Article
Pete Greskoff · Jun 27, 2018 8m read

Creating SSL-Enabled Mirror on InterSystems IRIS Using Public Key Infrastructure (PKI)

In this post, I am going to detail how to set up a mirror using SSL, including generating the certificates and keys via the **Public Key Infrastructure** built in to InterSystems IRIS Data Platform. I did a similar post in the past for Caché, so feel free to check that out here if you are not running InterSystems IRIS. Much like the original, the goal of this is to take you from new installations to a working mirror with SSL, including a primary, backup, and DR async member, along with a mirrored database. I will not go into security recommendations or restricting access to the files. This is meant to just simply get a mirror up and running. Example screenshots are taken on a 2018.1.1 version of IRIS, so yours may look slightly different.

Step 1: Configure Certificate Authority (CA) Server

On one of your instances (in my case the one that will be the first mirror member configured), go to the Management Portal and go to the [System Administration -> Security -> Public Key Infrastructure] page. Here you will ‘ Configure local Certificate Authority server ’ .

You can choose whatever File name root (this is the file name only, no path or extension) and Directory you want to have these files in. I ’ ll use ‘ CAServer ’ as the File name root, and the directory will be my <install-dir>/mgr/CAServer/. This will avoid future confusion when the client keys and certificates are put into the <install-dir>/mgr/ folder, as I ’ ll be using my first mirror member as the CA Server. Go to the next page.

You will then need to enter a password, and I ’ ll use ‘ serverpassword ’ in my example. You can then assign attribute values for your Distinguished Name. I ’ ll set Country to ‘ US ’ and Common Name to ‘ CASrv ’ . You can accept defaults for validity periods, leave the email section blank, and save.

You should see a message about files getting generated (.cer, .key, and .srl) in the directory you configured.

Step 2: Generate Key/Certificate For First Mirror Member

At this point, you need to generate the certificate and keys for the instance that will become your first mirror member. This time, go to the Management Portal where you will set up the first mirror member, and go to the [System Administration -> Security -> Public Key Infrastructure] page again (see screenshot above). You need to ‘ Configure local Certificate Authority client ’ . For the ‘ Certificate Authority server hostname ’ , you need to put either the machine name or IP address of the instance you used for step 1, and for the ‘ Certificate Authority WebServer port number ’ use that instance ’ s web server port (you can get this from the URL in that instance ’ s Management portal):

Make sure you are using the port number for the instance you configured as the CA Server, not the one you are setting up as the client (though they could be the same). You can put your own name as the technical contact (the phone number and email are optional) and save. You should get a message “ Certificate Authority client successfully configured. ”

Now you should go to ‘ Submit Certificate Signing Request to Certificate Authority server ’ . You ’ ll need a file name (I ’ m using ‘ MachineAclient ’) and password (‘ MachineApassword ’) as well as again setting values for a Distinguished Name (Country= ’ US ’ and Common Name= ’ MachineA ’). Note that for each certificate you make, at least one of these values must be different than what was entered for the CA certificate. Otherwise, you may run into failures at a later step.

At this point, you’ll need to go to the machine you configured to be your CA Server. From the same page, you need to ‘Process pending Certificate Signing Requests’. You should see one like this:

You should process this request, leaving default values, and ‘Issue Certificate’. You’ll need to enter your CA Server password from step 1 (‘server_password’ for me).

Finally, you need to get the certificate. Back on the first mirror member machine, from the same page, go to ‘Get Certificate(s) from Certificate Authority server’, and click ‘Get’ like here:

If this is not the same machine where you configured the CA Server, you’ll need to get a copy of the CA Server certificate (‘CA_Server.cer’) also on this machine. Click ‘Get Certificate Authority Certificate’. That’s the top left button in the image above.

Step 3: Configure The Mirror On First Mirror Member

First, start the ISCAgent per **this documentation** (and set it to start automatically on system startup if you don’t want to have to do this every time your machine reboots).

Then, in the Management Portal, go to the [System Administration -> Configuration -> Mirror Settings -> Enable Mirror Service] page to enable the service (if it isn’t already enabled). Next, go to the ‘Create a Mirror’ page in the same menu.

You will need to enter a mirror name (‘PKIMIRROR’ in my case). You should click ‘Set up SSL/TLS’, and then enter the information there. the first line is asking for that CA server certificate (CA_Server.cer). For ‘This server’s credentials’, you’ll need to enter the certificate and key that we generated in step 2. They will be in the <install>/mgr/ directory. You’ll also need to enter your password here (click the ‘Enter new password’ button as shown). This password is the one you chose in step 2 (‘MachineA_password’ for me). In my example, I am only allowing TLS v1.2 protocol as shown below.

For this example, I won’t use an arbiter or a Virtual IP, so you can un-check those boxes in the ‘Create Mirror’ page. We’ll accept the defaults for ‘Compression’, ‘Parralel Dejournaling’, ‘Mirror Member Name’, and ‘Mi Agent Port’ (since I didn’t configure the ISCAgent to be on a different port), but I’m going to change the ‘Superserver Address’ to use an IP instead of a host name (personal preference). Just make sure that the other future mirror members are able to reach this machine at the address you choose. Once you save this, take a look at the mirror monitor [System Operation -> Mirror Monitor]. It should look something like this:

Step 4: Generate Key/Certificate For Second Failover Mirror Member

This is the same process as step 2, but I’ll replace anything with ‘MachineA’ in the name with ‘MachineB’. As I mentioned before, make sure you change at least 1 of the fields in the Distinguished Name section from the CA certificate. You also need to be sure you get the correct certificate in the Get Certificate step, as you will see both client certificates.

Step 5: Join Mirror as Failover Member

Just like you did for the first mirror member, you need to start the ISCAgent and enable the mirror service for this instance (refer to step 3 for details on how to do this). Then, you can join the mirror as a failover member at [System Administration -> Configuration -> Mirror Settings -> Join as Failover].

You’ll need the ‘Mirror Name’, ‘Agent Address on Other System’ (the same as the one you configured as the Superserver address for the other member), and the instance name of the now-primary instance.

After you click ‘Next’, you should see a message indicating that the mirror requires SSL/TLS, so you should again use the ‘Set up SSL/TLS’ link. You’ll replace machine A’s files and password with machine B’s for this dialog.

Again, I’m only using TLSv1.2. Once you’ve saved that, you should be able to add information about this mirror member. Again, I’m going to change the hostnames to IP’s, but feel free to use any IP/hostname that the other member can contact this machine on. Note that the IP’s are the same for my members, as I have set this up with multiple instances on the same server.

Step 6: Authorize 2nd Failover Member on the Primary Member

Now we need to go back to the now primary instance where we created the mirror. From the [System Administration -> Configuration -> Mirror Settings -> Edit Mirror] page, you should see a box at the bottom titled ‘Pending New Members’ including the 2nd failover member that you just added. Check the box for that member and click Authorize (there should be a dialog popup to confirm).

Now if you go back to [System Operation -> Mirror Monitor], it should look like this (similar on both instances):

If you see something else, wait a minute and refresh the page.

Step 7: Generate Key/Certificate for Async Member

This is the same as step 2, but I’ll replace anything with ‘MachineA’ in the name with ‘MachineC’. As I mentioned before, make sure you change at least 1 of the fields in the Distinguished Name section from the CA certificate. Make sure you get the correct certificate in the ‘Get Certificate’ page, as you will see all 3 certificates.

Step 8: Join Mirror as Async Member

This is similar to step 5. The only difference is that you have the added option for an Async Member System Type (I will use Disaster Recovery, but you’re welcome to use one of the reporting options). You’ll again see a message about requiring SSL, and you’ll need to set that up similarly (MachineC instead of MachineB). Again, you’ll see a message after saving the configuration indicating that you should add this instance as an authorized async on the failover nodes.

Step 9: Authorize Async Member on the Primary Member

Follow the same procedure as in step 6. The mirror monitor should now look like this:

Step 10: Add a Mirrored Database

Having a mirror is no fun if you can’t mirror any data, so we may as well create a mirrored database. We will also create a namespace for this database. Go to your primary instance. First, go to [System Administration -> Configuration -> System Configuration -> Namespaces] and click ‘Create New Namespace’ from that page.

Choose a name for your namespace, and we’ll need to click ‘Create New Database’ next to ‘Select an existing database for Globals’. You’ll need to enter a name and directory for this new database. On the next page, be sure to change the ‘Mirrored database?’ drop-down to yes (THIS IS ESSENTIAL). The mirror name will default to the database name you chose. You can change it if you wish. We will use the default setting for all other options for the database (you can change them if you want, but this database must be journaled, as it is mirrored). Once you finish that, you will return to the namespace creation page, where you should select this new database for both ‘Globals and ‘Routines’. You can accept the defaults for the other options (don’t copy the namespace from anywhere).

Repeat this process for the backup and async. Make sure to use the same mirror name for the database. Since it ’ s a newly created mirrored database, there is no need to take a backup of the file and restore onto the other members.

Congratulations, you now have a working mirror using SSL with 3 members sharing a mirrored database! One final look this time at the async ’ s mirror monitor:

Other reference documentation:

Create a mirror

Create mirrored database

Create namespace and database

Edit failover member (contains some information on adding SSL to an existing mirror)

#Best Practices #High Availability #Mirroring #SSL #System Administration #InterSystems IRIS

Source URL:https://community.intersystems.com/post/creating-ssl-enabled-mirror-intersystems-iris-using-public-key-infrastructure-pki