <luna_client_cert_dir>/<client_name>Key.pem 2048
```

Generate the certificate from the private key.

```
openssl req -new -x509 -days 3650 -
key <luna_client_cert_dir>/<client_name>Key.pem -out <client_name>.pem
```

<luna_client_cert_dir> is the LunaSA client certificate directory on the client.

The *<client_name>* can be any name that is unique and does not contain any spaces or special characters.

The output of the `openssl req` command will be similar to the following. You are prompted for several fields for use in the certificate. The only required field is `Common Name`, which must be the same as *<client_name>*. You must also use this same name for the `--label` parameter in the `create-client` (p. 64) command. The remaining fields can be left blank.

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:.
State or Province Name (full name) []:.
Locality Name (eg, city) [Default City]:.
Organization Name (eg, company) [Default Company Ltd]:.
Organizational Unit Name (eg, section) []:.
Common Name (eg, your name or your server's hostname) []:<client_name>
Email Address []:.
```

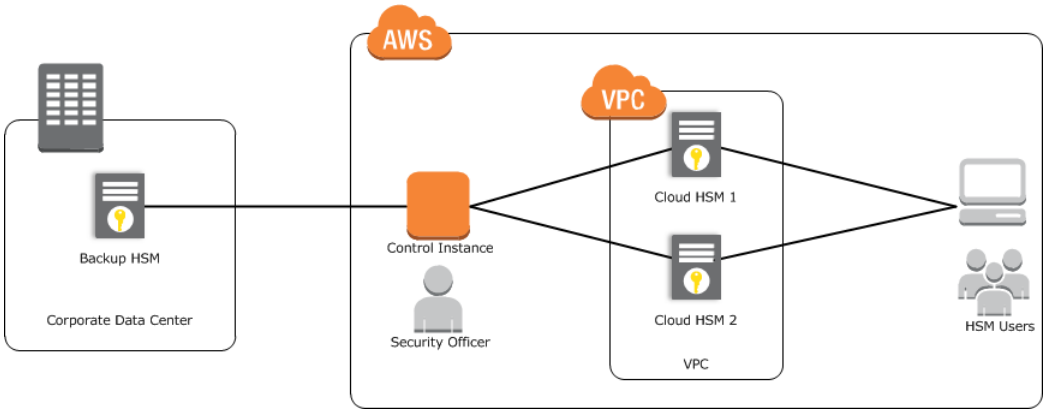
You pass the *<client_name>*`.pem` file for the `--certificate-filename` parameter in the `create-client` (p. 64) command.

Getting Started with AWS CloudHSM

AWS CloudHSM provides secure cryptographic key storage to customers by making hardware security modules (HSMs) available in the AWS cloud.

This guide gives you a hands-on introduction to using AWS CloudHSM, by walking you through the steps needed to set up and configure your HSM appliance, integrate third-party software applications with AWS CloudHSM, and write a simple application that uses the HSM appliance. This guide also describes best practices for using the AWS CloudHSM service.

The recommended configuration for using AWS CloudHSM is to use two HSMs configured in a high-availability configuration. For information about high-availability configuration, see High Availability and Load Balancing (p. 35).



This list summarizes the procedures needed to get up and running with AWS CloudHSM. Step-by-step instructions are detailed in the sections below.

To get started with AWS CloudHSM

1. If you have not already done so, follow the steps in Setting Up AWS CloudHSM (p. 3) to set up your HSM environment.

2. Provision one or more HSMs using the procedures in Provisioning Your HSMs (p. 23).
3. Initialize your HSMs using the procedures in Configuring Your HSM (p. 24).
4. Connect your on-premises HSM appliances to your HSM VPC as shown in Connect Your On-Premises HSM (p. 27).
5. Configure your HSM client (p. 27).
6. Configure HA (p. 35).
7. Select from the following two options:

   - Integrate AWS CloudHSM with third-party software applications. For more information, see Integrating Third-Party Applications with AWS CloudHSM (p. 47).
   - Sample Application (p. 107) to prepare to Building Your Own Applications (p. 49).

Important
This guide provides an abbreviated set of instructions that allow you to get started quickly with your AWS CloudHSM service. To secure production deployments, be sure to read the detailed descriptions and background information provided in the SafeNet Luna SA documentation in order to get a deeper understanding of the operation of the HSM. This guide does not attempt to provide those important details, which are essential for secure operation of the HSM.

Topics

Provisioning Your HSMs

You need the following information to provision your HSM.

- The identifier of the private subnet in which to provision the HSM. For more information, see Set up the AWS CloudHSM Environment (p. 6).
- The Amazon Resource Name (ARN) of the IAM role for AWS CloudHSM. For more information, see Set up the AWS CloudHSM Environment (p. 6).
- Your SSH public key. For more information, see Generating an SSH Key (p. 13).

Important
You are charged an upfront fee for each HSM you provision. If you accidentally provision an HSM and want to request a refund, delete the HSM (p. 50) and then go to the AWS Support Center to create a new case regarding **Account and Billing Support**.
To try the AWS CloudHSM service for free, you can request a two week trial. For more information about the free trial, go to Free Trial.

To provision your HSM, from your control instance, use the AWS CloudHSM CLI command `create-hsm (p. 67)`, as in the following example.

Note
The following example contains line breaks for readability. Do not include line breaks or backslash characters (\) when you use this command from your control instance.
Before you use the following command, ensure that you have set `aws_access_key_id`, `aws_secret_access_key`, and `aws_region` in a configuration file at `~/cloudhsm.conf`. For more information, see Configuration Files (p. 19).

```
$ cloudhsm create-hsm --conf_file ~/cloudhsm.conf \
```

```
--subnet-id <subnet_id> \
--ssh-public-key-file <public_key_file> \
--iam-role-arn <iam_role_arn> \
--syslog-ip <syslog_ip_address>
```

The following list describes each parameter used in the preceding example.

<subnet_id>
: The identifier of the subnet in your VPC in which to place the HSM.

<public_key_file>
: The file that contains the SSH public key to install on the HSM.

<iam_role_arn>
: The ARN of an IAM role that allows the AWS CloudHSM service to allocate an elastic network interface (ENI) on your behalf.

<syslog_ip_address>
: (Optional) The IP address of your syslog monitoring server. The AWS CloudHSM service supports the use of only one syslog monitoring server.

The response is similar to the following.

```
{
  "HsmArn": "<hsm_arn>",
  "RequestId": "<request_id>"
}
```

Make a note of the *<hsm_arn>* value because you need it to initialize the HSM.

Repeat this command to create as many HSMs as you need.

Configuring Your HSM

When you set up and configure your HSM, we recommend that you print out a copy of the Password Worksheet (p. 33), use it to record your HSM passwords, and store it in a secure place. We also recommended that you store at least one copy of this worksheet in secure off-site storage. AWS does not have the ability to recover your key material from an HSM for which you do not have the proper HSM security officer credentials.

When you create an HSM, the HSM is assigned an IP address. Because this IP address is only accessible from an instance within the same VPC that the HSM is in, you need to use the `control instance` to initialize and manage the HSM. The control instance was launched when the AWS CloudHSM environment was set up.

Topics

- Get the HSM ENI Identifier and IP Address (p. 25)
- Apply the Security Group (p. 25)
- Initialize the HSM (p. 26)
- Connect Your On-Premises HSM (p. 27)

Note
All of the example commands assume that you have set **aws_access_key_id**, **aws_secret_access_key**, and **aws_region** in a configuration file at `~/cloudhsm.conf`. For more information, see Configuration Files (p. 19).

Get the HSM ENI Identifier and IP Address

To find the IP address of an HSM, perform the following steps:

To find the IP address of an HSM

1. Connect to the control instance using SSH. The remaining steps are performed from the control instance.
2. If you do not know the ARN of the HSM, issue the **list-hsms (p. 87)** command and copy the ARN of the HSM in question.

```
$ cloudhsm list-hsms --conf_file ~/cloudhsm.conf
{
    "HsmList": [
        "<hsm1_arn>",
        "<hsm2_arn>"
    ],
    "RequestId": "<request_id>"
}
```

3. Issue the **describe-hsm (p. 78)** command, passing the ARN of the HSM. The ENI identifier is contained in the **EniId** field and the IP address of the HSM is contained in the **EniIp** field. Make note of these values as these are needed to initialize your HSM.

```
$ cloudhsm describe-hsm --conf_file ~/cloudhsm.conf --hsm-arn <hsm_arn>
{
    "EniId": "<eni_id>",
    "EniIp": "<eni_ip>",
    "HsmArn": "<hsm_arn>",
    "IamRoleArn": "<iam_role_arn>",
    "Partitions": [],
    "RequestId": "<request_id>",
    "SerialNumber": "<serial_number>",
    "SoftwareVersion": "5.1.3-1",
    "SshPublicKey": "<public_key_text>",
    "Status": "<status>",
    "SubnetId": "<subnet_id>",
    "SubscriptionStartDate": "2014-02-05T22:59:38.294Z",
    "SubscriptionType": "PRODUCTION",
    "VendorName": "SafeNet Inc."
}
```

Apply the Security Group

After you provision your HSM, you must apply the correct security group to your HSM.

To apply the security group

1. Open the Amazon EC2 console and select the region that your HSM was provisioned in.
2. Select **Network Interfaces** in the console navigation pane.
3. Find and select the network interface identifier of your HSM in the list of network interfaces, click **Actions**, and select **Change Security Groups**.
4. In the **Change Security Groups** dialog box, select the security group you created for your HSMs, and click **Save**.

5. (Optional) To aid in troubleshooting network connectivity to your HSM appliance, add incoming and outgoing rules to your security group for ICMP Echo Request and Echo Reply. These allow you to ping the HSM appliance, and allows the HSM appliance to respond.

Initialize the HSM

To initialize an HSM using the AWS CloudHSM CLI, perform the following steps from your control instance. If you need to initialize your HSM manually, see Manually Initialize an HSM (p. 98).

To initialize an HSM

1. (Optional) If needed, obtain the IP address of the HSM using the following **describe-hsm (p. 78)** command. This is required to connect to the HSM in the next step.

```
$ cloudhsm describe-hsm --conf_file ~/cloudhsm.conf --hsm-arn <hsm_arn>
```

   The output is similar to the following. Make note of the *<eni_ip>* value.

```
{
    "AvailabilityZone": "<az_id>",
    "EniId": "<eni_id>",
    "EniIp": "<eni_ip>",
    "HsmArn": "<hsm_arn>",
    "IamRoleArn": "arn:aws:iam::<account>:role/<role_name>",
    "Partitions": [
        "arn:aws:cloudhsm:<region>:<account>:<hsm_id>/<partition_id>",
        "arn:aws:cloudhsm:<region>:<account>:<hsm_id>/<partition_id>",
        "arn:aws:cloudhsm:<region>:<account>:<hsm_id>/<partition_id>"
    ],
    "RequestId": "<request_id>",
    "SerialNumber": "<serial_number>",
    "SoftwareVersion": "<version>",
    "SshPublicKey": "<public_key_contents>",
    "Status": "<status>",
    "SubnetId": "<subnet_id>",
    "SubscriptionStartDate": "<start_date>",
    "SubscriptionType": "<subscription_type>",
    "VendorName": "<vendor>"
}
```

2. Open a persistent SSH connection with the HSM by following the instructions in SSH Connections (p. 19), using the HSM IP address obtained in the previous step.
3. Initialize the HSM using the following **initialize-hsm (p. 82)** command.

```
$ cloudhsm initialize-hsm --conf_file ~/cloudhsm.conf \
--hsm-arn <hsm_arn> \
--label <label> \
--cloning-domain <cloning_domain> \
--so-password <so_password>
```

   The parameters are as follows:

   <hsm_arn>
   The identifier of the HSM you want to initialize.

<label>
A unique name for the HSM.

<cloning_domain>
The cloning domain for the HSM, which is a secret used to control cloning of key material from one HSM to another. If you are going to clone an HSM using the **clone-hsm (p. 61)** command, both the source and destination HSM must be initialized with the same cloning domain.

<so_password>
The password to set for the security officer account on the HSM. Record this password on your Password Worksheet (p. 33).

Initializing an HSM also sets the password for the HSM security officer account (also known as the administrator). This password must be the same for all HSMs in the same high-availability partition group. Record the security officer password on your Password Worksheet (p. 33) and do not lose it. We recommend that you print out a copy of the Password Worksheet (p. 33), use it to record your AWS CloudHSM passwords, and store it in a secure place. We also recommended that you store at least one copy of this worksheet in secure off-site storage. AWS does not have the ability to recover your key material from an HSM for which you do not have the proper HSM security officer credentials.

Repeat the **initialize-hsm (p. 82)** command for each HSM ARN that you want to initialize.

4. Close the persistent SSH connection with the HSM using the following command.

```
$ ssh -O stop <hsm_ip_address>
```

Connect Your On-Premises HSM

If you desire, you can connect the SafeNet Luna SA HSM appliances in your data center to your AWS instances using VPN or AWS Direct Connect. For more information, see the AWS Direct Connect detail page.

Configuring Your HSM Client

Read the following topics to learn how to install the HSM client software.

Topics

- Configuring a Linux HSM Client (p. 27)
- Configuring a Windows HSM Client (p. 29)

Configuring a Linux HSM Client

To configure a Linux HSM client, you must install the HSM client software on your Linux client instance. AWS CloudHSM offers a custom Amazon Machine Image (AMI) that you can use to launch an Amazon Elastic Compute Cloud (Amazon EC2) instance that is preconfigured with the HSM client software. If you set up your environment automatically with AWS CloudFormation, or if you used the AWS CloudHSM AMI to launch your client instance, you can skip to Creating a Network Trust Link Between a Linux Client and the HSM Appliance (p. 28).

You can also install the HSM client software manually. To manually install the HSM client software on an EC2 instance that was not launched from the CloudHSM Client AMI, see the following instructions.

The following steps are for the Amazon Linux x86 64-bit AMI and may require changes if you use a different system architecture.

To manually install and configure a Linux HSM client

1. Connect to the Linux instance on which to install the HSM client. The instance must be running in the same VPC as your HSM.
2. Download the client software package at https://s3.amazonaws.com/cloudhsm-software/Luna_5_4_Client_Software.tgz. You can verify the integrity of the downloaded package with the following SHA 256 digest:

   ```
   4777ae559cfa9421735f73b4c1a2fe69b2f43d4d774f36e4050e773c23372f4c
   ```

   This digest is also available at https://s3.amazonaws.com/cloudhsm-software/Luna_5_4_Client_Software.tgz.sha256.
3. Extract the files from the package, and then run the `610-012382-008_revC/linux/64/install.sh` file as root and install the **Luna SA** option.

Creating a Network Trust Link Between a Linux Client and the HSM Appliance

The following instructions use the `vtl` application, which is part of the Luna SA client tools installed previously. The `vtl` application is installed in `/usr/safenet/lunaclient/bin/`. You must include this path each time you use the command or add it to the `PATH` environment variable.

> **Note**
> To complete these steps for more than one HSM appliance, perform all steps for the first HSM. Then start over and complete all steps for the second HSM, and so on.

To create a network trust link between the client and the HSM appliance

These instructions apply to Amazon Linux x86 64-bit and may require changes based on your system architecture.

1. Copy the server certificate from the HSM to the client instance by entering the following command on the client.

   ```
   $ scp -i ~/.ssh/<private_key_file> manager@<hsm_ip_address>:server.pem .
   ```

   The *<private_key_file>* is the name of the SSH private key file used to connect to the HSM.

   The dot (**.**) at the end of the command is required and causes `scp` to copy the resulting file to the current directory.
2. Register the HSM certificate with the client.

   ```
   $ sudo vtl addServer -n <hsm_ip_address> -c server.pem

   New server <hsm_ip_address> successfully added to server list.
   ```
3. Create a client certificate for your client instance.

   ```
   $ sudo vtl createCert -n <client_name>

   Private Key created and written to:
   <client_cert_directory>/<client_name>Key.pem
   Certificate created and written to:
   ```

```
<client_cert_directory>/<client_name>.pem
```

The *<client_name>* can be any name that is unique and does not contain spaces or special characters.

Note
You can also create certificates to be shared among multiple instances. For more information, see Creating an AMI with the HSM Client Configuration (p. 105).

4. Copy the client certificate to the HSM.

```
$ scp -i
 ~/.ssh/<private_key_file> <client_cert_directory>/<client_name>.pem
 manager@<hsm_ip_address>:
```

Note
The colon (**:**) after the destination is required. Without it, `scp` does not recognize the supplied destination as a remote server.

5. Connect to the HSM over SSH. The *<private_key_file>* is the private portion of the SSH key that you provided when your HSM was provisioned.

```
$ ssh -i ~/.ssh/<private_key_file> manager@<hsm_ip_address>

lunash:>
```

6. Register the client with the `client register` command.

```
lunash:> client register -client <client_id> -hostname <client_name>

'client register' successful.
```

The *<client_name>* must be the same name used for the preceding `createCert` command. The *<client_id>* can be any name that is unique and does not contain spaces or special characters. To prevent confusion, we suggest you keep these two names the same.

Note
You can create certificates to be shared among multiple instances. For more information, see Creating an AMI with the HSM Client Configuration (p. 105).

To register the client with a high-availability partition group, see Register a Client with a High-Availability Partition Group (p. 39).

Configuring a Windows HSM Client

To configure a Windows HSM client, you must manually install the HSM client software on your Windows client instance.

To configure a Windows HSM client

1. Connect to the Windows instance on which to install the HSM client. The instance must be running in the same VPC as your HSM.
2. Download the client software package at https://s3.amazonaws.com/cloudhsm-software/Luna_5_4_Client_Software.tgz. You can verify the integrity of the downloaded package with the following SHA 256 digest:

   ```
   4777ae559cfa9421735f73b4c1a2fe69b2f43d4d774f36e4050e773c23372f4c
   ```

This digest is also available at https://s3.amazonaws.com/cloudhsm-software/Luna_5_4_Client_Software.tgz.sha256.

3. Extract the files from the package, and then run the `610-012382-008_revC\windows\64\LunaClient.msi` file and install the **Luna SA** option.

Creating a Network Trust Link Between a Windows Client and the HSM Appliance

The following instructions use the `vtl` application, which is part of the Luna SA client tools installed previously. The `vtl` application is installed in `%ProgramFiles%\SafeNet\LunaClient\bin\`. You must include this path each time you use the command or add it to the `PATH` environment variable.

Note
To complete these steps for more than one HSM appliance, perform all steps for the first HSM. Then start over and complete all steps for the second HSM, and so on.

To create a network trust link between a Windows client and the HSM appliance

1. Copy the server certificate from the HSM to the client instance using the `pscp` utility.

```
> pscp -i <private_key_file>.ppk manager@<hsm_ip_address>:server.pem .
```

The *<private_key_file>* is the path and file name of the PuTTY private key file that is used to connect to the HSM appliance.

Note
The dot (**.**) at the end of the command is required and causes `pscp` to copy the resulting file to the current directory.

2. Register the HSM server certificate with the client.

Important
You must execute this command as an administrator. To do this, right-click the **cmd.exe** icon and choose **Run as Administrator**.

```
> vtl addServer -n <hsm_ip_address> -c server.pem
```

3. Create a client certificate.

Important
You must execute this command as an administrator. To do this, right-click the **cmd.exe** icon and select **Run as Administrator**.

```
> vtl createCert -n <client_name>

Private Key created and written to:
<client_cert_directory>\<client_name>Key.pem
Certificate created and written to:
<client_cert_directory>\<client_name>.pem
```

The *<client_name>* can be any name that is unique and does not contain spaces or special characters.

4. Copy the client certificate to the HSM.

```
> pscp -i <private_key_file> <client_cert_directory>\<client_name>.pem
 manager@<hsm_ip_address>:
```

The *<private_key_file>* is the path and file name of the PuTTY private key file that is used to connect to the HSM appliance.

Note
The colon (**:**) after the destination is required. Without it, `pscp` does not recognize the supplied destination as a remote server.

5. Connect to your HSM using PuTTY and register the client.

```
lunash:> client register -client <client_id> -hostname <client_name>

'client register' successful.
```

The *<client_name>* must be the same name used for the `createCert` command above. The *<client_id>* can be any name that is unique and does not contain spaces or special characters. To prevent confusion, we suggest you keep these two names the same.

6. Assign the client to a partition.

```
lunash:> client assignPartition -client <client_id> -
partition <partition_name>
```

To register the client with a high-availability partition group, see Register a Client with a High-Availability Partition Group (p. 39).

Best Practices

Topics

General Best Practices

- Use a high availability (HA) configuration. AWS recommends that you use two or more HSM appliances, in separate Availability Zones, in an HA configuration, to avoid data loss in the case that an Availability Zone becomes unavailable. For more information about how to set up an HA configuration, see High Availability and Load Balancing (p. 35).
- Initializing an HSM irrevocably destroys the key material inside the HSM. Never initialize the HSM unless you are certain that the keys have been backed up somewhere else or that the keys are no longer required.
- Do not apply software patches or updates to the appliance. Contact AWS Support if you need the software updated.
- Do not change the network configuration of the appliance.
- Do not remove or change the syslog forwarding configuration that is provided on the appliance. You may add additional destinations for syslog messages, as long as you do not change or remove the ones that are already there.
- Do not change or remove any SNMP configuration that is provided on the appliance. You may add additional SNMP configuration as long as you do not disturb the configuration that is already present.
- Do not change the NTP configuration that is provided on the appliance.

Best Practices for Passwords

- Make a note of the HSM security officer (also known as the administrator) password on your Password Worksheet (p. 33) and do not lose the worksheet. We recommend that you print out a copy of the Password Worksheet (p. 33), use it to record your AWS CloudHSM passwords, and store it in a secure place. We also recommended that you store at least one copy of this worksheet in secure off-site storage. AWS does not have the ability to recover your key material from an HSM for which you do not have the proper HSM security officer credentials.

- Do not change the HSM appliance administrator password. AWS uses this password for service delivery.
- You should use an SSH key for the manager account login. For more information, see Generating an SSH Key (p. 13). AWS can re-create the manager account if you lose access to the account. You can optionally set a password for the manager account if you prefer.
- HSM partition passwords must be coordinated with clients and applications that depend on the passwords. For information about using IAM roles to distribute passwords, see the Using IAM roles to distribute non-AWS credentials to your EC2 instances blog post.

Password Worksheet

Use the following worksheet to compile information for your AWS CloudHSM appliances. Print this page and use it to record your AWS CloudHSM passwords, and store it in a secure place. We also recommended that you store at least one copy of this worksheet in secure off-site storage.

Security Officer Password
This password was set when you initialized the HSM appliance.

_____

Manager Password (Optional)
This password was optionally set with the **user password manager** command on the HSM appliance.

_____

Partition Passwords

Partition Label	Password	Cloning Domain

Operations and Maintenance

AWS monitors your HSM appliances, and may correct minor configuration issues related to availability of the appliance. Such operations do not interfere with your use of the HSM appliance.

If a management operation must be performed which could disrupt service, then AWS provides 24 hours' notice before performing the operation.

It is possible that, in unforeseen circumstances, AWS might have to perform maintenance on an emergency basis without prior notice. We try to avoid this situation. However, if availability is a concern, AWS strongly recommends that you use two or more HSM appliances in separate Availability Zones in a high availability configuration. The failure of a single HSM appliance in a non-HA configuration can result in the permanent loss of keys and data.

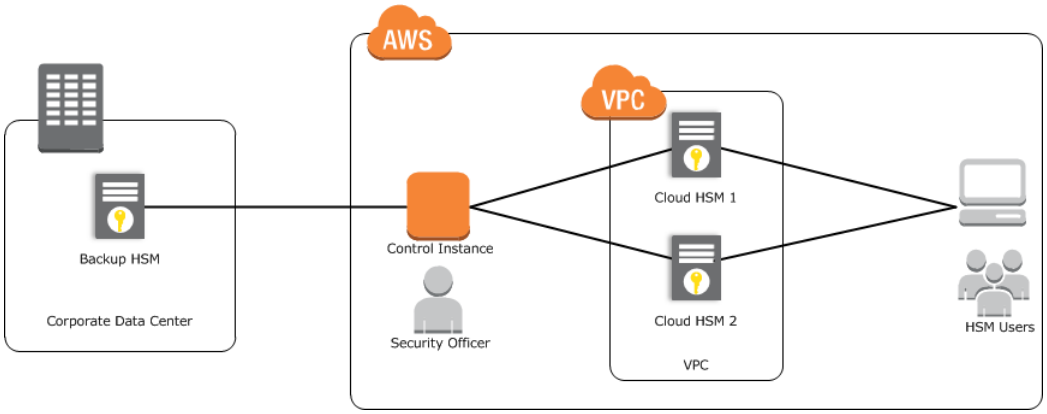
AWS does not perform routine maintenance on HSM appliances in multiple Availability Zones within the same region within the same 24-hour period.

For information about how to set up a high availability configuration, see High Availability and Load Balancing (p. 35).

For information about administration and maintenance of your HSM appliance, go to Administering Your Luna SA in the SafeNet Luna SA documentation.

High Availability and Load Balancing

The recommended configuration for using AWS CloudHSM is to use two HSMs configured in a high-availability (HA) configuration. The failure of a single HSM appliance in a non–HA configuration can result in the permanent loss of keys and data. A minimum of two HSMs are suggested for HA purposes, with each HSM in a different Availability Zone. With this configuration, if one of your HSMs is unavailable, your keys are still available. This topic contains information about how to set up a traditional HA configuration.

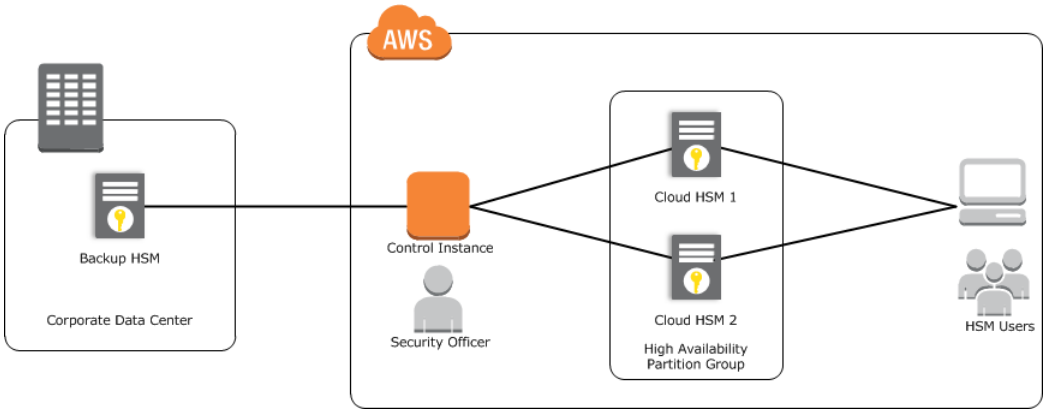


HA allows multiple HSMs to be grouped together to form one virtual device, or logical unit, as seen from the client, similar to clustering or RAID technologies. In an HA configuration, service is maintained even if one or more HSMs are unavailable. For example, if three HSMs are combined into an HA group, service is maintained even if two HSMs are offline.

When configured for HA, each HSM joins an HA group, managed through the HSM client. To HSM clients, the HA group appears as a single HSM. However, from an operational perspective, the members in the HA group share the transaction load, synchronize data with each other, and gracefully redistribute the processing capacity in the event of failure in a member HSM, to maintain uninterrupted service to the clients. HA provides load balancing across all member HSMs to increase performance and response time, while providing the assurance of HA service. All member HSMs are active (rather than one active and the rest passive). Calls are passed from each client application through the HSM client-side software (library) to one of the member HSMs on a least-busy basis.

For more information, go to Overview of Luna High Availability and Load Balancing. For more information about HA best practices, see Best Practices for High Availability and Load Balancing (p. 36).

The AWS CloudHSM service defines a resource known as a high-availability (HA) partition group. A high-availability partition group is a virtual partition that represents a group of partitions, typically distributed between several physical HSMs for high-availability. You use the AWS CloudHSM command line interface tools to create and manage your high-availability partition groups.



Topics

Best Practices for High Availability and Load Balancing

AWS recommends the following best practices for high availability (HA) and load balancing your HSM appliances.

Topics

General Best Practices

- When an HA group is shared by multiple AWS CloudHSM clients, the best practice is for these clients to select different primary HA members, for better fault tolerance and more equal distribution of the workload of cryptographic operations.

For more information, see the following topics in the SafeNet Luna SA documentation:

- Overview of Luna High Availability and Load Balancing
- HA with Luna SA

Best Practices for Loss and Recovery

Topics

- Recovering From the Loss of All High-Availability Members (p. 38)

High-Availability Recovery

High-availability (HA) recovery is hands-off resumption by failed HA group members. Prior to the introduction of this function, the HA feature provided redundancy and performance, but required that a failed/lost group member be manually reinstated. If the HA recovery feature is not switched on, HA still requires manual intervention to reinstate members. A member of a HA group may fail for the following reasons:

- The HSM appliance loses power, but regains power in less than the two hours that the HSM appliance preserves its activation state.
- The network connection is lost.

HA recovery works if the following are true:

- HA autoRecovery is enabled.
- The HA group has at least two nodes.
- The HA node is reachable (connected) at startup.
- The HA node recover retry limit is not reached. If it is reached or exceeded, the only option to restore the downed connections is a manual recovery.

If all HA nodes fail (there are no links from the HSM client), recovery is not possible.

The HA recovery logic in the library makes its first attempt at recovering a failed member when your application makes a call to its HSM appliance (the HA group). In other words, an idle HSM client does not attempt a recovery.

However, a busy HSM client would notice a slight pause every minute, as the library attempts to recover a dropped HA group members until the members are reinstated, or until the retry period has been reached/exceeded and it stops trying. Therefore, set the retry period according to your normal operational situation; for example, the types and durations of network interruptions you experience.

HA autoRecovery is not on by default. It must be explicitly enabled by following the instructions in Enabling Automatic Recovery (p. 104).For more information about HA and autoRecovery, go to the following topics in the SafeNet Luna SA documentation:

- Configuring HA
- Client - Create HA Group

Recovering From the Loss of a Subset of High-Availability Members

If there is a loss of a subset of HA members, AWS recommends the following procedure to recover group members.

When you are notified by AWS that the connection has been recovered, execute the following command to reintroduce disconnected members to the HA group:

```
vtl haAdmin recover -group <ha_group_label>
```

AWS also recommends retrying the connection for a short period of time, so that any disconnections caused by transient network outages can be automatically recovered. For example, retry the connection 5 times, at an interval of one try every minute, as shown below.

```
vtl haAdmin autoRecovery -interval 60
vtl haAdmin autoRecovery -retry 5
```

If you don't want to recover the group members manually, but still want to minimize the overhead caused by automatic recovery, use the following steps:

To recover group members and minimize recovery overhead

- Retry the connection once every 3 minutes, until the connection is successful.

```
vtl haAdmin autoRecovery -interval 180
vtl haAdmin autoRecovery -retry -1
```

To recover group members with a special cryptographic application

- For special cryptographic applications, discuss with SafeNet or AWS on a case-by-case basis.

Recovering From the Loss of All High-Availability Members

If there is a loss of all HA members (there is a complete loss of communication with all the members of your HA group), you can use `LunaSlotManager.reinitialize()`. If you use `LunaSlotManager.reinitialize()`, you do not have to restart your applications. Alternately, you can restart your applications and use manual recovery.

For more information about `LunaSlotManager.reinitialize()`, see LunaProvider: Recovering from the Loss of all HA Members Using LunaSlotManager.reinitialize() in the SafeNet Luna SA Technical Notes.

Important

- `LunaHAStatus.isOK()` returns `true` only when all HA members are present. This method returns `false` when at least one HA member is missing, and throws an exception when all HA members are missing.
- The `HA-only` option has to be enabled to keep the HA slot number unchanged.

Creating a HA Partition Group

Creating a HA partition group is a two-step process. You create the HA partition group, and then register the clients for use with the HA partition group.

Tasks

- Create the HA Partition Group (p. 38)
- Register a Client with a High-Availability Partition Group (p. 39)

Create the HA Partition Group

To create an HA partition group, complete the following procedure.

> **Note**
> All of the example commands assume that you have set **aws_access_key_id**, **aws_secret_access_key**, and **aws_region** in a configuration file at `~/cloudhsm.conf`. For more information, see Configuration Files (p. 19).

To create and initialize an HA partition group

1. Create an HA partition group using the following **create-hapg (p. 66)** command.

```
$ cloudhsm create-hapg --conf_file ~/cloudhsm.conf --group-label <label>
```

   <label> is a unique name for the HA partition group.

2. Add your initialized HSMs to the HA partition group using the following **add-hsm-to-hapg (p. 57)** command.

```
$ cloudhsm add-hsm-to-hapg --conf_file ~/cloudhsm.conf \
--hsm-arn <hsm_arn> \
--hapg-arn <hapg_arn> \
--cloning-domain <cloning_domain> \
--partition-password <partition_password> \
--so-password <so_password>
```

   The parameters are as follows:

   <hsm_arn>
   : The identifier of the HSM to add to the HA partition group.

   <hapg_arn>
   : The identifier of the HA partition group.

   <cloning_domain>
   : The cloning domain for the HA partition group.

   <partition_password>
   : The password for the member partitions. Record this password on your Password Worksheet (p. 33). This must be the same for all HSMs in the same HA partition group.

   <so_password>
   : The security officer password for *<hsm_arn>*.

   If the command is successful, the output is similar to the following:

```
{
   "Status": "Addition of HSM <hsm_arn> to HAPG <hapg_arn> successful"
}
```

   Save the partition password on your Password Worksheet (p. 33).

   Save the HA partition group ARN returned from the **create-hapg (p. 66)** command for later use.

3. Repeat the previous step for each HSM you want to include in the HA partition group.

Register a Client with a High-Availability Partition Group

To allow a client to use an HA partition group, you must complete the following tasks.

> **Note**
> All of the example commands assume that you have set **aws_access_key_id**, **aws_secret_access_key**, and **aws_region** in a configuration file at `~/cloudhsm.conf`. For more information, see Configuration Files (p. 19).

Tasks

Create the Client

Before you can create a client, you must create a certificate for the client as explained in Client Certificates (p. 20).

After you have the certificate, create the client using the following **create-client (p. 64)** command.

```
$ cloudhsm create-client --conf_file ~/cloudhsm.conf --certificate-
file <client_cert_file>
```

If the command is successful, the output is similar to the following:

```
{
    "ClientArn": "<client_arn>",
    "RequestId": "<request_id>"
}
```

Make note of the *<client_arn>* value as this is needed to register the client.

Register the Client

After the client is created, register the client with the HA partition group using the following **register-client-to-hapg (p. 90)** command.

```
$ cloudhsm register-client-to-hapg --conf_file ~/cloudhsm.conf \
--client-arn <client_arn> \
--hapg-arn <hapg_arn>
```

If the command is successful, the output is similar to the following:

```
{
    "Status": "Registration of the client <client_arn> to the HA partition
 group <hapg_arn> successful"
}
```

Generate the Client Configuration

After the client is registered, you get the client configuration file and server certificates.

To assign the client to the HA partition group, use the following **get-client-configuration (p. 80)** command on the client:

```
$ cloudhsm get-client-configuration --conf_file ~/cloudhsm.conf \
--client-arn <client_arn> \
--hapg-arns <hapg_arn> \
--cert-directory <server_cert_location> \
```

```
--config-directory /etc/
```

If the command is successful, the output is similar to the following:

```
The configuration file has been copied to /etc/
The server certificate has been copied to /usr/safenet/lunaclient/cert/server
```

Verify the Client Configuration

Verify your setup using the following command, then point your client application at the HSM, referring to this HSM by the HA group label that you specified.

```
>vtl haAdmin show
```

In the output, under the heading "HA Group and Member Information", confirm that the number of group members equals the number of HSMs in the HA partition group.

Replicating Keys Across HSMs

Note
All of the example commands assume that you have set **aws_access_key_id**, **aws_secret_access_key**, and **aws_region** in a configuration file at `~/cloudhsm.conf`. For more information, see Configuration Files (p. 19).

If needed, you can clone the contents of an existing HA partition group to a new HA partition group using the following **clone-hapg (p. 59)** command. When you create the new HA partition group, you must specify the same cloning domain and partition password as the source HA partition group.

```
$ cloudhsm clone-hapg --conf_file ~/cloudhsm.conf \
--src-hapg-arn <src_arn> \
--dest-hapg-arn <dest_arn> \
--hapg-password <hapg_password>
```

The parameters are as follows:

<src_arn>
The identifier of the HA partition group to clone from. Both HA partition groups must have the same cloning domain and partition password.

<dest_arn>
The identifier of the HA partition group to clone to. Both HA partition groups must have the same cloning domain and partition password.

<hapg_password>
The password for the partition group. Both HA partition groups must have the same password.

If the command is successful, the output is similar to the following:

```
cloudhsmcli.hapg_cloner: Backing up existing config files
cloudhsmcli.hapg_cloner: Collecting information about the HA Partition groups
cloudhsmcli.hapg_cloner: Setting up a cloning environment
cloudhsmcli.hapg_cloner: Cloning the HA partition groups
cloudhsmcli.hapg_cloner: Cleaning up the cloning environment
cloudhsmcli.hapg_cloner: Restoring existing config files
{
   "Status": "Completed cloning the HA partition group <src_arn> to the HA
 partition group <dest_arn>"
}
```

Save the HA partition group ARN returned from the **clone-hapg (p. 59)** command for later use.

Backing Up and Restoring HSM Data to a Luna SA Backup HSM

In addition to the AWS recommendation that you use two or more HSM appliances in a high-availability configuration to prevent the loss of keys and data, you can also perform a remote backup/restore of a Luna SA partition if you have purchased a Luna Backup HSM. For more information on the Luna Backup HSM, download the Luna Backup HSM Product Brief.

The Luna Backup HSM ensures that your sensitive cryptographic material remains strongly protected in hardware even when it is not being used. You can easily back up and duplicate keys securely to the Luna Backup HSM for safekeeping in case of emergency, failure, or disaster.

The remote backup capabilities allow administrators to move copies of their sensitive cryptographic material securely to other SafeNet HSMs. With a single Luna Backup HSM, an administrator can back up and restore keys to and from up to 20 Luna HSM appliances.

The Luna Backup HSM is attached to a client machine directly via USB. The client machine is either a Windows or a Linux machine that is outside of AWS, that has the SafeNet Luna client software installed on it. The client machine must also have IP connectivity to your HSM in the AWS cloud.

Topics

Backing Up HSM Data Using Windows

To back up HSM data using Windows

1. Connect the Luna Backup HSM to your Windows computer using USB. For more information about the Luna Backup HSM, see the Luna Backup HSM Product Brief.
2. Install the Luna Remote Backup Driver (610-011646-001) from the following location:

   http://c3.safenet-inc.com/downloads/F/E/FEAB55E0-5B3F-4DFD-8DEF-B068C5531AED/610-011646-001.tar
3. In **Control Panel**, open **Device Manager**, select **Luna G5 Device**, then right-click and select **Update Driver Software**.

4. Complete the steps in Configuring Your HSM (p. 24) and Configuring Your HSM Client (p. 27).
5. Using PuTTY, connect to your HSM over SSH.
6. Execute the following command on your HSM to display the details of the HSM appliance:

```
lunash:> hsm show
```

7. Execute the following command on your HSM to display the contents of the partition:

```
lunash:> par showc -par pm
```

8. Establish an NTLS connection by executing the following command from the Windows command prompt:

```
C:\> vtl verify
```

9. List the available slots by executing the following command:

```
C:\> vtl listslots
```

10. Restore the Luna Backup HSM appliance to its factory settings by executing the following command. When prompted, type **yes** to confirm.

```
C:\> vtl backup token factoryreset -target 2
```

11. Initialize the Luna Backup HSM appliance by executing the following command. Type **yes** when prompted to initialize the HSM, and **no** when prompted to use PED authentication.

```
C:\> vtl backup token init -target 2 -label BackupHSM
```

Important
It is important that your HSM uses password authentication.

12. Execute the remote backup command:

```
C:\> vtl backup -source 1 -target 2 -partition pm_backup
```

13. Type **yes** when prompted to create the new backup.
14. If you want to check the details of the backup, execute the following command:

```
C:\> vtl backup token show -target 2
```

Restoring HSM Data from a Luna Backup HSM

To restore HSM data

1. Using PuTTY, connect to your HSM over SSH.
2. Log into the HSM as the HSM administrator (Security Officer).

```
lunash:> hsm login
```

3. Clear the contents of the partition by executing the following from your HSM. When prompted, enter your password for this partition and type **proceed**.

```
lunash:> partition clear -partition pm
```

4. Verify that the partition is cleared by executing the following command:

```
lunash:> partition showcontents -partition pm
```

5. Confirm that no objects exist on the HSM partition by executing the following command from the Windows command prompt:

```
C:\> cmu li
```

6. Initiate the restore by executing the following command. Enter the passwords when prompted.

```
C:\> vtl backup restore -source 2 -partition pm_backup -target 1
```

7. Confirm that the restore was successful by executing the following from the HSM. Enter your password when prompted.

```
lunash:> partition showcontents -partition pm
```

8. Verify that the client can access the HSM objects that have been restored by executing the following command:

```
C:\> cmu li
```

Integrating Third-Party Applications with AWS CloudHSM

This chapter describes how to use third-party applications with AWS CloudHSM.

Topics

If the application that you are looking for is not listed, contact AWS Support or see HSM Interoperability on the SafeNet website.

Transparent Data Encryption with AWS CloudHSM

Transparent Data Encryption (TDE) reduces the risk of confidential data theft by encrypting sensitive data, such as credit card numbers, stored in application table columns or tablespaces (the containers for all objects stored in a database).

The following topic describes how to configure an Oracle or Microsoft SQL Server database using TDE while storing the master encryption key in AWS CloudHSM.

Oracle Database TDE with AWS CloudHSM

These instructions explain how to integrate an Oracle database and your HSM, and also cover the necessary information to install, configure, and integrate an Oracle database with AWS CloudHSM.

To set up TDE for Oracle Database 11g

The following instructions are explained in detail in the Oracle Database LunaSA/PCI Integration Guide on the SafeNet website.

1. Set up your Luna SA/PCI/HSM appliances. For more information, see the instructions in Setting Up AWS CloudHSM (p. 3).
2. Install Oracle Database 11g on the target machine.
3. Integrate Oracle Database 11g R1 (11.1.0.6 or 11.1.0.7) or 11g R2 (11.2.0.1, 11.2.0.2, or 11.2.0.3) with your HSMs.

Microsoft SQL Server with AWS CloudHSM

The following topic describes how to use Microsoft SQL Server TDE and the Extensible Key Management (EKM) Library with AWS CloudHSM.

For more information about the EKM library, go to http://technet.microsoft.com/en-us/library/bb895340.aspx

To set up TDE for Microsoft SQL Server and the EKM Library

The following instructions are explained in detail in the Microsoft SQL Server Integration Guide on the SafeNet website.

1. Set up your HSM appliance(s). Refer to the instructions in Setting Up AWS CloudHSM (p. 3).
2. Integrate Luna SA/PCI/HSM appliances with Microsoft SQL Server.
3. Download and install the EKM libraries from SafeNet.

Volume Encryption for Amazon Elastic Block Store

To use volume encryption for Amazon Elastic Block Store (Amazon EBS) with SafeNet KeySecure, SafeNet ProtectV, and AWS CloudHSM, see the Gemalto SafeNet products in AWS Marketplace.

Encryption with Amazon Simple Storage Service (S3) and SafeNet KeySecure

For information about how to use Amazon Simple Storage Service (Amazon S3) encryption with SafeNet ProtectApp and SafeNet KeySecure, see the SafeNet KMIP and Amazon S3 Integration Guide on the SafeNet website.

Setting Up SSL Termination on an Apache Web Server with Private Keys Stored in AWS CloudHSM

The SafeNet Luna HSM appliances integrate with the Apache HTTP server to provide significant performance improvements by offloading cryptographic operations from the Apache HTTP Server to the SafeNet Luna HSM appliances. In addition, the Luna HSM appliances provide extra security by protecting and managing the server's high-value SSL private key within a FIPS 140-2 certified hardware security module. For more information about the libraries that are required for the Apache

integration, see the Apache HTTP Server Integration Guide on the SafeNet website. SafeNet's OpenSSL Toolkit might also be required for integrating with the Apache web server.

Building Your Own Applications

For more information about how to configure your applications to use one or more of the API operations provided by the SafeNet client, go to Configured and Registered Client Using an HSM Partition and Integrating Luna SA with Your Applications in the SafeNet Luna SA documentation.

How to Stop Using an HSM

AWS does not ordinarily de-provision an HSM appliance that contains key material. This protects you, as well as AWS, from risks associated with accidentally destroying key material that is still in use.

> **Important**
> If you need to stop using an HSM appliance (such as when your subscription ends), back up the contents of the HSM to another HSM that you control, or confirm that the keys stored within the HSM are no longer needed.

Complete the following steps to stop using an HSM appliance.

To stop using an HSM appliance

1. From your control instance, connect to your HSM over SSH. *<private_key_file>* is the private portion of the SSH key you provided when your HSM was provisioned.

```
$ ssh -i <private_key_file> manager@<hsm_ip_address>
```

2. Zeroize the HSM by attempting to log into the HSM as the HSM administrator with an invalid password three times. For more information, see How do I zeroize my HSM (p. 54).
3. Declassify the HSM appliance by first executing the following command to rotate all logs.

```
lunash:> syslog rotate
```

4. Delete all logs.

```
lunash:> syslog cleanup
```

5. You use one of the following methods to de-provision the HSM:

   - Use the AWS CloudHSM command line interface tools to de-provision the HSM with the **delete-hsm (p. 72)** command.
   - Use the AWS CloudHSM API to de-provision the HSM with the **DeleteHsm** operation. For more information, see the *AWS CloudHSM Developer Guide*.

If you experience difficulties de-provisioning your HSM, please contact the AWS Support Center.

AWS reserves the right to terminate service and reinitialize an HSM in the case of non-payment.

SafeNet Luna SA Documentation

For more information about the SafeNet Luna SA appliance configuration, operation, and maintenance, go to the following documentation:

Luna SA 5.3
- Luna SA 5.3 Product Documentation

Luna SA 5.1
- Luna SA 5.1 Product Documentation

Logging AWS CloudHSM API Calls by Using CloudTrail

AWS CloudHSM is integrated with AWS CloudTrail, a service that captures API calls made by or on behalf of AWS CloudHSM in your AWS account and delivers the log files to an Amazon S3 bucket that you specify. CloudTrail captures API calls from the AWS CloudHSM API and CLI. Using the information collected by CloudTrail, you can determine what request was made to AWS CloudHSM, the source IP address from which the request was made, who made the request, when it was made, and so on. For more information about CloudTrail, including how to configure and enable it, see the *AWS CloudTrail User Guide*.

AWS CloudHSM Information in CloudTrail

When CloudTrail logging is enabled in your AWS account, API calls made to AWS CloudHSM actions are tracked in log files. AWS CloudHSM records are written together with other AWS service records in a log file. CloudTrail determines when to create and write to a new file based on a time period and file size.

Every log entry contains information about who generated the request. The user identity information in the log helps you determine whether the request was made with root or IAM user credentials, with temporary security credentials for a role or federated user, or by another AWS service. For more information, see the **userIdentity** field in the CloudTrail Event Reference.

You can store your log files in your bucket for as long as you want, but you can also define Amazon S3 lifecycle rules to archive or delete log files automatically. By default, your log files are encrypted by using Amazon S3 server-side encryption (SSE).

You can choose to have CloudTrail publish Amazon SNS notifications when new log files are delivered if you want to take quick action upon log file delivery. For more information, see Configuring Amazon SNS Notifications.

You can also aggregate AWS CloudHSM log files from multiple AWS regions and multiple AWS accounts into a single Amazon S3 bucket. For more information, see Aggregating CloudTrail Log Files to a Single Amazon S3 Bucket.

Understanding AWS CloudHSM Log File Entries

CloudTrail log files can contain one or more log entries where each entry is made up of multiple JSON-formatted events. A log entry represents a single request from any source and includes information about the requested action, any parameters, the date and time of the action, and so on. The log entries are not guaranteed to be in any particular order. That is, they are not an ordered stack trace of the public API calls.

Sensitive information, such as passwords, authentication tokens, file comments, and file contents are redacted in the log entries.

The following example shows an example of a CloudTrail log entry for AWS CloudHSM.

```
{
  "Records" : [
    {
      "userIdentity" :
      {
        "type" : "IAMUser",
        "principalId" : "<user_id>",
        "arn" : "<user_arn>",
        "accountId" : "<account_id>",
        "accessKeyId" : "<access_key_id>",
        "userName" : "<username>"
      },
      "eventTime" : "<event_time>",
      "eventSource" : "cloudhsm.amazonaws.com",
      "eventName" : "CreateHsm",
      "awsRegion" : "<region>",
      "sourceIPAddress" : "<IP_address>",
      "userAgent" : "<user_agent>",
      "requestParameters" :
      {
        "iamRoleArn" : "<IAM_role_arn>",
        "sshKey" : "<SSH_public_key>",
        "syslogIp" : "<syslog_ip>",
        "subscriptionType" : "<subscription_type>",
        "subnetId" : "<subnet_id>"
      },
      "responseElements" :
      {
        "hsmArn" : "<hsm_arn>"
      },
      "requestID" : "<request_id>",
      "eventID" : "<event_id>"
    }
  ]
}
```

Troubleshooting AWS CloudHSM

For frequently asked questions about AWS CloudHSM, see AWS CloudHSM FAQs.

Topics

My HSM isn't working. What do I do?

Contact AWS Support. Your incident will be routed to the team that supports AWS CloudHSM.

How do I zeroize my HSM

An HSM can be in one of two states: zeroized or not zeroized. Zeroized means that the HSM is blank and ready for customer use. Not zeroized means that it has key material or configuration on it already. If you need to keep any of the keys on your HSM, back up the HSM before you zeroize it. For information about backing up your key information, see Backing Up and Restoring HSM Data to a Luna SA Backup HSM (p. 44).

To zeroize your HSM, use SSH to connect to the HSM, and then attempt to log in as the administrator three times using an invalid password. This zeroizes the HSM.

```
lunash:> hsm login
```

Replace a Failed HSM

If one of your HSMs fails, you can replace it with the following procedure.

> **Note**
> All of the example commands assume that you have set **aws_access_key_id**, **aws_secret_access_key**, and **aws_region** in a configuration file at `~/cloudhsm.conf`. For more information, see Configuration Files (p. 19).

1. Remove the failed HSM from the HA partition group using the following **remove-hsm-from-hapg (p. 92)** command.

   This command requires SSH connectivity with the HSM. For more information, see SSH Connections (p. 19).

   ```
   $ cloudhsm remove-hsm-from-hapg --conf_file ~/cloudhsm.conf \
   --hsm-arn <hsm_arn> \
   --so-password <so_password>
   ```

   The parameters are as follows:

   <hsm_arn>
   The identifier of the failed HSM.

   <so_password>
   The security officer password for *<hsm_dest_arn>*.

2. Copy the HA partition group membership and key material to the new HSM using the following **clone-hsm (p. 61)** command.

   This command requires SSH connectivity with both the source and destination HSMs. For more information, see SSH Connections (p. 19).

   Warning
   You must not use the **clone-hsm (p. 61)** command from an instance that is also a client of the HSM being cloned.

   ```
   $ cloudhsm clone-hsm --conf_file ~/cloudhsm.conf \
   --source-hsm-arn <hsm_source_arn> \
   --dest-hsm-arn <hsm_dest_arn> \
   --so-password <so_password>
   ```

   The parameters are as follows:

   <hsm_source_arn>
   The identifier of an operational HSM in the same HA partition group as the failed HSM. This cannot be the failed HSM.

   <hsm_dest_arn>
   The identifier of the new HSM you want to clone to.

   <so_password>
   The security officer password for *<hsm_dest_arn>*.

   This command prompts the user for the password and cloning domain for every partition on the source HSM.

   This command copies all of the partitions and key material from the HA partition group that the source HSM is a member of to the destination HSM, and joins the destination HSM to the HA partition group.

AWS CloudHSM Command Line Interface Tools Reference

This is the *AWS CloudHSM Command Line Interface Tools Reference*. It provides descriptions, syntax, and usage examples for each of the commands for the AWS CloudHSM service.

Topics

Updating the AWS CloudHSM CLI Tools

To update the AWS CloudHSM CLI tools, download the latest stable egg file by running the following command on the instance:

```
$ wget https://s3.amazonaws.com/cloudhsm-software/CloudHsmCLI.egg
```

Update the CLI tools on the instance by running the following command, which overwrites the existing version:

```
$ sudo easy_install-2.7 -s /usr/local/bin CloudHsmCLI.egg
```

To verify that you have the AWS CloudHSM CLI tools correctly installed, issue the **version (p. 94)** command:

```
$ cloudhsm version
{
    "Version": "<version>"
}
```

AWS CloudHSM CLI Command Reference

Every AWS CloudHSM CLI command begins with **cloudhsm**, followed by the command identifier, and then the command options. For example:

```
$ cloudhsm [command] [option] ...
```

To display the list of commands supported by the AWS CloudHSM CLI tools, you can pass the **--help** option to the **cloudhsm** command.

```
$ cloudhsm --help
```

The AWS CloudHSM CLI tools contain the following commands:

Topics

add-hsm-to-hapg

Description

Adds an HSM to a high-availability (HA) partition group. A partition corresponding to the HA partition group is created on the HSM.

This command requires SSH connectivity with the HSM. For more information, see SSH Connections (p. 19).

Usage

```
cloudhsm add-hsm-to-hapg
```

```
--hsm-arn <value>
--hapg-arn <value>
--so-password <value>
--partition-password <value>
--cloning-domain <value>
--aws-region <value>
--aws-access-key-id <value>
--aws-secret-access-key <value>
[--aws-host <value>]
[--aws-port <value>]
[--conf_file <value>]
[--quiet]
[--verbose]
[--help]
```

Options

--hsm-arn

The ARN that identifies the HSM to add.

Required: Yes.

This can be specified in the **hsm_arn** setting in **--conf_file**.

--hapg-arn

The ARN that identifies the HA partition group to add the HSM to.

Required: Yes.

This can be specified in the **hapg_arn** setting in **--conf_file**.

--so-password

The HSM security officer password.

Required: Yes.

This can be specified in the **so_password** setting in **--conf_file**.

--partition-password

The password to set for the member partitions. The clients use this password to access the partition group.

Required: Yes

This can be specified in the **partition_password** setting in **--conf_file**.

--cloning-domain

The cloning domain for the partitions in the group. This is not the same as the **--cloning-domain** that is used in the initialize-hsm (p. 82) command.

Required: Yes

This can be specified in the **cloning_domain** setting in **--conf_file**.

--aws-region

The region identifier, such as `us-east-1`.

Required: Yes.

This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id

Your access key ID. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key

Your secret access key. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

The status of the operation.

```
{
   "Status": "Addition of HSM <hsm-arn> to HAPG <hapg-arn> successful"
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- remove-hsm-from-hapg (p. 92)

clone-hapg

Description

Copies the contents of a high-availability (HA) partition group to another HA partition group.

The cloning domain and partition password must be the same for both the source and destination HA partition group.

This command requires SSH connectivity with all HSMs in the both the source and destination HA partition groups. For more information, see SSH Connections (p. 19).

Warning
You must not issue this command from an instance that is also a client of the HA partition group being cloned.

Usage

```
cloudhsm clone-hapg
   --src-hapg-arn <value>
   --dest-hapg-arn <value>
   --hapg-password <value>
   [--force]
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--src-hapg-arn

The ARN that identifies the HA partition group to copy from. Both HA partition groups must have the same cloning domain and partition password.

Required: Yes

This can be specified in the **src_hapg_arn** setting in **--conf_file**.

--dest-hapg-arn

The ARN that identifies the HA partition group to copy to. Both HA partition groups must have the same cloning domain and partition password.

Required: Yes

This can be specified in the **dest_hapg_arn** setting in **--conf_file**.

--hapg-password

The password to be used to access the HA partition group. This password must be the same for both the source and destination HA partition group

Required: Yes

This can be specified in the **partition_password** setting in **--conf_file**.

--force

Do not display the safety check messages.

Required: No

This can be specified in the **force** setting in **--conf_file**.

--aws-region

The region identifier, such as `us-east-1`.

Required: Yes.

This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id

Your access key ID. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key

Your secret access key. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

The status of the operation.

```
cloudhsmcli.hapg_cloner: Backing up existing config files
cloudhsmcli.hapg_cloner: Collecting information about the HA Partition groups
cloudhsmcli.hapg_cloner: Setting up a cloning environment
cloudhsmcli.hapg_cloner: Cloning the HA partition groups
cloudhsmcli.hapg_cloner: Cleaning up the cloning environment
cloudhsmcli.hapg_cloner: Restoring existing config files
{
   "Status": "Completed cloning the HA partition group <src-hapg-arn> to the
 HA partition group <dest-hapg-arn>"
}
```

AWS CloudHSM command line tools display errors on stderr.

clone-hsm

Description

Copies the high-availability (HA) partition group memberships and key material from one HSM to another.

Both the source and destination HSM must be initialized with the same cloning domain.

This command prompts the user for the password and cloning domain for every partition on the source HSM.

This command copies all of the partitions and key material from the HA partition group that the source HSM is a member of to the destination HSM, and joins the destination HSM to the HA partition group.

You can remove a failed HSM from the HA partition group with the **remove-hsm-from-hapg (p. 92)** command.

This command requires SSH connectivity with both the source and destination HSMs. For more information, see SSH Connections (p. 19).

Warning
Running this command on a client of the HSM being cloned may temporarily disrupt HSM-backed applications running on the client.

Usage

```
cloudhsm clone-hsm
   --src-hsm-arn <value>
   --dest-hsm-arn <value>
   [--force]
   --so-password <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--src-hsm-arn
The ARN that identifies the HSM to copy from.
Required: Yes
This can be specified in the **src_hsm_arn** setting in **--conf_file**.

--dest-hsm-arn
The ARN that identifies the HSM to copy to.
Required: Yes
This can be specified in the **dest_hsm_arn** setting in **--conf_file**.

--force
Do not display the safety check messages.
Required: No
This can be specified in the **force** setting in **--conf_file**.

--so-password
The security officer password for **--dest-hsm-arn**.
Required: Yes
This can be specified in the **so_password** setting in **--conf_file**.

--ssh-username
The SSH username used to authenticate with **--dest-hsm-arn**.
Required: Yes
This can be specified in the **ssh_username** setting in **--conf_file**.

--ssh-password
The SSH password used to authenticate with **--dest-hsm-arn**.
Required: One of **--ssh-key-filename** or **--ssh-password**
This can be specified in the **ssh_password** setting in **--conf_file**.

--ssh-key-filename

The file that contains the private SSH key used to authenticate with **--dest-hsm-arn**. The public key was installed on the HSM appliance when it was provisioned.

Required: One of **--ssh-key-filename** or **--ssh-password**

This can be specified in the **ssh_key_filename** setting in **--conf_file**.

--ssh-key-passphrase

The passphrase to unlock the **--ssh-key-filename** private key file.

Required: If **--ssh-key-filename** is used.

This can be specified in the **ssh_key_passphrase** setting in **--conf_file**.

--aws-region

The region identifier, such as `us-east-1`.

Required: Yes.

This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id

Your access key ID. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key

Your secret access key. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host

Overrides the AWS CloudHSM service host.

Required: No.

This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port

Overrides the AWS CloudHSM service port.

Required: No.

This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file

The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).

Required: No.

--quiet

Quiet output. Only errors are reported.

Required: No.

--verbose

Verbose output.

Required: No.

--help

Displays help for the command.

Required: No.

Output

The status of the operation.

```
cloudhsmcli.hsm_cloner: Backing up existing config files
```

```
cloudhsmcli.hsm_cloner: Collecting information about the HSMs
cloudhsmcli.hsm_cloner: Creating partitions on the destination HSM
Please provide the password for <partition1>:
Please provide the cloning domain for <partition1>:
cloudhsmcli.hsm_cloner: A partition was created: <partition1_label>
 (<dest_partition1_ID>)
Please provide the password for <partition2>:
Please provide the cloning domain for <partition2>:
cloudhsmcli.hsm_cloner: A partition was created: <partition2_label>
 (<dest_partition2_ID>)
Please provide the password for <partition3>:
Please provide the cloning domain for <partition3>:
cloudhsmcli.hsm_cloner: A partition was created: <partition3_label>
 (<dest_partition3_ID>)
cloudhsmcli.hsm_cloner: Setting up a cloning environment
cloudhsmcli.hsm_cloner: Replicating keys from the source HSM
cloudhsmcli.hsm_cloner: Replicated keys from the
 partition <src_partition1_ID> to <dest_partition1_ID> successfully
cloudhsmcli.hsm_cloner: Replicated keys from the
 partition <src_partition2_ID> to <dest_partition2_ID> successfully
cloudhsmcli.hsm_cloner: Replicated keys from the
 partition <src_partition3_ID> to <dest_partition3_ID> successfully
cloudhsmcli.hsm_cloner: Cleaning up the cloning environment
cloudhsmcli.hsm_cloner: Restoring existing config files
cloudhsmcli.hsm_cloner: Cloning the client/partition configuration on the HSM
{
   "Status": "Completed cloning the HSM <src-hsm-arn> to the HSM <dest-hsm-
arn>"
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- Replace a Failed HSM (p. 54)

create-client

Description

Creates an HSM client.

Usage

```
cloudhsm create-client
   --certificate-file <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--certificate-file
The file that contains the base64-encoded X.509 v3 PEM certificate to be installed on the HSMs used by this client. For more information, see Client Certificates (p. 20).
Required: Yes
This can be specified in the **certificate_file** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

The ARN of the client.

```
{
    "ClientArn": "<client_arn>",
    "RequestId": "<request_id>"
}
```

AWS CloudHSM command line tools display errors on stderr.

create-hapg

Description

Creates an empty high-availability (HA) partition group. A HA partition group is a group of partitions that spans multiple physical HSMs.

You add HSMs and partitions to the HA partition group with the **add-hsm-to-hapg (p. 57)** command.

Usage

```
cloudhsm create-hapg
   --group-label <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--group-label
The label of the new HA partition group.
Required: Yes
This can be specified in the **group_label** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains the ARN of the HA partition group.

```
{
    "HapgArn": "<hapg_arn>",
    "RequestId": "<request_id>"
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- Create the HA Partition Group (p. 38)

create-hsm

Description

Creates an uninitialized HSM instance.

There is an upfront fee charged for each HSM instance you create with the create-hsm (p. 67) command. If you accidentally provision an HSM and want to request a refund, please delete the instance using the delete-hsm (p. 72) command, go to the AWS Support Center, create a new case, and select **Account and Billing Support**.

Important
It can take up to 20 minutes to create and provision an HSM. You can monitor the status of the HSM with the describe-hsm (p. 78) command. The HSM is ready to be initialized when the status changes to `RUNNING`.

Usage

```
cloudhsm create-hsm
   --subnet-id <value>
   --ssh-public-key-file <value>
   --iam-role-arn <value>
   [--hsm-ip <value>]
   [--external-id <value>]
```

```
    [--syslog-ip <value>]
    --aws-region <value>
    --aws-access-key-id <value>
    --aws-secret-access-key <value>
    [--aws-host <value>]
    [--aws-port <value>]
    [--conf_file <value>]
    [--quiet]
    [--verbose]
    [--help]
```

Arguments

--subnet-id

The identifier of the subnet in your VPC in which to place the HSM.

Required: Yes

This can be specified in the **subnet_id** setting in **--conf_file**.

--ssh-public-key-file

The file that contains the SSH public key to install on the HSM. This is used to log in to the manager account on the HSM.

Required: Yes

This can be specified in the **ssh_public_key_file** setting in **--conf_file**.

--iam-role-arn

The ARN of an IAM role to enable the AWS CloudHSM service to allocate an ENI on your behalf.

Required: Yes

This can be specified in the **iam_role_arn** setting in **--conf_file**.

--hsm-ip

The desired IP address of the HSM. This IP address will be assigned to the ENI that is attached to the HSM.

If an IP address is not specified, an IP address is randomly chosen from the CIDR range of the subnet.

Required: No

This can be specified in the **hsm_ip** setting in **--conf_file**.

--external-id

The external ID from **--iam-role-arn**, if present.

Required: No

This can be specified in the **external_id** setting in **--conf_file**.

--syslog-ip

The new IP address of the syslog monitoring server. The AWS CloudHSM service only supports one syslog monitoring server.

> **Note**
> This option is only available in CLI version 2.2015.01.22.17.26.52 and later. For more information, see version (p. 94).

Required: No

This can be specified in the **syslog_ip** setting in **--conf_file**.

--aws-region

The region identifier, such as `us-east-1`.

Required: Yes.

This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id

Your access key ID. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key

Your secret access key. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host

Overrides the AWS CloudHSM service host.

Required: No.

This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port

Overrides the AWS CloudHSM service port.

Required: No.

This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file

The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).

Required: No.

--quiet

Quiet output. Only errors are reported.

Required: No.

--verbose

Verbose output.

Required: No.

--help

Displays help for the command.

Required: No.

Output

The ARN of the HSM. Because this command causes an upfront fee to be charged to your account, you are prompted to verify the operation before the HSM is created.

```
{
    "HsmArn": "<hsm_arn>",
    "RequestId": "<request_id>"
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- Provisioning Your HSMs (p. 23)

delete-client

Description

Deletes an HSM client.

Usage

```
cloudhsm delete-client
   --client-arn <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--client-arn
The ARN that identifies the client to delete.
Required: Yes
This can be specified in the **client_arn** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

Output

A JSON block that contains the status of the operation.

```
{
   "RequestId": <request_id>,
   "Status": <status>
}
```

AWS CloudHSM command line tools display errors on stderr.

delete-hapg

Description

Deletes a high-availability (HA) partition group. The partitions that make up the HA partition group, as well as the key material they contain, are not deleted by this command.

Usage

```
cloudhsm delete-hapg
   --hapg-arn <value>
   [--force]
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--hapg-arn
The ARN that identifies the HA partition group to delete.
Required: Yes
This can be specified in the **hapg_arn** setting in **--conf_file**.

--force
Do not display the safety check messages.
Required: No
This can be specified in the **force** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.

This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains the status of the operation.

```
{
   "Status": <status>
}
```

AWS CloudHSM command line tools display errors on stderr.

delete-hsm

Description

De-provisions an HSM.

The HSM must be zeroized prior to calling this command. For more information, see How do I zeroize my HSM (p. 54).

Usage

```
cloudhsm delete-hsm
   --hsm-arn <value>
   [--force]
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--hsm-arn
The ARN that identifies the HSM to delete.

Required: Yes

This can be specified in the **hsm_arn** setting in **--conf_file**.

--force

Do not display the safety check messages.

Required: No

This can be specified in the **force** setting in **--conf_file**.

--aws-region

The region identifier, such as `us-east-1`.

Required: Yes.

This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id

Your access key ID. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key

Your secret access key. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host

Overrides the AWS CloudHSM service host.

Required: No.

This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port

Overrides the AWS CloudHSM service port.

Required: No.

This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file

The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).

Required: No.

--quiet

Quiet output. Only errors are reported.

Required: No.

--verbose

Verbose output.

Required: No.

--help

Displays help for the command.

Required: No.

Output

A JSON block that contains the status of the operation.

```
{
   "Status": "<status>"
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- create-hsm (p. 67)

deregister-client-from-hapg

Description

Removes an HSM client from a high-availability (HA) partition group.

This command requires SSH connectivity with all HSMs in the high-availability partition group. For more information, see SSH Connections (p. 19).

Usage

```
cloudhsm deregister-client-from-hapg
   --client-arn <value>
   --hapg-arn <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--client-arn
The ARN that identifies the client.
Required: Yes
This can be specified in the **client_arn** setting in **--conf_file**.

--hapg-arn
The ARN of the HA partition group.
Required: Yes
This can be specified in the **hapg_arn** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains the status of the operation.

```
{
   "Status": <status>
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- register-client-to-hapg (p. 90)

describe-client

Description

Retrieves information about an HSM client.

Usage

```
cloudhsm describe-client
   --client-arn <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
```

```
    [--aws-host <value>]
    [--aws-port <value>]
    [--conf_file <value>]
    [--quiet]
    [--verbose]
    [--help]
```

Arguments

--client-arn
The ARN that identifies the client to retrieve the information for.
Required: Yes
This can be specified in the **client_arn** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains information about the specified client.

```
{
    "Certificate": "<certificate>",
    "CertificateFingerprint": "<certificate_fingerprint>",
    "ClientArn": "<client_arn>",
    "Label": "<label>",
    "LastModifiedTimestamp": "<last_modified>"
}
```

AWS CloudHSM command line tools display errors on stderr.

describe-hapg

Description

Retrieves information about a high-availability (HA) partition group.

Usage

```
cloudhsm describe-hapg
   --hapg-arn <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--hapg-arn
The ARN that identifies the HA partition group to get information for.
Required: Yes
This can be specified in the **hapg_arn** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains information about the specified HA partition group.

```
{
    "HapgArn": "<hapg_arn>",
    "HapgSerial": "<hapg_serial>",
    "HsmsLastActionFailed": [],
    "HsmsPendingDeletion": [],
    "HsmsPendingRegistration": [],
    "Label": "<hapg_label>",
    "LastModifiedTimestamp": "<last_modified>",
    "PartitionSerialList": [
        "<partition_serial_1>",
        "<partition_serial_2>"
    ],
    "State": "<state>"
}
```

AWS CloudHSM command line tools display errors on stderr.

describe-hsm

Description

Retrieves information about an HSM.

Usage

```
cloudhsm describe-hsm
   --hsm-arn <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
```

```
    [--aws-host <value>]
    [--aws-port <value>]
    [--conf_file <value>]
    [--quiet]
    [--verbose]
    [--help]
```

Arguments

--hsm-arn
The ARN that identifies the HSM to get information for.
Required: Yes
This can be specified in the **hsm_arn** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains information about the specified HSM.

```
{
    "EniId": "<eni_id>",
    "EniIp": "<eni_ip>",
    "HsmArn": "<hsm_arn>",
    "IamRoleArn": "<iam_role_arn>",
    "SerialNumber": "<serial_number>",
    "SoftwareVersion": "<version>",
    "SshPublicKey": "<public_key_contents>",
    "Status": "<status>",
    "SubnetId": "<subnet_id>",
    "SubscriptionStartDate": "<start_date>",
    "SubscriptionType": "<subscription_type>",
    "VendorName": "<vendor>"
}
```

AWS CloudHSM command line tools display errors on stderr.

get-client-configuration

Description

Obtains the configuration file and server certificates for a client. This command must be run on every client assigned to the specified high-availability (HA) partition group.

You must re-issue this command after you make any changes to the HA partition group, such as adding or removing an HSM.

This command requires write access to certain files and directories on the local system. For more information, see Setting the Necessary File and Directory Ownership (p. 18).

This command requires SSH connectivity with all HSMs in the HA partition groups. For more information, see SSH Connections (p. 19).

Usage

```
cloudhsm get-client-configuration
   --client-arn <value>
   --hapg-arns <value1 value2 ...>
   [--cert-directory <value>]
   [--config-directory <value>]
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--client-arn
The ARN that identifies the client to retrieve the information for.
Required: Yes

This can be specified in the **client_arn** setting in **--conf_file**.

--hapg-arns
A list of ARNs that identify the HA partition groups that are associated with the client. Each ARN in the list is separated by a space.

Required: Yes

This can be specified in the **hapg_arns** setting in **--conf_file**.

--cert-directory
The local directory where the server certificate will be written. If this parameter is not specified, the server certificate is written to the current working directory. The server certificate must be placed in the server certificate directory. The location of the server certificate directory varies depending on the version of the LunaSA client software installed.

Client software version 5.1
`/usr/lunasa/cert/server`

Client software version 5.4
`/usr/safenet/lunaclient/cert/server`

Required: No

This can be specified in the **cert_directory** setting in **--conf_file**.

--config-directory
The local directory where the Chrystoki.conf file will be written. If this parameter is not specified, the configuration file is written to the current working directory. The Chrystoki.conf file must be placed in the `/etc/` directory.

Required: No

This can be specified in the **config_directory** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.

Required: Yes.

This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.

Required: No.

This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.

Required: No.

This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).

Required: No.

--quiet
Quiet output. Only errors are reported.

Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

Information about where the configuration file and certificates were written.

```
The configuration file has been copied to <config-directory>
The server certificate has been copied to <cert-directory>
```

AWS CloudHSM command line tools display errors on stderr.

initialize-hsm

Description

Performs the initial configuration of an HSM. You must have already allocated the HSM resource and have the resulting Amazon resource name (ARN) that identifies the HSM. You can use the create-hsm (p. 67) command to create an HSM instance. You can use the list-hsms (p. 87) command to obtain a list of the HSM ARNs.

The HSM must be zeroized prior to calling this command. For more information, see How do I zeroize my HSM (p. 54).

This command requires SSH connectivity with the HSM. For more information, see SSH Connections (p. 19).

Note
Initializing an HSM creates the HSM security officer account (also known as the administrator) and requires that a password be created and assigned to that account. Make a note of the password on your Password Worksheet (p. 33) and do not lose it. We recommend that you print out a copy of the Password Worksheet (p. 33), use it to record your AWS CloudHSM passwords, and store it in a secure place. We also recommended that you store at least one copy of this worksheet in secure off-site storage. AWS does not have the ability to recover your key material from an HSM for which you do not have the proper HSM security officer credentials.

Usage

```
cloudhsm initialize-hsm
   --hsm-arn <value>
   --label <value>
   --so-password <value>
   --cloning-domain <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
```

```
   [--verbose]
   [--help]
```

Options

--hsm-arn
The ARN that identifies the HSM to initialize.
Required: Yes.
This can be specified in the **hsm_arn** setting in **--conf_file**.

--label
The label for the HSM. Use only letters and numbers. Special characters are not allowed.
Required: Yes.
This can be specified in the **label** setting in **--conf_file**.

--so-password
The HSM security officer password.
Required: Yes.
This can be specified in the **so_password** setting in **--conf_file**.

--cloning-domain
The cloning domain to set for the HSM.
Required: Yes.
This can be specified in the **cloning_domain** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.

Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains the status of the operation.

```
{
    "Status": "Initialization of the HSM successful"
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- Configuring Your HSM (p. 24)

list-clients

Description

Retrieves the identifiers of the clients belonging to the current customer.

Usage

```
cloudhsm list-clients
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host

Overrides the AWS CloudHSM service host.

Required: No.

This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port

Overrides the AWS CloudHSM service port.

Required: No.

This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file

The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).

Required: No.

--quiet

Quiet output. Only errors are reported.

Required: No.

--verbose

Verbose output.

Required: No.

--help

Displays help for the command.

Required: No.

Output

A JSON block that contains the list of ARNs that identify the clients.

```
{
    "ClientList": [
        "<client1_arn>",
        "<client2_arn>"
    ]
}
```

AWS CloudHSM command line tools display errors on stderr.

list-hapgs

Description

Retrieves the identifiers of all of the high-availability (HA) partition groups belonging to the current customer.

Usage

```
cloudhsm list-hapgs
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
```

```
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains the list of ARNs that identify the high-availability partition groups.

```
{
    "HapgList": [
        "<hapg1_arn>",
```

```
        "<hapg2_arn>"
    ]
}
```

AWS CloudHSM command line tools display errors on stderr.

list-hsms

Description

Retrieves the identifiers of all of the HSMs provisioned for the current customer.

Usage

```
cloudhsm list-hsms
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains the list of ARNs that identify the HSMs.

```
{
    "HsmList": [
        "<hsm1_arn>",
        "<hsm2_arn>"
    ]
}
```

AWS CloudHSM command line tools display errors on stderr.

modify-hsm

Description

Modifies an existing HSM instance.

> **Important**
> This command can result in the HSM being offline for up to 15 minutes while the AWS CloudHSM service is reconfigured. If you are modifying a production HSM, you should ensure that your AWS CloudHSM service is configured for high availability, and consider executing this command during a maintenance window.

Usage

```
cloudhsm modify-hsm
   --hsm-arn <value>
   [--subnet-id <value>]
   [--iam-role-arn <value>]
   [--hsm-ip <value>]
   [--external-id <value>]
   [--syslog-ip <value>]
   [--force]
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
```

```
[--help]
```

Arguments

--hsm-arn

The ARN that identifies the HSM to modify.

Required: Yes

This can be specified in the **hsm_arn** setting in **--conf_file**.

--subnet-id

The identifier of the new subnet in your VPC in which to place the HSM. The new subnet must be in the same Availability Zone as the current subnet.

Required: No

This can be specified in the **subnet_id** setting in **--conf_file**.

--iam-role-arn

The ARN of the new IAM role that enables the AWS CloudHSM service to allocate an ENI on your behalf.

Required: No

This can be specified in the **iam_role_arn** setting in **--conf_file**.

--hsm-ip

The new IP address of the HSM. This IP address is assigned to the ENI that is attached to the HSM. The subnet that the new IP address belongs to must be in the same Availability Zone as the subnet of the previous IP address.

Required: No

This can be specified in the **hsm_ip** setting in **--conf_file**.

--external-id

The new external ID from **--iam-role-arn**.

Required: No

This can be specified in the **external_id** setting in **--conf_file**.

--syslog-ip

The new IP address of the syslog monitoring server. The AWS CloudHSM service only supports one syslog monitoring server.

Required: No

This can be specified in the **syslog_ip** setting in **--conf_file**.

--force

Do not display the safety check messages.

Required: No

This can be specified in the **force** setting in **--conf_file**.

--aws-region

The region identifier, such as `us-east-1`.

Required: Yes.

This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id

Your access key ID. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key

Your secret access key. For more information, see Authentication (p. 18).

Required: Yes.

This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

The ARN of the HSM.

```
{
    "HsmArn": "<hsm_arn>",
    "RequestId": "<request_id>"
}
```

AWS CloudHSM command line tools display errors on stderr.

Version

This command is only available in CLI version 2.2015.01.22.17.26.52 and later. For more information, see version (p. 94).

Related Topics

- create-hsm (p. 67)

register-client-to-hapg

Description

Adds a an HSM client to a high-availability (HA) partition group.

You must re-issue this command after you make any changes to the HA partition group, such as adding an HSM.

This command requires SSH connectivity with all HSMs in the high-availability partition group. For more information, see SSH Connections (p. 19).

Usage

```
cloudhsm register-client-to-hapg
   --client-arn <value>
   --hapg-arn <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
   [--help]
```

Arguments

--client-arn
The ARN that identifies the client.
Required: Yes.
This can be specified in the **client_arn** setting in **--conf_file**.

--hapg-arn
The ARN that identifies the HA partition group.
Required: Yes
This can be specified in the **hapg_arn** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).

Required: No.

--quiet

Quiet output. Only errors are reported.

Required: No.

--verbose

Verbose output.

Required: No.

--help

Displays help for the command.

Required: No.

Output

A JSON block that contains the status of the operation.

```
{
   "Status": <status>
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- deregister-client-from-hapg (p. 74)

remove-hsm-from-hapg

Description

Removes an HSM from a high-availability (HA) partition group. The partition and key material corresponding to the HA partition group is removed from the HSM.

This command can be used to remove a failed HSM from its HA partition group.

This command requires SSH connectivity with the HSM. For more information, see SSH Connections (p. 19).

Usage

```
cloudhsm remove-hsm-from-hapg
   --hsm-arn <value>
   --hapg-arn <value>
   --so-password <value>
   --aws-region <value>
   --aws-access-key-id <value>
   --aws-secret-access-key <value>
   [--aws-host <value>]
   [--aws-port <value>]
   [--conf_file <value>]
   [--quiet]
   [--verbose]
```

```
[--help]
```

Arguments

--hsm-arn
The ARN that identifies the HSM to remove.
Required: Yes

--hapg-arn
The ARN that identifies the HA partition group to remove the HSM from.
Required: Yes.
This can be specified in the **hapg_arn** setting in **--conf_file**.

--so-password
The HSM security officer password.
Required: Yes.
This can be specified in the **so_password** setting in **--conf_file**.

--aws-region
The region identifier, such as `us-east-1`.
Required: Yes.
This can be specified in the **aws_region** setting in **--conf_file**.

--aws-access-key-id
Your access key ID. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_access_key_id** setting in **--conf_file**.

--aws-secret-access-key
Your secret access key. For more information, see Authentication (p. 18).
Required: Yes.
This can be specified in the **aws_secret_access_key** setting in **--conf_file**.

--aws-host
Overrides the AWS CloudHSM service host.
Required: No.
This can be specified in the **aws_host** setting in **--conf_file**.

--aws-port
Overrides the AWS CloudHSM service port.
Required: No.
This can be specified in the **aws_port** setting in **--conf_file**.

--conf_file
The path and file name of the configuration file to use. For more information, see Configuration Files (p. 19).
Required: No.

--quiet
Quiet output. Only errors are reported.
Required: No.

--verbose
Verbose output.
Required: No.

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains the status of the operation.

```
{
   "Status": "<status>"
}
```

AWS CloudHSM command line tools display errors on stderr.

Related Topics

- add-hsm-to-hapg (p. 57)

version

Description

Retrieves the version information for the AWS CloudHSM CLI tools.

Usage

```
cloudhsm version [--help]
```

Arguments

--help
Displays help for the command.
Required: No.

Output

A JSON block that contains the version information.

```
{
    "Version": "<version>"
}
```

AWS CloudHSM command line tools display errors on stderr.

Troubleshooting

The following sections show some of the more common errors you may encounter when using the AWS CloudHSM CLI.

Topics

- RuntimeError: Luna is requesting a password. (p. 95)
- The delete-hsm command appears to succeed, but the HSM is not deleted. (p. 95)

RuntimeError: Luna is requesting a password.

When you use certain commands, you get the following error message:

```
RuntimeError: Luna is requesting a password. This indicates that there is
no persistent SSH connection to the HSM. Consult the CloudHSM CLI docs for
instructions on how to set up a persistent connection.
```

This error occurs when you use a command that requires a persistent SSH connection with an HSM, such as initialize-hsm (p. 82). For more information about persistent SSH connections, see SSH Connections (p. 19).

The delete-hsm command appears to succeed, but the HSM is not deleted.

This error can occur when you use the delete-hsm (p. 72) command to delete an HSM that is not zeroized. To determine if this is case, after issuing the delete-hsm (p. 72) command, use the describe-hsm (p. 78) command to get information about the HSM. If the `StatusDetails` field contains a message such as `"The CloudHSM must be zeroized before it can be deleted."`, then you will need to zeroize the HSM. For more information about how to zeroize your HSM, see How do I zeroize my HSM (p. 54).

AWS CloudHSM Limits

The following list contains the limits for the AWS CloudHSM service. Unless indicated otherwise, these limits are per region and per AWS account.

- HSM appliances: 3
- High-availability partition groups: 20
- Clients: 800

To request an increase to these limits, use the service limit increase form in the AWS Support Center.

Appendices

The following are the appendices for the AWS CloudHSM User Guide. They provide additional information for some of the AWS CloudHSM operation and usage.

Topics

Getting Started Manually

The following topics explain how to provision, initialize, and use an HSM without using the CLI.

Topics

Manually Provisioning an HSM

You need the following information to provision your HSM.

- The identifiers of the private subnets to provision the HSMs in.
- The ARN of the AWS CloudHSM IAM role.
- Your SSH public key. For more information, see Generating an SSH Key (p. 13).

> **Important**
> There is an upfront fee charged for each HSM instance you provision. If you accidentally provision an HSM and want to request a refund, please delete the instance, go to the AWS Support Center, create a new case, and select **Account and Billing Support**

To try the AWS CloudHSM service for free, you can request a two week trial. For more information about the free trial, go to Free Trial.

Manually Initialize an HSM

To manually initialize and configure an HSM

Use the following procedures to initialize your HSM. Repeat as needed for each HSM.

1. If needed, copy your SSH private key file to the control instance. This is the private portion of the key that you used to provision the HSM. For more information, see Copying the Private Key (p. 15).
2. From the control instance, connect to your HSM appliance over SSH. *<private_key_file>* is the private portion of the SSH key you provided when your HSM was provisioned.

```
$ ssh -i <private_key_file> manager@<hsm_ip_address>
```

3. (Optional) Set a password for the manager by executing the following command. This step is optional. You can continue to use the SSH key pair to connect to the HSM over SSH if you desire.

```
lunash:> user password manager
```

   You are prompted to enter the new password twice. Note the new manager password on your Password Worksheet (p. 33).
4. Check the time zone, date, and time on the HSM with the **status date** command.

```
lunash:> status date

Fri Feb  7 20:09:20 UTC 2014

Command Result : 0 (Success)
```

   If the time zone is not correct, set the time zone with the **sysconf timezone set** command. If the date and/or time are not correct, set them with the **sysconf time** command. For more information, go to Set System Date and Time in the SafeNet Luna SA documentation.

   Note
   AWS configures the time of each HSM to use the UTC time zone. This is also the default setting for Amazon Linux AMIs. Only change the time zone if your HSM client uses a different time zone than UTC.
   If you change the time zone, you must change it before setting the system date and time; otherwise, the time zone change adjusts the time you just set.
5. Initialize the HSM by executing the following:

```
lunash:> hsm init -label <hsm_label>
```

   The name *<hsm_label>* must be a unique name without spaces or special characters.

   Note
   If you plan to use high-availability and load balancing among multiple HSM appliances, as recommended by AWS, see High Availability and Load Balancing (p. 35) for additional instructions.

   Initializing an HSM permanently deletes the keys and entire cryptographic domain on the HSM. After initializing the HSM, any previously existing keys are destroyed.

Initializing an HSM sets the password for the HSM security officer account (also known as the administrator). Record the security officer password on your Password Worksheet (p. 33) and do not lose it. We recommend that you print out a copy of the Password Worksheet (p. 33), use it to record your AWS CloudHSM passwords, and store it in a secure place. We also recommended that you store at least one copy of this worksheet in secure off-site storage. AWS does not have the ability to recover your key material from an HSM for which you do not have the proper HSM security officer credentials.

6. Create a key pair for the HSM server. This generates a certificate from the public key.

```
lunash:> sysconf regenCert
```

7. Make an association between the HSM appliance and an NTLS interface by executing the following:

```
lunash:> ntls bind eth0
```

8. Execute the following commands to log in to the HSM using the HSM administrator password, and then create a partition:

```
lunash:> hsm login

lunash:> partition create -partition <partition_name>
```

The name *<partition_name>* must be a unique name without spaces or special characters.

9. When prompted, type **proceed**.
10. Supply the new partition password when prompted. Record the partition name and password on your Password Worksheet (p. 33), as it is used in the following situations:

- To authenticate the administrator performing partition management tasks via the Luna shell.
- To authenticate client applications that want to use the HSM.

High-Availability

The following topics discuss implementing high-availability for your HSMs without using the CLI.

Topics

Configure High-Availability

To set up high availability (HA) and load balancing for your HSMs one HSM at a time, complete the following procedure.

Configure HA redundancy and load balancing

1. Set up the network that contains the HSMs that will be used in the HA group.
2. From your control instance, connect to your HSM over SSH. *<private_key_file>* is the private portion of the SSH key you provided when your HSM was provisioned.

```
$ ssh -i <private_key_file> manager@<hsm_ip_address>
```

3. View the policy settings needed for the HSM by issuing the **hsm showPolicies** command.

```
lunash:> hsm showPolicies
HSM Label: <hsm_label>

Serial #: <hsm_serial>

Firmware: 6.2.1

The following capabilities describe this HSM, and cannot be altered
except via firmware or capability updates.

Description                 Value
===========                 =====

Enable cloning              Allowed
.
.
.
Enable network replication  Allowed
.
.
.

The following policies describe the current configuration of
this HSM and may by changed by the HSM Administrator.

Changing policies marked "destructive" will zeroize (erase
completely) the entire HSM.
Description                  Value    Code    Destructive
===========                  =====    ====    ===========
.
Allow cloning                  On      7      Yes
.
.
Allow network replication      On      16     No
.
.
.

Command Result : 0 (Success)
```

Make note of the following policy values:

- Enable cloning
- Enable network replication
- Allow cloning
- Allow network replication

If any of these policies are not set to **Allowed**, change them with the **hsm changePolicy** command.

```
lunash:> hsm changePolicy -policy <policy_code> -value <policy_value>
```

Note
Cloning to a hardware token is the backup method for which your HSMs are configured. All HSMs in an HA group must use the same backup method.

4. Initialize your HSMs into a common cloning domain. For password-authenticated appliances, they must share the same cloning domain.

   Warning
   Initializing an HSM permanently deletes the keys and entire cryptographic domain on the HSM. After initializing the HSM, any previously existing keys are destroyed.

   Note

   - If you have already configured your HSM appliance in Configuring Your HSM Client (p. 27), the following steps help you reconfigure your HSM appliance for HA.
   - Three of the values are required, but the only one that you should type at the command line is a label for the HSM (`-label`). Typing the password and the cloning domain at the command line makes them visible to anyone who can see the computer screen, or to anyone who later scrolls back in your console or ssh session buffer. If you omit the password and the cloning domain, the Luna shell prompts you for them, and hides your input with ************ characters. This is preferable from a security standpoint. Additionally, you are prompted to re-enter each string, thus helping to ensure that the string you type is the one you meant to type.

```
lunash:> hsm -init -label <hsm_label>
> Please enter a password for the security officer
> ********
Please re-enter password to confirm:
> ********
Please enter the cloning domain to use for initializing this
HSM (press <enter> to use the default domain):
> ********
Please re-enter domain to confirm:
> ********
CAUTION:  Are you sure you wish to re-initialize this HSM?
All partitions and data will be erased.
Type 'proceed' to initialize the HSM, or 'quit' to quit now.
> proceed
'hsm - init' successful.
```

5. On each HSM, perform the following steps:

   a. Log into the HSM as the HSM administrator (Security Officer).

```
lunash:> hsm login


  Please enter the HSM Administrators' password:
  > ***********

'hsm login' successful.


Command Result : 0 (Success)
```

b. Create a partition. When prompted, type **proceed**, and enter the partition password. The partition password and cloning domain must be the same for all partitions that will be part of the same HA group.

```
lunash:> partition create -partition <partition_name> -
domain <cloning_domain>

Please ensure that you have purchased licenses for at least this
 number of partitions: 3

If you are sure to continue then type 'proceed', otherwise type 'quit'

> proceed
Proceeding...

  Please enter a password for the partition:
  > **********

  Please re-enter password to confirm:
  > **********

'partition create' successful.


Command Result : 0 (Success)
```

<partition_name> should be a unique name without spaces or special characters.

c. Record the partition serial numbers and passwords, and store this information in a secure place.

```
lunash:> partition show

   Partition SN:                            <partition1_serial>
   Partition Name:                          <partition1_name>
   Partition Owner Locked Out:              no
   Partition Owner PIN To Be Changed:       no
   Partition Owner Login Attempts Left:     10 before Owner is Locked
 Out
   Legacy Domain Has Been Set:              no
   Partition Storage Information (Bytes):   Total=102701, Used=0,
 Free=102701
   Partition Object Count:                  0


   Partition SN:                            <partition2_serial>
   Partition Name:                          <partition2_name>
   Partition Owner Locked Out:              no
   Partition Owner PIN To Be Changed:       no
   Partition Owner Login Attempts Left:     10 before Owner is Locked
 Out
   Legacy Domain Has Been Set:              no
   Partition Storage Information (Bytes):   Total=102701, Used=0,
 Free=102701
   Partition Object Count:                  0

Command Result : 0 (Success)
```

d. Proceed with a normal client setup as described in Configuring Your HSM Client (p. 27).

e. Register your client computer with each partition that will be part of the HA group. On each HSM, assign the partition to its respective client. Repeat for each HSM in the HA group.

```
lunash:> client assignPartition -client <client_name> -
partition <partition1_name>
lunash:> client assignPartition -client <client_name> -
partition <partition2_name>
```

6. On the client, create a new HA group with the **vtl haAdmin newGroup** command. This group uses partition1 as the primary partition.

   Important
   On Windows clients, you must execute the next command as an administrator. To do this, right-click the cmd.exe window and select **Run as Administrator**.

```
>vtl haAdmin newGroup -label <partition_group_label> -
serialNum <partition1_serial> -password <partition1_password>
```

```
New group with label "<partition_group_label>" created at group
 number <partition_group_serial>.
Group configuration is:
  HA Group Label: <partition_group_label>
  HA Group Number:  <partition_group_serial>
  HA Group Slot #: <slot_number>
  Synchronization: enabled
  Group Members:  <partition1_serial>
  Standby members:  <none>
  In Sync: yes
```

When you create a new HA group, the `vtl` utility create the serial number for the group.

7. Your Chrystoki.conf (Linux/UNIX)/crystoki.ini (Windows) file should now have a new section:

```
VirtualToken = {
VirtualToken00Members = <partition1_serial>;
VirtualToken00SN = <partition_group_serial>;
VirtualToken00Label = <partition_group_label>;
}
```

   Important
   Do not alter the Chrystoki.conf/crystoki.ini file.

8. Add another member to the HA group (Partition2 on the second appliance) with the **vtl haAdmin addMember** command.

   Important
   On Windows clients, you must execute the next command as an administrator. To do this, right-click the cmd.exe window and select **Run as Administrator**.

```
>vtl haAdmin addMember -group <partition_group_serial> -
serialNum <partition2_serial> -password <partition2_password>
```

```
Member <partition2_serial> successfully added to
 group <partition_group_serial>.
New group configuration is:
```

```
  HA Group Label: <partition_group_label>
  HA Group Number:  <partition_group_serial>
  HA Group Slot #: <slot_number>
  Synchronization: enabled
  Group Members:  <partition1_serial>, <partition2_serial>
  Standby members:  <none>
  In Sync: yes

Please use the command 'vtl haAdmin -synchronize' when you are ready to
 replicate data
between all members of the HA group. (If you have additional members to
 add, you may
wish to wait until you have added them before synchronizing to save time
 by avoiding
multiple synchronizations.)
```

9. Verify your setup using the following command, then point your client application at the HSM, referring to that HSM by the HA group label that you assigned.

```
>vtl haAdmin show
```

High Availability Failover and Automatic Recovery

The following instructions use the **configurator** and **vtl** applications, which are part of the Luna SA client tools. The location of these applications varies depending on the client operating system. You either need to include this path in the command, or add it to the PATH environment variable.

Linux

```
/usr/safenet/lunaclient/bin/
```

Windows

```
%ProgramFiles%\SafeNet\LunaClient\bin\
```

Configuring High Availability Failover

AWS and SafeNet recommend keeping the default 20-second failover timeout. This is configurable by executing the following command:

```
>configurator setValue -s "LunaSA Client" -e ReceiveTimeout -v <milliseconds>
```

Enabling Automatic Recovery

Automatic recovery (autoRecovery) is disabled by default.

To enable autoRecovery

- To enable autoRecovery, execute the following command:

```
>vtl haAdmin -autoRecovery -retry <count>
```

Configuring the Retry Interval

To configure the retry interval

- To configure the retry interval, execute the following command:

```
>vtl haAdmin -autoRecovery -interval <seconds>
```

Recovering an HSM

This section explains how to recover an HSM in an HA group after the client loses connectivity to one of the HSMs in the group. If network connectivity is lost, the HSM client permanently stops trying to connect to the HSM after the retry period is exceeded. The retry period is `number-of-retries * retry-interval`, where the default/recommended configuration is to retry 10 times with an interval of 60 seconds, for a total of 10 minutes. After the retry period is exceeded, the HSM client removes the disconnected HSM from the HA group, and it must be manually recovered. Follow the instructions below to recover an HSM.

To recover an HSM, issue the **vtl haAdmin recover** command from the client that has lost connectivity to the HSM.

```
vtl haAdmin recover –group <group_label> -serialNum <partition_serial>
```

Important
Do not perform a manual resynchronization between the members of the HA group. For more information, see Best Practices for Loss and Recovery (p. 36).

Connecting Multiple Client Instances to AWS CloudHSM with One Certificate

When you use multiple servers with AWS CloudHSM, normally each server generates a unique certificate using that instance's IP address and registers this certificate with AWS CloudHSM; additional steps must then be taken to allow this instance access to the HSM appliance. However, you can avoid the need to create unique certificates per server by creating either an AMI with the HSM client configuration or an Amazon S3 bucket. Either of these solutions can be used with Auto Scaling groups to allow client instances to scale up and down. This allows you to have a scalable services layer that integrates with AWS CloudHSM.

Topics

Creating an AMI with the HSM Client Configuration

Create an AMI with the client configuration, and then create multiple instances from the AMI. You can use a name instead of an IP address when creating the certificate on the HSM client, and you can create multiple instances from the same AMI without re-creating or changing the certificate.

Note
If you use a name instead of an IP address when creating the certificate on the HSM client, make sure that the registered client name on the HSM appliance matches exactly.

To create an AMI with the client configuration and prepare the HSM client

1. Execute the following commands on the HSM client, where **`ClientCertName`** is the name you have chosen for the certificate on the HSM client.

```
C:\Program Files\LunaSA>vtl createCert -n ClientCertName
Private Key created and written to: C:\Program Files\LunaSA\cert\client
\ClientCertNameKey.pem
Certificate created and written to: C:\Program Files\LunaSA\cert\client
\ClientCertName.pem

C:\Program Files\LunaSA>pscp "%programfiles%\LunaSA\cert\client
\ClientCertName.pem" manager@10.0.0.23:
manager@10.0.0.23's password:

ClientCertName.pem        | 1 kB |   1.1 kB/s | ETA: 00:00:00 | 100%
```

2. Execute the following commands on the HSM, where **`ClientName`** is the name of your HSM client and **`ClientCertName`** is your certificate name.

```
[hsm6105.iad6] lunash:>c reg -c ClientName -h ClientCertName

'client register' successful.

Command Result : 0 (Success)
[hsm6105.iad6] lunash:>c l

registered client 1: ClientName
```

3. After completing the steps above, create an AMI that includes the client configuration, then create one or more Amazon EC2 instances from the AMI. Each Amazon EC2 instance can connect to the HSM appliance using the same certificate, and instances started from Auto Scaling groups can establish a secure connection to AWS CloudHSM.

   For more information about creating AMIs, see Creating Your Own AMIs in the *Amazon EC2 User Guide for Linux Instances* guide.

   For more information about creating instances from AMIs, see Launch Your Instance in the *Amazon EC2 User Guide for Linux Instances* guide.

Create an Amazon S3 Bucket and Roles

If you prefer not to create an AMI, you can create an Amazon S3 bucket with the certificates and keys in them, then create a role with an attached policy that allows read-only access to that bucket, and use the role when launching the instance for your application (including with Auto Scaling). Then you can write scripts in the instance to access the files from Amazon S3.

To create an Amazon S3 bucket and roles

1. Create an Amazon S3 bucket. For more information, see Create a Bucket in the *Amazon Simple Storage Service Getting Started Guide*.
2. Change permissions on the Amazon S3 bucket to reduce permissions to the minimum set of people necessary.
3. Upload the certificates into the Amazon S3 bucket.
4. Create a role for your application. For more information, see Creating a Role in the *IAM User Guide* guide.

5. As part of creating the role, modify the role's policy to allow read-only access to the Amazon S3 bucket; for example, `"Resource": ["arn:aws:S3:::bucket/*"]`.
6. Use the role when launching your application.
7. Write scripts on the application instance to download the certificate files from the Amazon S3 bucket.

   This allows you to update the certificates from time to time, and also does not require you to figure out how to secure your AMI to prevent credential leakage.

To learn more about using IAM roles with Amazon S3 buckets, see Using IAM roles to distribute non-AWS credentials to your EC2 instances in the AWS Security blog or Using IAM Roles for EC2 Instances with the SDK for Java in the *AWS SDK for Java Developer Guide*.

Sample Application

The SafeNet Luna products include an API that allows you to use an HSM with your application. The following are two sample applications that use an HSM, one written in C and the other in Java.

Topics

Sample Application Using C

The following procedure shows how to build a sample program that uses the SafeNet PKCS#11 library to encrypt and decrypt a string, using the HSM to perform the cryptographic operations. The sample source code is written in the C programming language.

To build the sample C application

1. Install the SafeNet client and certificates on your instance in your VPC, as described in the previous sections.
2. Download the sample source code to your instance.
3. On UNIX/Linux, do the following:

   a.
   ```
   $ mkdir Sample
   ```

   b.
   ```
   $ mv P11Sample.zip Sample
   ```

   c.
   ```
   $ cd Sample/
   ```

   d.
   ```
   $ unzip P11Sample.zip
   ```

   e.
   ```
   $ more README.txt
   ```

4. Follow the instructions in the README.txt file for installing make, gcc, setting the `SfntLibPath` environment variable, building the sample application, and running it.

Sample Application Using Java

The following instructions show how to use Luna JSP, which consists of a single JCA/JCE service provider, to build a Java-based sample application that uses SafeNet Luna products for secure cryptographic operations.

The Luna JSP comes with several sample applications that show you how to use the Luna provider. Install these sample applications with the SafeNet client software.

The sample applications include detailed comments. For more information, go to Luna JSP in the SafeNet Luna SA documentation.

If not already installed, you will need to install the Java development environment on the client instance. On an Amazon Linux instance, run the following command.

```
$ sudo yum install java-devel
```

To compile and run the Java applications

1. Create a workspaces directory and change to that folder.

   ```
   $ mkdir ~/workspaces
   ```

   ```
   $ cd ~/workspaces
   ```

2. Copy the Java sample code to your workspaces folder.

   ```
   $ cp -r /usr/safenet/lunaclient/jsp/ luna
   ```

   This copies the sample file tree into the `~/workspaces/luna` folder.

3. Change the directory to the luna sample folder.

   ```
   $ cd luna/samples
   ```

4. Update the HSM partition password in the Java sample code. Throughout the examples, the password `userpin` is used for the partition password. You need to find all instances of `userpin` in the sample code and replace it with your partition password.
5. Compile the sample code.

   ```
   $ javac -classpath .:../lib/LunaProvider.jar ./com/safenetinc/luna/sample/
   *.java
   ```

6. Add the LunaProvider to the `java.security` file.

   a. Open the file in a text editor.

      ```
      $ sudo vi $JAVA_HOME/lib/security/java.security
      ```

   b. Add the following line after the last `security.provider` entry. Replace *`<priority_order>`* with your desired priority number.

      ```
      security.provider.<priority_order>=com.safenetinc.luna.provider.LunaProvider
      ```

7. Execute the desired example. The following command executes the `KeyStoreLunaDemo` example.

```
$ java -Djava.library.path=../lib/ -classpath .:../
lib/LunaProvider.jar:../lib/libLunaAPI.so
 com.safenetinc.luna.sample.KeyStoreLunaDemo
```

For more information, go to the following SafeNet Luna SA documentation topics:

- Java Applications via Luna JSP
- Linux Installation
- Java

AWS CloudHSM Upgrade Guide

Use the information in this guide to upgrade the AWS CloudHSM Luna SA appliance and client software, and HSM firmware.

> **Note**
> The upgrade paths detailed in this guide are the only upgrades supported by AWS CloudHSM. Any software or firmware versions not documented in this guide are unsupported. If you require a different firmware or software version, please open a support case using the AWS Support Center.

Topics

Upgrade Version 5.1 to 5.3

The following procedure is used to upgrade the AWS CloudHSM Luna SA appliance and client software from version 5.1 to 5.3. After the upgrade is complete, you should use the SafeNet product documentation for Luna SA version 5.3, which can be found at http://cloudhsm-safenet-docs-5.3.s3-website-us-east-1.amazonaws.com/.

Upgrading from 5.1 to 5.3 is performed by upgrading the software in separate steps. Before proceeding, note that AWS has identified the following issues that may be encountered.

- If the name of the client used for NTLS connections has a capital T ('T') as the eighth character of the name, the client will not work after the upgrade. To avoid this, change the name of the client before performing the upgrade.
- The syslog configuration for the HSM appliance will be lost. Notify the AWS CloudHSM team by sending email to aws-cloudhsm-support@amazon.com after you perform the upgrade, and we will update the syslog configuration for you.

To upgrade to version 5.3

1. Upgrade Client Software to 5.4 (p. 110).
2. Back up all key material on the HSM to another HSM or a backup HSM (p. 44).

3. Upgrade the Luna SA appliance software from version 5.1.X to 5.1.5 (p. 111).
4. Upgrade the Luna SA appliance software from version 5.1.5 to 5.2.6 (p. 112).
5. Upgrade the Luna SA appliance software from version 5.2.6 to 5.3.5 (p. 112).

Upgrade Client Software to 5.4

Upgrade the Luna SA client software from version 5.1.1 to version 5.4. Perform the following steps on all clients that use your HSM.

To upgrade the client software

1. Stop all applications and services that are using the HSM.
2. Uninstall the existing version of the client software.
   - On Linux clients, run the `/usr/lunasa/bin/uninstall.sh` file as root. Your configuration file is preserved in `/etc/Christoki.conf.rpmsave` and your certificates are preserved in the `/usr/lunasa/cert` directory. Do not delete these files.
   - On Windows clients, uninstall the client software and any patches using the **Programs and Features** Control Panel utility. Your `chrystoki.ini` configuration file and certificates are not deleted from the `Program Files\LunaSA` directory. Do not delete these files.
3. Download the client software package at https://s3.amazonaws.com/cloudhsm-software/Luna_5_4_Client_Software.tgz. You can verify the integrity of the downloaded package with the following SHA 256 digest:

   ```
   4777ae559cfa9421735f73b4c1a2fe69b2f43d4d774f36e4050e773c23372f4c
   ```

   This digest is also available at https://s3.amazonaws.com/cloudhsm-software/Luna_5_4_Client_Software.tgz.sha256.
4. Extract the files from the package, and then install the appropriate client software for your operating system.

   #### Linux Clients

   1. For Linux clients, run the `610-012382-008_revC/linux/64/install.sh` file as root and install the **Luna SA** option. By default, this installs the client software into a different directory than where the original version was installed.
   2. Move your original certificates from `/usr/lunasa/cert` to the `/usr/safenet/lunaclient/cert` directory.
   3. Edit the following entries in your `/etc/Christoki.conf` file. Modify the paths to match the new install location. For example, if the original value for `ClientCertFile` was `/usr/lunasa/cert/client/linux_client.pem`, change this to `/usr/safenet/lunaclient/cert/client/linux_client.pem`.

      ClientCertFile
      : Set to the path of the client certificate file you restored in the previous step.

      ClientPrivKeyFile
      : Set to the path of the client private key file you restored in the previous step.

      ServerCAFile
      : Set to the path of the server CA file you restored in the previous step.

      SSLConfigFile
      : Set to the path of the openssl.cnf file in the `/usr/safenet/lunaclient/bin` directory.

   4. Update the `PATH` environment variable, if needed, to point to the `/usr/safenet/lunaclient/bin` directory.

Windows Clients

1. For Windows clients, run the `610-012382-008_revC\windows\64\LunaClient.msi` file and install the **Luna SA** option. By default, this installs the client software into a different directory than where the original version was installed. Your existing configuration file and certificates are preserved in the original directory.
2. Move your original certificates from `C:\Program Files\LunaSA\cert` to the `C:\Program Files\SafeNet\LunaClient\cert` directory.
3. Copy the following entries from your original `chrystoki.ini` file to the new `C:\Program Files\SafeNet\LunaClient\chrystoki.ini` file. Modify the paths to match the new install location. For example, if the original value for `ClientCertFile` was `C:\Program Files\LunaSA\cert\client\windows_client.pem`, change this to `C:\Program Files\SafeNet\LunaClient\cert\client\windows_client.pem`.

   ClientCertFile
   : Set to the path of the client certificate file you restored in the previous step.

   ClientPrivKeyFile
   : Set to the path of the client private key file you restored in the previous step.

   All ServerName* entries

   All ServerPort* entries

4. Update the `PATH` environment variable, if needed, to point to the `C:\Program Files\SafeNet\LunaClient\` directory.

Upgrade Appliance Software to 5.1.5

Upgrade the Luna SA appliance software from version 5.1.X to version 5.1.5. Perform the following steps on a control instance that has IP connectivity to the HSM appliance.

To upgrade the Luna SA appliance software

1. Stop all applications and services that are using the HSM.
2. Download the Luna SA appliance software upgrade package from https://s3.amazonaws.com/cloudhsm-software/630-010165-018_REVA.tar and extract the files from the archive.
3. Copy the `lunasa_update-5.1.5-2.spkg` file to the HSM appliance, where *<private_key_file>* is the private portion of the SSH key that you provided when your HSM was provisioned.

```
$ scp -i <private_key_file> lunasa_update-5.1.5-2.spkg
 manager@<hsm_ip_address>:
```

4. Connect to the HSM appliance and log in to the HSM with the following commands:

```
$ ssh -i <private_key_file> manager@<hsm_ip_address>

lunash:> hsm login
```

5. Verify and install the Luna SA appliance software update with the following commands. The value to use for *<auth_code>* is in the `lunasa_update-5.1.5-2.auth` file contained in the `630-010165-018_REVA.tar` archive.

```
lunash:> package verify lunasa_update-5.1.5-2.spkg -authcode <auth_code>
```

```
lunash:> package update lunasa_update-5.1.5-2.spkg -authcode <auth_code>
```

6. Reboot the HSM appliance with the following command:

```
lunash:> sysconf appliance reboot
```

Upgrade Appliance Software to 5.2.6

Upgrade the Luna SA appliance software from version 5.1.5 to version 5.2.6. Perform the following steps on a control instance that has IP connectivity to the HSM appliance.

To upgrade the Luna SA appliance software

1. Stop all applications and services that are using the HSM.
2. Download the Luna SA appliance software and HSM firmware upgrade package from https://s3.amazonaws.com/cloudhsm-software/630-010165-022_RevA.tar and extract the files from the archive.
3. Copy the `lunasa_update-5.2.6-1.spkg` file to the HSM appliance, where *<private_key_file>* is the private portion of the SSH key that you provided when your HSM was provisioned.

```
$ scp -i <private_key_file> lunasa_update-5.2.6-1.spkg
 manager@<hsm_ip_address>:
```

4. Connect to the HSM appliance and log in to the HSM with the following commands:

```
$ ssh -i <private_key_file> manager@<hsm_ip_address>

lunash:> hsm login
```

5. Verify and install the Luna SA appliance software update with the following commands. The value to use for *<auth_code>* is in the `lunasa_update-5.2.6-1.auth` file contained in the `630-010165-022_RevA.tar` archive.

```
lunash:> package verify lunasa_update-5.2.6-1.spkg -authcode <auth_code>

lunash:> package update lunasa_update-5.2.6-1.spkg -authcode <auth_code>
```

6. Reboot the HSM appliance with the following command:

```
lunash:> sysconf appliance reboot
```

Do not update the HSM firmware. If you require support for a specific encryption algorithm, you can optionally upgrade the firmware on your HSM to after the software upgrade has been completed. For firmware upgrade information, see Upgrade HSM Firmware (p. 114).

Upgrade Appliance Software to 5.3.5

Upgrade the Luna SA appliance software from version 5.2.6 to version 5.3.5. Perform the following steps on a control instance that has IP connectivity to the HSM appliance.

To upgrade the Luna SA appliance software

1. Stop all applications and services that are using the HSM.

2. Download the Luna SA appliance software upgrade package from https://s3.amazonaws.com/cloudhsm-software/630-010165-023_RevA.tar and extract the files from the archive.
3. Copy the `lunasa_update-5.3.5-1.spkg` file to the HSM appliance, where *<private_key_file>* is the private portion of the SSH key that you provided when your HSM was provisioned.

```
$ scp -i <private_key_file> lunasa_update-5.3.5-1.spkg
 manager@<hsm_ip_address>:
```

4. Connect to the HSM appliance and log in to the HSM with the following commands:

```
$ ssh -i <private_key_file> manager@<hsm_ip_address>

lunash:> hsm login
```

5. Verify and install the Luna SA appliance software update with the following commands. The value to use for *<auth_code>* is in the `lunasa_update-5.3.5-1.auth` file contained in the `630-010165-023_RevA.tar` archive.

```
lunash:> package verify lunasa_update-5.3.5-1.spkg -authcode <auth_code>

lunash:> package update lunasa_update-5.3.5-1.spkg -authcode <auth_code>
```

6. Reboot the HSM appliance with the following command:

```
lunash:> sysconf appliance reboot
```

Disable NTLS IP Checking

After the HSM is upgraded, you must disable NTLS IP checking to allow the HSM to operate within its VPC. To do this, run the following command from the HSM appliance shell:

```
lunash:> ntls ipcheck disable
```

Upgrade Version 5.3.X to 5.3.5

The following procedure is used to upgrade the AWS CloudHSM Luna SA appliance software from version 5.3.X to 5.3.5. LunaSA client software 5.3.0 does not require an upgrade to use LunaSA appliance software 5.3.5.

This procedure should only be performed with permission from, and in consultation with, the AWS CloudHSM team.

To upgrade in steps

1. Back up all key material on the HSM to another HSM or a backup HSM. For more information, see Backing Up and Restoring HSM Data to a Luna SA Backup HSM.
2. Notify the AWS CloudHSM support team that you are planning to upgrade.
   - If you have a premium support plan, open a support case using the AWS Support Center.
   - If you do not have a premium support plan, send email to aws-cloudhsm-support@amazon.com.
3. Upgrade the Luna SA appliance software from version 5.3.X to 5.3.5.

Upgrade the Luna SA Appliance Software

Upgrade the Luna SA appliance software to version 5.3.5. To upgrade the Luna SA appliance software, perform the following steps on a control instance that has IP connectivity to the HSM appliance.

To upgrade the Luna SA appliance software

1. Stop all applications and services that are using the HSM.
2. Download the Luna SA appliance software upgrade package from https://s3.amazonaws.com/cloudhsm-software/630-010165-023_RevA.tar and extract the files from the archive.
3. Keep the contents of the `lunasa_update-5.3.5-1.auth` file for later use.
4. Copy the `lunasa_update-5.3.5-1.spkg` file to the HSM appliance, where *<private_key_file>* is the private portion of the SSH key that you provided when your HSM was provisioned.

```
$ scp -i <private_key_file> lunasa_update-5.3.5-1.spkg
 manager@<hsm_ip_address>:
```

5. Connect to the HSM appliance.

```
$ ssh -i <private_key_file> manager@<hsm_ip_address>
```

6. Verify that the package has been copied to the HSM.

```
lunash:> package listfile
  <file_size>  <date> lunasa_update-5.3.5-1.spkg
```

7. Log in to the HSM as the security officer.

```
lunash:> hsm login
```

8. Verify the Luna SA appliance software update with the following commands. The value to use for *<auth_code>* is in the `lunasa_update-5.3.5-1.auth` file contained in the `630-010165-023_RevA.tar` archive.

```
lunash:> package verify lunasa_update-5.3.5-1.spkg -authcode <auth_code>
```

9. Install the Luna SA appliance software update with the following commands. The value to use for *<auth_code>* is in the `lunasa_update-5.3.5-1.auth` file contained in the `630-010165-023_RevA.tar` archive.

```
lunash:> package update lunasa_update-5.3.5-1.spkg -authcode <auth_code>
```

10. Reboot the HSM appliance with the following command:

```
lunash:> sysconf appliance reboot
```

Upgrade HSM Firmware

If your HSM is running firmware version 6.2.1, 6.10.2, or 6.20.1, you should apply the update available on this page as soon as possible to resolve a critical issue. This update is a single package that will upgrade any firmware version.

When applying this update to appliance software 5.1, the firmware is automatically updated to version 6.2.5. When applying this update to any later version of the appliance software, you must choose which firmware version to apply. The following options are available:

- 6.2.5 (FIPS validated)
- 6.10.7 (FIPS candidate; will replace 6.2.5 when FIPS validation is achieved)
- 6.20.2 (latest version, not a FIPS candidate)

If you do not require FIPS validation, we recommend installing firmware version 6.20.2. Otherwise, you can install firmware version 6.10.7 or 6.2.5 depending on your requirements.

For more information about the changes in firmware versions 6.20.x, go to the SafeNet Luna HSM 5.3.5 Customer Release Notes. For more information about the changes in firmware versions 6.10.x, go to the SafeNet Luna HSM 5.2.6 Customer Release Notes.

To upgrade the HSM firmware

1. Download the Luna SA appliance firmware upgrade package from https://s3.amazonaws.com/cloudhsm-software/lunafwsuite.tgz, and then extract the files from the archive.
2. Use the following command to copy the `lunafwsuite.spkg` file to the HSM appliance, where *<private_key_file>* is the private portion of the SSH key that you provided when your HSM was provisioned.

   ```
   $ scp -i <private_key_file> lunafwsuite.spkg manager@<hsm_ip_address>:
   ```
3. Use the following commands to connect to the HSM appliance and log in to the HSM:

   ```
   $ ssh -i <private_key_file> manager@<hsm_ip_address>
   ```

   ```
   lunash:> hsm login
   ```
4. Use the following command to update the firmware package. The value to use for *<auth_code>* is in the `lunafwupdate.auth` file contained in the `lunafwsuite.tgz` archive.

   ```
   lunash:> package update lunafwsuite.spkg -authcode <auth_code>
   ```
5. Use the following command to update the HSM appliance firmware.

   ```
   lunash:> hsm update firmware
   ```

   If your HSM is running appliance software version 5.3.x, you must choose a firmware version during this step. If you do not require FIPS validation, we recommend choosing version 6.20.2. You can choose version 6.10.7 (FIPS candidate) or version 6.2.5 (FIPS validated) if required.
6. Use the following command to reboot the HSM appliance:

   ```
   lunash:> sysconf appliance reboot
   ```

   You can verify the new firmware version by issuing the **hsm show** command.

Document History

The following table describes the important changes to the documentation in this release of AWS CloudHSM.

- **Latest documentation update:** December 8th, 2016

Change	Description	Date Changed
Update	Added support for the Canada (Central) region.	December 8th, 2016
Update	Added support for the US East (Ohio) region.	October 17th, 2016
Update	Added support for the US West (N. California) region.	September 20th, 2016
Update	Updated the instructions for Automatically Setting Up Your AWS CloudHSM Environment Using AWS CloudFormation (p. 6). Updated the instructions for Manually Setting Up Your AWS CloudHSM Environment (p. 9). Updated the instructions for Configuring a Linux HSM Client (p. 27). Updated the instructions for Configuring a Windows HSM Client (p. 29). Updated the instructions that explain how to Upgrade Client Software to 5.4 (p. 110).	September 15th, 2016
Update	Modified the instructions for Installing the CLI Tools (p. 16).	November 18th, 2015

Change	Description	Date Changed
Update	Added information about Payment Card Industry (PCI) Data Security Standard (DSS) Compliance (p. 1) for AWS CloudHSM.	October 28th, 2015
Update	Simplified the instructions for setting up an AWS CloudHSM environment automatically using AWS CloudFormation. For details, see Setting Up Your AWS CloudHSM Environment Using AWS CloudFormation (p. 8).	July 23rd, 2015
Update	Added new instructions for installing the AWS CloudHSM CLI Tools on Amazon Linux. For details, see Installing the AWS CloudHSM CLI Tools (p. 17). Updated the recommended SSH configuration for connecting to an HSM appliance. For details, see SSH Connections (p. 19). Added a new limit for HSM appliances and corrected other limits. For details, see AWS CloudHSM Limits (p. 96).	July 10th, 2015
Update	Added new supported HSM firmware versions and updated the recommendations and instructions for upgrading HSM firmware. For details, see Upgrade HSM Firmware (p. 114).	July 9th, 2015
Update	Added support for Asia Pacific (Singapore) and Asia Pacific (Tokyo) regions.	June 8th, 2015
New guide name AWS CloudHSM CLI added	The AWS CloudHSM Getting Started Guide is now the *AWS CloudHSM User Guide*. This includes a major rewrite and restructuring of the guide. See the command line interface documentation at AWS CloudHSM Command Line Interface Tools Reference (p. 56).	January 8th, 2015

Change	Description	Date Changed
Update	Added support for US West (Oregon) and Asia Pacific (Sydney) regions; new sections on high availability and load balancing; new sections on resources to deploy and integrate with third-party applications; and instructions on how to use a new AWS CloudFormation template to set up your AWS CloudHSM environment.	November 5th, 2013
Initial Release	First release of the AWS CloudHSM Getting Started Guide.	March 26th, 2013