Externalizing ISAM Runtime Database to DB2

# Contributors

|  |  |
| --- | --- |
| William Daly | [wadaly@gmail.com](mailto:wadaly@gmail.com) |
| Julia Lam | [julialam001@gmail.com](mailto:julialam001@gmail.com) |
|  |  |

# Overview

In this document, we provide instructions to perform the following:

1. Create an ISAM lab environment on AWS
2. Externalize the ISAM runtime database using manual methods
3. Externalize the ISAM runtime database using automated methods

# Assumptions

1. The ISAM runtime database is in active use and must be migrated to preserve user authentication state data.
2. ISAM is using an embedded ISAM runtime database in the current state.
3. DB2 will be used for the external runtime database in the future state.
4. DB2 will not be configured for HADR.
5. The ISAM virtual appliance form factor is being used. These instructions do not apply to ISAM running on Docker, which does not support an embedded ISAM runtime database.
6. An ISAM trial key will be used to activate the ISAM virtual appliance's Web, AAC, and Federation capabilities.

# Prerequisites

1. Amazon Web Services account
2. ISAM trial license

# Environment

Amazon Web Services

Ansible

cURL

DB2 for Docker

Docker

Docker Compose

Git

ISAM 9.0.7.2

Ubuntu

# Procedure

## Create ISAM Lab Environment

We will create the ISAM lab environment one time and use it for both the manual and automated ISAM V9 DB2 runtime database externalization.

### Deploy ISAM Virtual Appliance

Login into your AWS account.

<https://aws.amazon.com>

Navigate to EC2->Instances->Launch Instance.

Search AWS Marketplace for IBM Security Access Manager v9.

Next to the **IBM Security Access Manager v9** offering, click on the Select button.

On the IBM Security Access Manager v9 pricing details page, click on the **Continue** button.

Choose the EC2 instance type **t2.medium** or some other instance type which meets your requirements and click on the **Next: Configure Instance Details** button.

Accept the defaults on the Configure Instance Details page and click on the **Next: Add Storage** button.

Accept the defaults on the Add Storage page and click on the **Next: Add Tags** button.

Click on the **Add Tag** button. For Key, enter "Name". For Value, enter "ISAM V9", then click on the **Next: Configure Security Group** button.

Note: During installation of the ISAM virtual appliance a default security group will be configured to permit connectivity from 0.0.0.0/0 (all sources) to SSH on port 22 and HTTPS on port 443 for the ISAM virtual appliance. This is neither necessary nor advisable. You can lock down this security group so it permits connectivity from only your workstation. Determine your workstation's IP address by navigating to the following URL: <https://www.whatsmyip.org/>. Your IP address will be shown at the top of the page. In the rows for SSH and HTTPS in the security group configuration, replace the CIDR string "0.0.0.0/0" with the value of your IP address followed by "/32". For example, if you IP address is shown as "1.2.3.4", then replace the value "0.0.0.0/0" in both the SSH and HTTPS row with "1.2.3.4/32". In addition, add a custom TCP rule to permit connectivity from "1.2.3.4/32" to port 444.

Note: Add more detail to include security allowance to talk with Isam

Add an additional rule to the security group to permit all traffic for the ISAM V9 security group.

Change the security group name and description to ISAM V9. This will promote ease of reference.

Click on the "Review and Launch" button.

You will be taken to the Review Instance Launch page. Click on the Launch button.

You will be prompted to select an existing key pair or create a new key pair. You will need this SSH key pair to connect to the virtual appliance text console (but not to connect the ISAM local management interface). If you are unfamiliar with connecting to AWS EC2 Linux type instances using SSH keys, the references section of this document contains a link where you can find more information.

Select the checkbox acknowledging you have access to the SSH key and click on the **Launch Instance** button.

Navigate to EC2-Instances in the AWS console and select the ISAM V9 EC2 instance from the Instances list.

In your AWS console, access you ISAM V9 Instance ID and Public DNS name in the Description tab. Record these two values for your future reference.

In your browser, (Firefox, Edge, Safari) enter **https:/**/ followed by the DNS name for your ISAM V9 virtual appliance.

Note: Your IP address can change. If you are working on this lab over successive days and you are no longer able to reach your AWS resources, confirm your IP address and update your security groups if required.

You will be prompted with a browser warning. The reason for this warning is that the ISAM appliance SSL certificate is not issued by a certificate authority which your browser knows and trusts. You must accept the risk in order to continue connecting the ISAM V9 virtual appliance.

When you are prompted to log in, for the username, enter "admin" and for the password, enter the value of the Instance ID to the ISAM V9 instance assigned by AWS EC2.

You will be taken to the Software License Agreement. Select the check box next to "I agree" and click **Next page** to continue.

You will be taken to the FIPS 140-2 Mode Configuration page. Click the **Next page** button.

You will be taken to the Licensing page. Click the **Next page** button.

You will be taken to the Snapshots page. Click the **Next page** button.

You will be taken to the Administrator Settings page. Click on the **Next page** button.

You will be taken to the Networking Configuration page. Enter the following value in the **Host name** text box: ***isam.test.com***. and click the **Save Configuration** button.

Click on the **Next page** button.

You will be taken the Data/Time page. Click on the **Next page** button.

You will be taken to the Complete Setup page. Click the **Complete Setup** button.

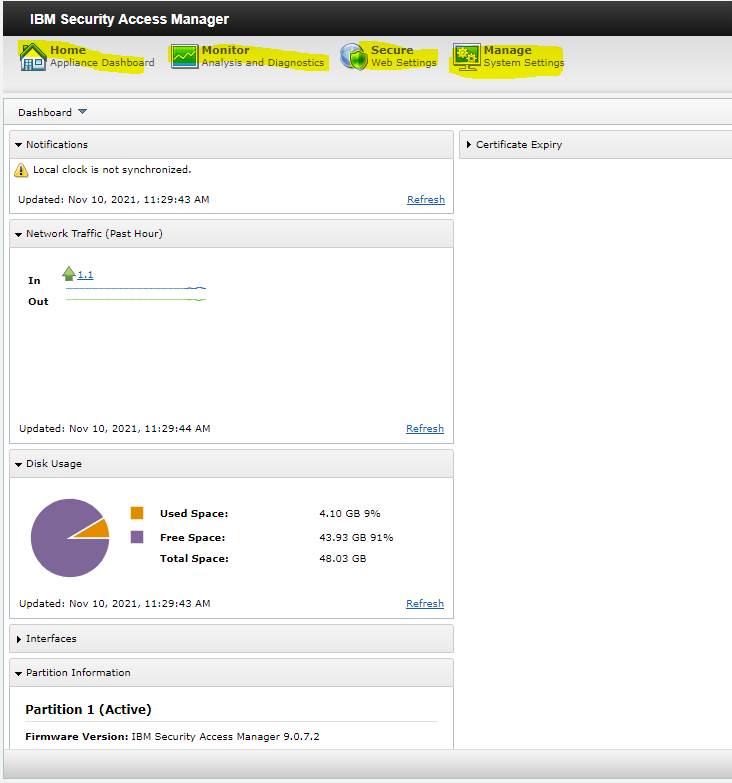
It will take a few moments for the ISAM virtual appliance to complete the setup.

### Apply ISAM Trial License

If you do not already have an ISAM trial key, you can register, request, and download one using the following URL: <https://ibm.biz/isamtrial>.

Reconnect to the ISAM virtual appliance using the DNS name.

Note: You should see four menu items at the top of the ISAM V9 Local Management Interface as shown below.



In the ISAM V9 LMI, navigate to Manage System Settings->Updates and Licensing->Trial.

Click the **Import** button.

Use the file dialog to locate your ISAM trial license file.

Click the Save Configuration button.

Allow several moments for the ISAM LMI to restart.

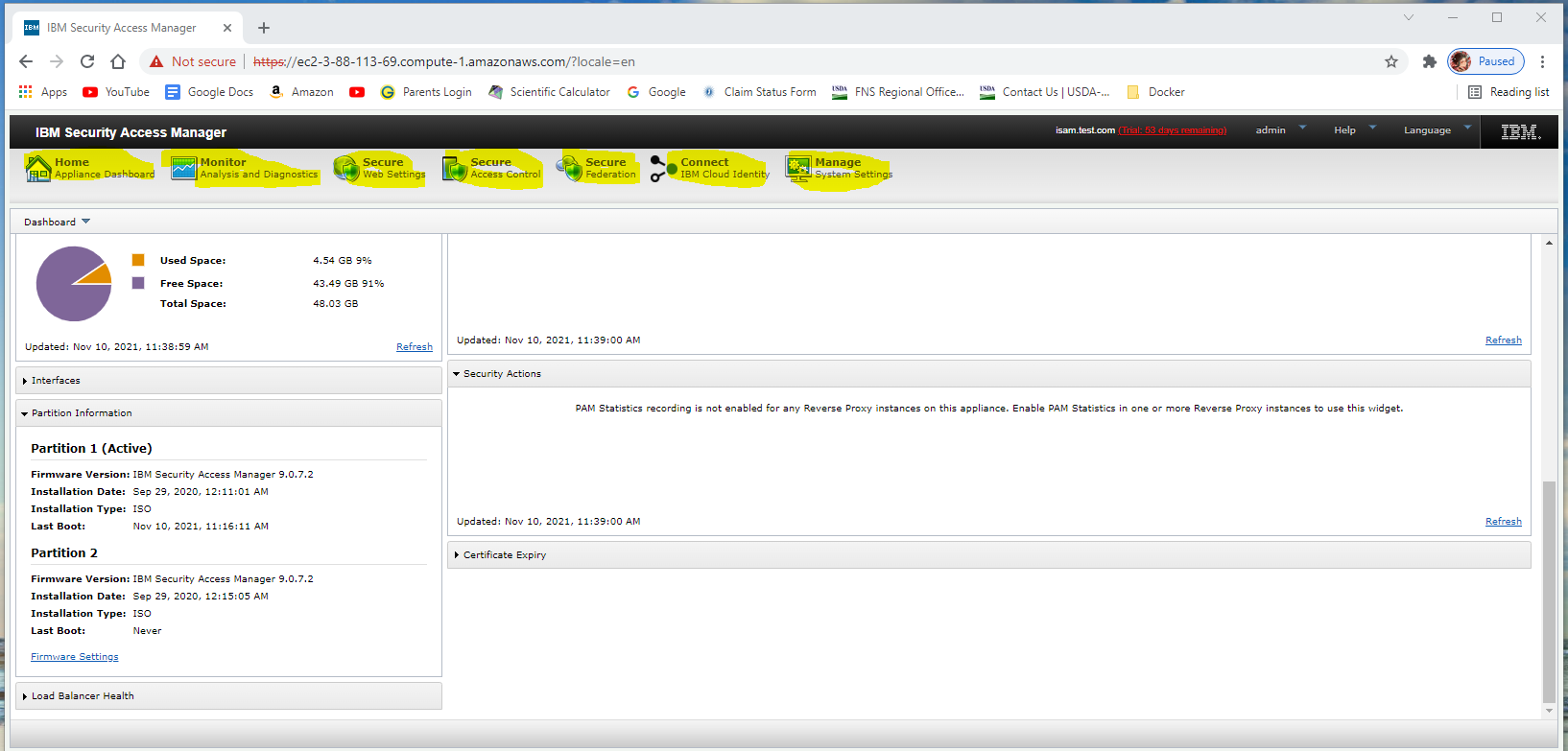
When the LMI restarts you will see the following:

Insert screenshot here

Click on the link at the bottom of the blue bordered dialog to reload the ISAM V9 LMI, or simply navigate back to the ISAM V9 DNS name captured earlier.

If the trial license has been accepted, you should now see an expanded list of menu items in the ISAM V9 LMI:

Insert screenshot here



The ISAM V9 appliance web, AAC, and federation features have now been activated.

### Set the password for the embedded ISAM LDAP server

Navigate to Secure Web Settings->Manage->Runtime Component.

Click the **Manage** button and navigate to Embedded LDAP->Change Password.

For **Administrator Password** and **Confirm Administrator Password**, enter the value "**Passw0rd**".

Click the **Submit** button.

You have now set the password for the embedded ISAM LDAP server.

A yellow banner will appear containing a link which reads **Click here to review the changes or apply them to the system**. Click this link, then click the Deploy button to deploy pending changes.

### Configure the ISAM V9 runtime environment

In the ISAM V9 LMI, navigate to Secure Web Settings->Manage->Runtime Component.

Click the **Configure** button.

In the Main tab, for Policy Server, select the **Local** radio button. Also, in the Main tab, for User Registry, select the **LDAP Local** radio button.

Click the **Next** button.

In the Policy Server tab, for the **Administrator Password** and **Confirm Administrator Password**, enter the value "Passw0rd".

Click the **Finish** button.

When the configuration completes successfully, the Runtime Component Status will show up as available.

### Create Reverse Proxy

In the ISAM V9 LMI, navigate to Secure Web Settings->Manage->Reverse Proxy.

Click the **New** button.

In the Instance tab, for **Instance Name**, enter the value **default**, and for **IP Address of the Primary Interface**, enter **0.0.0.0**.

Click the **Next** button.

In the IBM Security Access Manager tab, for **Administrator Password**, enter the value "Passw0rd".

Click the **Next** button.

In the Transport tab, ensure that **Enable HTTP** is not checked, **Enable HTTPS** is checked, and that **HTTPS Port** is set to the value **444**.

Click the **Finish** button.

The ISAM reverse proxy is now configured.

### Configure Reverse Proxy for DSC

In the ISAM V9 LMI, navigate to Secure Web Settings->Manage->Reverse Proxy

Select the **default** reverse proxy and click on the **Edit** button.

Click on the **Session** tab and select **Enable Distributed Session Cache**.

Click the **Save** button.

A yellow banner will appear containing a link which reads **Click here to review the changes or apply them to the system**. Click this link, then click the Deploy button to deploy pending changes.

Ensure the the **default** reverse proxy is selected and click the **Restart** button.

The reverse proxy is now configured for DSC.

### Create Users

We will now create five users in order to test the ISAM V9 AAC capability.

Log into the ISAM console using your SSH key. Substitute the public DNS address of the ISAM V9 virtual appliance where shown.

|  |
| --- |
| ssh -i us-east-1.pem admin@<ISAM\_DNS\_NAME>  ...  ...  Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  ...  ...  isam.test.com> |

Log in to the pdadmin command line interface.

|  |
| --- |
| isam.test.com> isam  isam.test.com:isam> admin  pdadmin> login  Enter User ID: sec\_master  Enter Password:  pdadmin sec\_master> |

Issue the following pdadmin commands to create 5 ISAM test user accounts.

|  |
| --- |
| user create user001 uid=user001,dc=iswga User 001 Passw0rd  user modify user001 account-valid yes  user create user002 uid=user002,dc=iswga User 002 Passw0rd  user modify user002 account-valid yes  user create user003 uid=user003,dc=iswga User 003 Passw0rd  user modify user003 account-valid yes  user create user004 uid=user004,dc=iswga User 004 Passw0rd  user modify user004 account-valid yes  user create user005 uid=user005,dc=iswga User 005 Passw0rd  user modify user005 account-valid yes |

Log out of pdadmin and log out of the ISAM console.

|  |
| --- |
| pdadmin sec\_master> exit  isam.test.com:isam> exit |

The ISAM test users are now created.

### Set the easuser password required by the AAC component

In the ISAM V9 LMI, navigate to Secure Access Control->Global Settings->User Registry.

In the Users table, select **easuser**.

Click the **Set Password** button.

In the **New Password** and the **Confirm New Password** boxes, enter the value "**Passw0rd**".

Click the **OK** button.

A yellow banner will appear containing a link which reads **Click here to review the changes or apply them to the system**. Click this link, then click the Deploy button to deploy pending changes.

The easuser password is now updated.

### Configure AAC

In the ISAM V9 LMI, navigate to Secure Web Settings->Manage->Reverse Proxy

Select the **default** reverse proxy and click Manage->AAC and Federation Configuration->Authentication and Context Based Access Configuration.

In the Authentication and Context Based Access Configuration dialog, click the **Next** button.

In the AAC Runtime tab, in the Password field, enter the value "**Passw0rd"**.

Leave all other values unchanged.

Click the **Finish** button.

A yellow banner will appear containing a link which reads **Click here to review the changes or apply them to the system**. Click this link, then click the Deploy button to deploy pending changes.

Select the default reverse proxy instance and click the **Restart** button.

AAC is now configured.

### Configure the Username Password Authentication Mechanism for AAC

In the ISAM V9 LMI, navigate to Secure Access Control->Policy->Authentication.

Click the **Mechanisms** link.

Scroll down and select the **Username Password** method.

Click on the **Modify Authentication Mechanism** icon.

Select the **Properties** tab.

Enter the following values by selecting each item separately and then clicking the **Modify Property** button (which resembles a pencil on a sheet of paper).

|  |  |
| --- | --- |
| Name | Value |
| LDAP Bind DN | cn=root,secAuthority=Default |
| LDAP Bind Password | Passw0rd |
| LDAP Host Name | localhost |

Click the **Save** button.

A yellow banner will appear containing a link which reads **Click here to review the changes or apply them to the system**. Click this link, then click the Deploy button to deploy pending changes.

The Username Password authentication mechanism is now configured.

### Restart the ISAM Virtual Appliance

### Log into AAC with test users

We will log into AAC using user002 and user004.

In a new browser tab, navigate to the following URL:

https://{{ISAM\_DNS\_NAME}}:444/mga/sps/authsvc?PolicyId=urn:ibm:security:authentication:asf:password\_eula

Log in as user002 with the password "Passw0rd".

Select the radio button next to the label " I accept the terms of the license agreement".

Click the **submit** button.

Repeat these steps for user004.

When we migrate the embedded ISAM runtime database to DB2, we will test these users again. For users user002 and user004 we should not be prompted again to accept the licensing terms again. However, for users user001, user003, and user005, we should be prompted. This will serve as out validation test that the database migration has been completed successfully.

### Deploy an Ubuntu AWS EC2 Instance

Navigate to the following location for the user Ubuntu userdata content:

<https://github.com/wadaly/devops_demos/blob/main/isam_externalize_runtime_db2/ec2_ubuntu_user_data.txt>

Login into your AWS account.

Navigate to EC2->Instances->Launch Instance.

On the Choose an Amazon Machine Image (AMI) page, search for Ubuntu Server 20.04.

Choose Ubuntu Server 20.04 LTS (HVM), SSD Volume Type and click the **Select** button.

On the Choose an Instance Type page, select a t2.medium instance type, or different instance type according to your needs.

Click the Next: Configure Instance Details button.

Accept all defaults according to your needs, except for User data. For User data, enter the User data content retrieved from Github.

Click the Next: Add Storage button.

Configure 40 GiB of storage and click the Next: Add Tags button.

Click the Add Tag button. For Key, enter "Name" in the text field. For Value, enter "Ansible, Docker, Ubuntu" in the text field. Click Next: Configure Security Group.

On the Configure Security Group page, configure the security group to accept connections from only your IP address. Use the same method you used when configuring the ISAM security group.

Add an additional rule to the security group to permit all traffic for the ISAM V9 security group.

Change the security group name and description to AnsibleDockerUbuntu. This will promote ease of reference.

Click the Review and Launch button.

On the Review Instance Launch page, click the Launch button.

On the select an existing key pair or create a new key pair page, click the check box to acknowledge that you have access to the SSH private key file and click the Launch Instance button.

You have now deployed the Ansible Docker Ubuntu EC2 instance

### Launch DB2 container

Log into the Ansible Docker Ubuntu EC2 instance using your SSH key.

|  |
| --- |
| Williams-MacBook-Pro:Downloads wadaly$  ssh -i us-east-1.pem ubuntu@<ANSIBLE\_DOCKER\_UBUNTU\_DNS\_NAME>  Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.11.0-1020-aws x86\_64) |

Use Got to clone the Git repository containing the Docker Compose file.

|  |
| --- |
| ubuntu@ip-172-31-80-219:~$ pwd  /home/ubuntu  ubuntu@ip-172-31-80-219:~$ ls -l  total 0  ubuntu@ip-172-31-80-219:~$ git clone https://github.com/wadaly/devops\_demos.git  Cloning into 'devops\_demos'...  remote: Enumerating objects: 126, done.  remote: Counting objects: 100% (126/126), done.  remote: Compressing objects: 100% (113/113), done.  remote: Total 126 (delta 58), reused 0 (delta 0), pack-reused 0  Receiving objects: 100% (126/126), 26.48 KiB | 3.78 MiB/s, done.  Resolving deltas: 100% (58/58), done.  ubuntu@ip-172-31-80-219:~$ |

Navigate to /home/ubuntu/devops\_demos/isam\_externalize\_runtime\_db2.

|  |
| --- |
| ubuntu@ip-172-31-80-219:~$ cd /home/ubuntu/devops\_demos/isam\_externalize\_runtime\_db2  ubuntu@ip-172-31-80-219:~/devops\_demos/isam\_externalize\_runtime\_db2$ ls -l  total 40  -rw-rw-r-- 1 ubuntu ubuntu 959 Nov 7 21:06 README.md  -rw-rw-r-- 1 ubuntu ubuntu 703 Nov 7 21:06 docker-compose-isamdb2.yml  drwxrwxr-x 2 ubuntu ubuntu 4096 Nov 7 21:06 docs  -rw-rw-r-- 1 ubuntu ubuntu 915 Nov 7 21:06 ec2\_ubuntu\_user\_data.txt  drwxrwxr-x 2 ubuntu ubuntu 4096 Nov 7 21:06 inventory  -rw-rw-r-- 1 ubuntu ubuntu 177 Nov 7 21:06 isam.env  -rw-rw-r-- 1 ubuntu ubuntu 11513 Nov 7 21:06 isam\_playbook.yml  drwxrwxr-x 2 ubuntu ubuntu 4096 Nov 7 21:06 videos  ubuntu@ip-172-31-80-219:~/devops\_demos/isam\_externalize\_runtime\_db2$ |

Edit the .env file to provide the value "Passw0rd" for the value of the ISAM\_RT\_DB\_PW token.

|  |
| --- |
| ubuntu@ip-172-31-80-219:~/devops\_demos/isam\_externalize\_runtime\_db2$ cat .env  ISAM\_RT\_DB\_INST=db2inst1  ISAM\_RT\_DB\_PW=<secret>  ISAM\_RT\_DB\_NAME=isamdb  ISAM\_RT\_DB\_TZ=America/New\_York  ubuntu@ip-172-31-80-219:~/devops\_demos/isam\_externalize\_runtime\_db2$ |

Launch the ISAM runtime DB2 Docker container using Docker Compose.

|  |
| --- |
| ubuntu@ip-172-31-80-219:~/devops\_demos/isam\_externalize\_runtime\_db2$ docker-compose --file docker-compose-isamdb2.yml up -d  Creating network "isam\_externalize\_runtime\_db2\_default" with the default driver  Creating volume "isam\_externalize\_runtime\_db2\_db2data\_runtime" with default driver  Pulling isamdb2 (ibmcom/db2:)...  latest: Pulling from ibmcom/db2  93156a512b98: Pull complete  f8c518873786: Pull complete  5d4974261da2: Pull complete  2d3a12d55319: Pull complete  d8d137bd0181: Pull complete  0b0c43213599: Pull complete  650e3bc372c5: Pull complete  5e228a7979f4: Pull complete  a63dd39fe354: Pull complete  99abac3871f1: Pull complete  c54c8a1d0024: Pull complete  77be002fb54b: Pull complete  845b9a8effa4: Pull complete  Digest: sha256:f596bc24ab88ceb4bc7845a7dec1af14d9f50ef95d22742b9cce9fe8386001f2  Status: Downloaded newer image for ibmcom/db2:latest  Creating isamdb2 ... done  ubuntu@ip-172-31-80-219:~/devops\_demos/isam\_externalize\_runtime\_db2$ |

Docker will download the ibmcom/db2 Docker image the first time you run the container.

### Configure DB2

In the ISAM V9 LMI, navigate to Manage System Settings->Secure Settings->File Downloads.

Expand the access\_control folder.

Expand the database folder.

Expand the db2 folder.

Expand the runtime folder.

Highlight isam\_access\_control\_db2.sql and click the Export button.

Save isam\_access\_control\_db2.sql to your local workstation.

Copy isam\_access\_control\_db2.sql to the Docker host.

Example

|  |
| --- |
| scp -i us-east-1.pem isam\_access\_control\_db2.sql ec2-user@<ANSIBLE\_DOCKER\_HOST\_DNS\_NAME>:/tmp/isam\_access\_control\_db2.sql  isam\_access\_control\_db2.sql 100% 25KB 460.2KB/s 00:00 |

On the Docker host, use docker cp to copy isam\_access\_control\_db2.sql to the isamdb2 Docker container.

Example

|  |
| --- |
| docker cp /tmp/isam\_access\_control\_db2.sql isamdb2:/tmp/isam\_access\_control\_db2.sql |

From the docker host, log into the isamdb2 docker image.

|  |
| --- |
| docker exec -it isamdb2 bash |

On the isamdb2 docker container, su to the db2inst1 instance owner ID.

|  |
| --- |
| [root@isamdb2 tmp]# su - db2inst1  Last login: Tue Nov 2 13:55:44 EDT 2021  [db2inst1@isamdb2 ~]$ |

Edit the isam\_access\_control\_db2.sql file to perform the following global replacements:

|  |  |
| --- | --- |
| Replace | |
| Search Token | Replacement Value |
| &DBINSTANCE | db2inst1 |
| &DBUSER | db2inst1 |
| &DBPASSWORD | Passw0rd |

Use the db2 CLI to run the isam\_access\_control\_db2.sql file.

|  |
| --- |
| db2 -tsvf /tmp/isam\_access\_control\_db2.sql |

Review output for errors.

|  |
| --- |
| ATTACH TO db2inst1  Instance Attachment Information  Instance server = DB2/LINUXX8664 11.5.6.0  Authorization ID = DB2INST1  Local instance alias = DB2INST1  CREATE DATABASE HVDB ALIAS HVDB using codeset UTF-8 territory us PAGESIZE 16384 WITH "HVDB Tables"  DB20000I The CREATE DATABASE command completed successfully.  CONNECT TO HVDB USER db2inst1 USING  Database Connection Information  Database server = DB2/LINUXX8664 11.5.6.0  SQL authorization ID = DB2INST1  Local database alias = HVDB  CREATE TABLE HVDB\_SCHEMA\_UPDATES ( DSU\_INSTALL\_DATE TIMESTAMP NOT NULL, DSU\_VERSION INTEGER NOT NULL, DSU\_FILE VARCHAR(256) NOT NULL )  DB20000I The SQL command completed successfully.  COMMIT  DB20000I The SQL command completed successfully.  INSERT INTO HVDB\_SCHEMA\_UPDATES VALUES(CURRENT\_TIMESTAMP,201907029,'Install')  DB20000I The SQL command completed successfully.  COMMIT  DB20000I The SQL command completed successfully.  CREATE TABLE SCIM\_EAS\_EXT\_USERS ( EXT\_UID VARCHAR(64) NOT NULL, USER\_SHORTNAME VARCHAR(64) NOT NULL, CFGID INTEGER NOT NULL, CONSTRAINT EAS\_EXT\_UID\_UNIQE UNIQUE (EXT\_UID), CONSTRAINT SCIM\_EAS\_EXT\_USER\_PK PRIMARY KEY (EXT\_UID, USER\_SHORTNAME) )  DB20000I The SQL command completed successfully.  CREATE TABLE SCIM\_EAS\_EXT\_METHODS ( EXT\_MID VARCHAR(64) NOT NULL, EXT\_UID VARCHAR(64) NOT NULL, SCIM\_SCHEMA VARCHAR(256) NOT NULL, CONSTRAINT SCIM\_EAS\_EXT\_METH\_PK PRIMARY KEY (EXT\_MID), FOREIGN KEY (EXT\_UID) REFERENCES SCIM\_EAS\_EXT\_USERS(EXT\_UID) ON DELETE CASCADE )  DB20000I The SQL command completed successfully.  COMMIT  DB20000I The SQL command completed successfully.  ...snip...  ...snip...  DISCONNECT ALL  DB20000I The SQL DISCONNECT command completed successfully.  DETACH  DB20000I The DETACH command completed successfully.  [db2inst1@isamdb2 ~]$ |

### Export the ISAM Runtime Database

In the ISAM V9 LMI, navigate to Manage System Settings->Network Settings->Cluster Configuration.

Click on the Runtime Database tab.

Under the Database export section, ensure that Type is specified as DB2 and click the Export button.

Save the hvdb\_db2.zip file to your local machine.

Copy the file to the hvdb\_db2.zip file to the Docker host using scp.

Example

|  |
| --- |
| scp -i us-east-1.pem hvdb\_db2.zip ec2-user@<DOCKER\_HOST\_DNS\_NAME>:/tmp/hvdb\_db2.zip  hvdb\_db2.zip 100% 3798 140.0KB/s 00:00 |

On the Docker host, use docker cp to copy hvdb\_db2.zip to the isamdb2 Docker container.

Example

|  |
| --- |
| docker cp /tmp/hvdb\_db2.zip isamdb2:/tmp/hvdb\_db2.zip |

From the docker host, log into the isamdb2 docker container.

|  |
| --- |
| docker exec -it isamdb2 bash |

On the isamdb2 docker container, su to the db2inst1 instance owner ID.

|  |
| --- |
| su - db2inst1 |

Create a directory under /tmp for the hvdb\_db2.zip archive contents.

|  |
| --- |
| cd /tmp  mkdir hvdb\_db2 |

Navigate to directory /tmp/hvdb\_db2

|  |
| --- |
| cd hvdb\_db2 |

Extract the contents of /tmp/hvdb\_db2.zip.

|  |
| --- |
| unzip /tmp/hvdb\_db2.zip |

Review README.txt file.

|  |
| --- |
| IBM DB2 SQL Database Data Import  ================================  Steps:  1) Setup the initial empty runtime database instance using the files  located under the File Downloads area on the appliance for DB2.  2) Generate the ZIP file for the current appliance runtime database data  for DB2.  3) On the system containing the runtime database instance login as the DB2  instance user.  4) Create a temporary directory and unzip the contents of the appliance  runtime database data ZIP file into this temporary directory.  6) Ensure the IBM DB2 db2 executable is in the PATH.  7) If not already done (possibly by step (1)), attach and connect to the  runtime database, for example:  db2 ATTACH TO db2inst1  db2 CONNECT TO hvdb  8) In the temporary directory invoke the script to load the data:  db2 -vct -f import\_data\_hvdb.sql  The script will delete the existing data in the database and insert the  exported data extracted from the ZIP file.  9) The temporary directory and the files contained in it along with the  ZIP file can be removed.  10) Restart the appliance LMI. |

Initialize DB2 connections.

|  |
| --- |
| **db2 ATTACH TO db2inst1**  Instance Attachment Information  Instance server = DB2/LINUXX8664 11.5.6.0  Authorization ID = DB2INST1  Local instance alias = DB2INST1  **db2 CONNECT TO hvdb**  Database Connection Information  Database server = DB2/LINUXX8664 11.5.6.0  SQL authorization ID = DB2INST1  Local database alias = HVDB |

Run db2 import script.

|  |
| --- |
| db2 -vct -f import\_data\_hvdb.sql |

Review output for errors.

|  |
| --- |
| db2 -vct -f import\_data\_hvdb.sql  SET INTEGRITY FOR alias\_svc\_aliasuserpartner OFF  DB20000I The SQL command completed successfully.  LOAD FROM 'null.del' of DEL REPLACE INTO alias\_svc\_aliasuserpartner  SQL3501W The table space(s) in which the table resides will not be placed in  backup pending state since forward recovery is disabled for the database.  SQL3109N The utility is beginning to load data from file  "/tmp/hvdb\_db2/null.del".  SQL3500W The utility is beginning the "LOAD" phase at time "11/02/2021  18:29:53.151189".  SQL3519W Begin Load Consistency Point. Input record count = "0".  SQL3520W Load Consistency Point was successful.  SQL3110N The utility has completed processing. "0" rows were read from the  input file.  SQL3519W Begin Load Consistency Point. Input record count = "0".  SQL3520W Load Consistency Point was successful.  SQL3515W The utility has finished the "LOAD" phase at time "11/02/2021  18:29:53.238291".  SQL3500W The utility is beginning the "BUILD" phase at time "11/02/2021  18:29:53.242378".  SQL3213I The indexing mode is "REBUILD".  SQL3515W The utility has finished the "BUILD" phase at time "11/02/2021  18:29:53.312003".  Number of rows read = 0  Number of rows skipped = 0  Number of rows loaded = 0  Number of rows rejected = 0  Number of rows deleted = 0  Number of rows committed = 0  ...snip...  ...snip...  Number of rows read = 3  Number of rows skipped = 0  Number of rows loaded = 3  Number of rows rejected = 0  Number of rows deleted = 0  Number of rows committed = 3  SET INTEGRITY FOR user\_attributes\_values IMMEDIATE CHECKED  DB20000I The SQL command completed successfully.  COMMIT  DB20000I The SQL command completed successfully.  SET INTEGRITY FOR OAuthDBSchema.idaas\_oauth20client OFF  SQL3601W The statement caused one or more tables to automatically be placed  in the Set Integrity Pending state. SQLSTATE=01586  SET INTEGRITY FOR OAuthDBSchema.idaas\_oauth20client IMMEDIATE CHECKED  DB20000I The SQL command completed successfully.  COMMIT  DB20000I The SQL command completed successfully.  SET INTEGRITY FOR OAuthDBSchema.oauth20cache OFF  DB20000I The SQL command completed successfully.  SET INTEGRITY FOR OAuthDBSchema.oauth20cache IMMEDIATE CHECKED  DB20000I The SQL command completed successfully.  COMMIT  DB20000I The SQL command completed successfully.  SET INTEGRITY FOR OAuthDBSchema.oauth20clientconfig OFF  DB20000I The SQL command completed successfully.  SET INTEGRITY FOR OAuthDBSchema.oauth20clientconfig IMMEDIATE CHECKED  DB20000I The SQL command completed successfully.  COMMIT  DB20000I The SQL command completed successfully.  SET INTEGRITY FOR OAuthDBSchema.oauth20consentcache OFF  DB20000I The SQL command completed successfully.  SET INTEGRITY FOR OAuthDBSchema.oauth20consentcache IMMEDIATE CHECKED  DB20000I The SQL command completed successfully.  COMMIT  DB20000I The SQL command completed successfully.  [db2inst1@isamdb2 hvdb\_db2]$ |

### Identify the private IP address of the Docker host

Log into AWS.

Navigate to ECS->Instances.

Locate the private IP for the Docker host in the EC2 instance Description tab. Example: 172.31.31.249

### Create an ISAM hosts file entry for the Docker host

In the ISAM V9 LMI, navigate to Manage System Settings->Network Settings->Hosts File.

Click the New button.

For Address, enter the private IP address for the Docker host. Example: 172.31.31.249.

For Hostname, enter isamdb2.

### Reconfigure ISAM runtime database on the ISAM V9 virtual appliance

In the ISAM V9 LMI, navigate to Manage System Settings->Network Settings->Cluster Configuration.

Select the Runtime Database tab.

In the **Location of the database** section, select the **Remote to the cluster** radio button.

For **Type**, select **DB2**.

In the Database connection section, use the following values:

|  |  |
| --- | --- |
| Address | isamdb2 |
| Port | 50000 |
| Username | db2inst1 |
| Password | Passw0rd |
| Database name | hvdb |
| High availability | <unchecked> |

After entering the value above, click the Save button.

A yellow banner will appear containing a link which reads **Click here to review the changes or apply them to the system**. Click this link, then click the Deploy button to deploy pending changes.

|  |
| --- |
|  |

|  |
| --- |
|  |

### Testing the ISAM V9 runtime database externalization

In this test we will log into the ISAM reverse proxy using AAC for each of the following users:

* user001
* user002
* user003
* user004
* user005

Previously, we logged in with user002 and user004 and accepted the Enterprise User License Agreement (EULA). Since we have migrated the data in the ISAM V9 runtime database, the expected results is that user002 and user004 will not be prompted to accept the EULA following authentication, however, user001, user003, and user005 will.

For each user, perform the following steps.

In a new browser tab, navigate to the following URL:

https://{{ISAM\_DNS\_NAME}}:444/mga/sps/authsvc?PolicyId=urn:ibm:security:authentication:asf:password\_eula

Where ISAM\_DNS\_NAME is the public DNS name reserved for ISAM V9 as shown in the AWS EC2 console. For example:

<https://ec2-23-22-90-35.compute-1.amazonaws.com:444/mga/sps/authsvc?PolicyId=urn:ibm:security:authentication:asf:password_eula>

The ISAM V9 runtime database externalization test is now complete.

# Automated Database Externalization

The following steps will reproduce the ISAM V9 runtime database externalization, but in an automated fashion.

### Reconfigure the ISAM runtime database to embedded

Warning: The follows steps will reinitialize the ISAM V9 runtime database. All session state and authentication state data will be lost. This step should not be performed in a Production environment or any other environment where user session data should be preserved.

In the ISAM V9 LMI, navigate to Manage System Settings->Network Settings->Cluster Configuration.

Select the **Runtime Database** tab.

In the **Location of the database** section, select the **Local to the cluster** radio button.

At the bottom of the page, click the **Save** button.

A yellow banner will appear containing a link which reads **Click here to review the changes or apply them to the system**. Click this link, then click the Deploy button to deploy pending changes.

### Recreate ldapdb2 database

On the Docker host, run the following to stop the DB2 container and deallocate its volume storage.

|  |
| --- |
| docker-compose --file docker-compose-isamdb2.yml down -v |

Next, run the following to create a new, empty DB2 database instance.

|  |
| --- |
| docker-compose --file docker-compose-isamdb2.yml up -d |

The ldabdb2 database is now recreated.

### Repeat user login test

Refer to section **Log into AAC with test users** earlier in this document and repeat the steps described in this section.

### Update hosts file and isam.env file on Ansible Docker Ubuntu EC2 instance

On the Ansible Docker Ubuntu EC2 instance, navigate to /home/ubuntu/devops\_demos/isam\_externalize\_runtime\_db2.

|  |
| --- |
| cd /home/ubuntu/devops\_demos/isam\_externalize\_runtime\_db2 |

Edit the isam.env file.

For isam\_admin\_pw, provide the EC2 instance ID assigned by AWS. Example: i-0a1b2c3d4e5f60a1b.

For isam\_rt\_db\_pw, provide the value "Passw0rd".

For isam\_rt\_db\_host, provide the private IP address of the Ansible Docker Ubuntu EC2 instance assigned by AWS. Example: 1.2.3.4.

isam.env

|  |
| --- |
| isam\_admin\_id: "i-0a1b2c3d4e5f60a1b"  isam\_admin\_pw: "Passw0rd"  isam\_lmi\_port: "443"  isam\_rt\_db\_inst: "db2inst1"  isam\_rt\_db\_host: 1.2.3.4  isam\_rt\_db\_id: "db2inst1"  isam\_rt\_db\_pw: "Passw0rd" |

Update the /home/ubuntu/devops\_demos/isam\_externalize\_runtime\_db2/inventory/hosts file and replace <isam\_v9\_priv\_ip> with the value of the AWS-assigned private IP for you ISAM V9 virtual appliance.

hosts

|  |
| --- |
| [isam\_cluster\_master]  <isam\_v9\_priv\_ip> |

Run the ansible-playbook isam\_playbook.yml

|  |
| --- |
| ansible-playbook -u inventory/ isam\_playbook.yml |

The ISAM V9 runtime database is now externalized.

### Repeat validation steps

Refer to section **Testing the ISAM V9 runtime database externalization** from earlier in this document.

The ISAM V9 runtime database has now been externalized and validated.

# References

|  |  |  |
| --- | --- | --- |
| Item | URL | Remarks |
| Connecting to your Linux instance using SSH | <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstancesLinux.html> |  |
| Deploying an external runtime database | <https://www.ibm.com/docs/en/sva/9.0.7?topic=database-deploying-external-runtime> | To optimize performance or increase storage capacity for the appliance, you can deploy an external runtime database. You can configure the appliance to connect to SolidDB, DB2®, PostgreSQL, or Oracle database on an external server. |
| IBM DB2 Docker Image | <https://hub.docker.com/r/ibmcom/db2> |  |
| IBM Security Access Manager Version 9.0.7 June 2019 Administration topics | <https://www.ibm.com/support/pages/system/files/inline-files/$FILE/ISAM907_admin_isam.pdf> |  |
| Installing DB2 for Docker on Windows | <https://www.ibm.com/docs/en/db2/11.5?topic=system-windows> |  |
| Installing Docker Compose | <https://docs.docker.com/compose/install/> |  |
| Installing Docker on Amazon Linux 2 | <https://docs.aws.amazon.com/AmazonECS/latest/developerguide/docker-basics.html> |  |
| ISAM Configuration Database | <https://www.ibm.com/docs/en/sva/9.0.7?topic=configuration-database> | The configuration database stores configuration data, including policy information. This data is shared with all appliances in the cluster. |
| ISAM Runtime Database | <https://www.ibm.com/docs/en/sva/9.0.7?topic=administration-runtime-database> | The runtime database stores user data such as session attributes and device fingerprints. |
| ISAM Trial License | <https://ibm.biz/isamtrial> | Here you can register, request, renew, and download an ISAM trial key to activate the web, AAC, and federation features |
| ISAM V9 @ AWS Marketplace | <https://aws.amazon.com/marketplace/pp/prodview-kqpfs7sjttx4i> |  |

# Appendix A

Launch a new EC2 instance with the following settings:

|  |  |  |
| --- | --- | --- |
|  |  | Remarks |
| AMI | Amazon Linux 2 | ami-01cc34ab2709337aa as tested |
| Instance Type | t2.medium |  |
| Instance Details | See User Data below |  |
| Storage | 40 GiB |  |
| Tags | Name |  |
| Security Group | SSH port 22 from your workstation only |  |

The database load returns a warning state. Exit code 2.

<https://www.ibm.com/docs/en/db2/11.5?topic=clp-return-codes>