**Configure Hyper-V Certificate-Based Authentication (HTTPS) Replica in a clustered environment**

Again, we are going to use our AD Certificate Services to issue the required certificates. Now, for [the cluster](http://www.vkernel.ro/blog/creating-a-hyper-v-server-2012-cluster), the certificate needs to have the FQDN of the Replica Broker name in the Common Name field or in the Subject Alternative Name field. Don’t worry, I’ll go trough this and show you how to issue the certificate as we move forward, but first you have to request a computer certificate for all your Hyper-V servers that are participating in replication. I already talked about how to request computer certificates in the section above, so in case you skipped it, [click here](http://www.vkernel.ro/blog/configuring-hyper-v-replica-using-certificate-based-authentication-https#anchor009) to see how it’s done; and again if you want to automate the certificate enrollment using group policy [read this guide](http://www.vkernel.ro/blog/set-up-automatic-certificate-enrollment-autoenroll). For this I’m going to use a Hyper-V host joined to the domain that will replicate VMs to a [Windows 2012 Hyper-V cluster](http://www.vkernel.ro/blog/creating-a-hyper-v-server-2012-cluster).

Before we can issue a certificate for the Replica Broker we need to create a new certificate template on our AD CS. This is because the actual **Computer** certificate template takes the information from AD (the name of the computer), and it will issue the certificate with the Hyper-V host name in the Common Name of the certificate. This is not good, we need to have the Replica Broker name in that field. Log in to your AD CS server, open the **Certification Authority** management console, right-click **Certificate Templates** and choose **Manage**.



Find the **Computer** template, right-click it and choose **Duplicate Template**. Once the template properties window opens, click the **General** tab and in the **Template display name** field type a name for this template. Still here, you can increase the validity period so you don’t have to renew the certificate every year. I increased mine to two years, but I’ll leave this up to you.



Click the **Request Handling** tab and check the **Allow private key to be exported** box. This is so we can export the certificate with it’s private key, later on.



The most important tab here for us is the **Subject Name** tab. As you can see, and as I told you, the information in the certificate using the **Computer** template is build from AD. To change that just click the **Supply in the request** radio button, and **OK** on the warning message. Don’t worry about that message because you can secure the template ( Security tab ) so only specific users can use it to request certificates, but I am not going to worry about this right now. Click **OK** when done.



Go back to your **Certification Authority** console right-click the **Certificate Templates** folder again and choose **New > Certificate Template to Issue**. Search the template you just created, select it and hit **OK**.

    

Now we need to request a certificate for the Replica Broker role using the new template, so log in to one of your cluster nodes, open the **Certificates** store console (**Start > Run > mmc**) and choose **Computer account > Local computer**. Right-click the personal folder and choose **All Tasks > Request New Certificate**.



On the wizard that opens, accept the defaults until you reach the **Request Certificates** page. As you can see, you now have a new template in the list, but this new template needs more information before the CA can issue a certificate from it. Click the link bellow the template to open its properties.

On the **Subject** tab, the most important setting to configure is the common name of the certificate. From the **Type** box choose **Common Name** and in the **Value** box type the FQDN of your Replica Broker role. Hit the **Add** button when you are done. Before you click **OK**, verify that the private key is exportable. You can check this from the **Private Key tab > Key options**. This is so you don’t have to request a certificate with the same name on all your cluster nodes, it  is way easier to just export it once and import it on the rest of the servers. Click **OK**

    

and **Enroll**.

    

Once you click **Finish** on the wizard a new certificate with its private key appears in the console. By now you should have two certificates on this Hyper-V host, one that has the name of the machine in the common name field and the other one with the common name of your Replica Broker role.



The next step is to export this certificate with its private key and import it on all the cluster nodes. Right-click the certificate and choose **All Tasks > Export**.



Once you get to the **Export Private Key** page make sure you choose the first option **Yes, export the private key**.

    

On the next page of the wizard check the **Export all extended properties** box.



Type a password for the certificate and click **Next**. Make sure you remember this password because it’s needed on the import process later on.



Click **Browse** and choose where to store the certificate, then continue the wizard.



Once you click the **Finish** button, you should get a message that the export was successful.

    

Now go and import the certificate to your other Hyper-V servers that are part of the cluster. The easiest way is to just double-click the certificate and follow the **Certificate Import Wizard**. On the first screen select **Local Machine**.



Click **Browse** and provide the certificate. I had the certificate on another server and used and UNC path to import it, so don’t get scared about what you see in that box.



Here is the part where you need that password that you typed during the export wizard. Put in the password and click **Next**. As you can see you also have the option to mark the certificate private key as exportable. In my opinion, if you already have the  certificate and its private key somewhere safe, you don’t have to go with this option, but it is up to you.



Do not let the wizard to automatically select the certificate store, instead go with the second option, **Place all certificates in the following store**. Click **Browse** and choose **Personal**.

     

Click **Finish**. You should get a message that the import was successful. Repeat this for all the remaining nodes in the cluster.



Right now certificates are in place, every Hyper-V server has a computer certificate and the cluster nodes also have the Replica Broker certificate. If only one of the two certificates is missing from one of the cluster nodes, you will get the following error message:

*In addition to the certificate for Hyper-V Replica Broker, each node of the cluster must also have a valid certificate with following properties: “…” Each node of the cluster must also have a computer authentication certificate that can be validated to the same root certificate as the certificate for Hyper-V Replica Broker.*



I got this because one of my nodes was missing the computer certificate. It only had the certificate for the Hyper-V Replica Broker; so make sure you have all the certificates in place.



Now let’s go and configure the Hyper-V cluster as a replica server. From one of the nodes open the **Failover Cluster Manager** console, go to **Roles**, right-click the Replica Broker role and choose **Replication Settings**.



If you followed this guide from the begging you should already be familiar with this view. Check the box **Enable this cluster as a Replica server** then **Use certificate-based Authentication (HTTPS)**. Now click the **Select Certificate** button, select the certificate and click **OK**, then choose where to store the replica VMs and **OK** again. Don’t forget to open port 443 if the servers are protected by a firewall.



To test this, move to the stand-alone Hyper-V server, right-click on one of the VMs and choose **Enable Replication**.



On the **Specify Replica Server** page you need to type the **Hyper-V Replica Broker** FQDN, NOT the name of one of the nodes in the cluster. To be more specific, the name you type here needs to correspond to the name from the Replica Broker certificate.



Click the **Select Certificate** button and choose your certificate.



Follow the wizard and configure your own settings. Once you click **Finish** the VM will start to replicate.



And the replication is working great. Now before I move forward to the last section of this article I want to mention that you can also [issue SAN certificates](http://www.vkernel.ro/blog/configure-internal-windows-ca-to-issue-san-certificates) and wildcard certificates to configure the HTTPS replication. It is up to you, and the company needs.