Domain Controller certificates: Kerberos Authentication template

ldap389.info/en/2010/09/06/powershell-domain-controller-certificate-kerberos-authentication-template

ldap389 September 6, 2010

Sep 06 2010

When you install Windows 2008 Certification Authority a new domain controller certificate template named *Kerberos Authentication* is available. It replaces the *Domain Controller Authentication* template. If you need more information about the new certificate templates shipped with a Windows 2008 CA you can read this article.

Here is a tab that outlines the specific attributes of the *Domain Controller Authentication* and *Kerberos Authentication* templates:

	Domain Controller Authentication	Kerberos Authentication
Key Usage	Client Authentication Server Authentication Smart Card Logon	Client Authentication Server Authentication Smart Card Logon **KDC Authentication.**
Subject Alternate Name	DNS Name : **Domain Controller** FQDN.	DNS Name : **Domain** FQDN. DNS Name : **Domain** NetBios name.

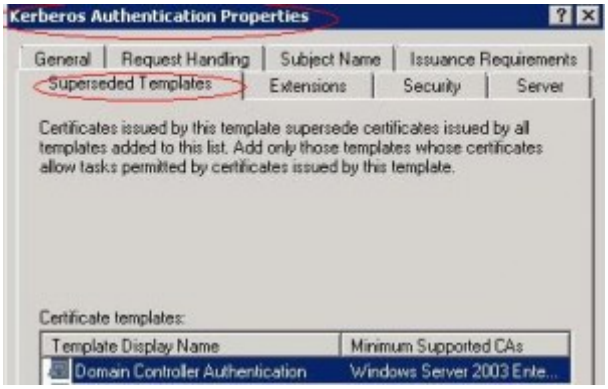
For more information about the *KDC Authentication* key usage that help assure that smart card users are authenticating against a valid Kerberos domain controller you can read this document: Enabling Strict KDC Validation in Windows Kerberos.

Having the domain name rather than the domain controller name in the *Subject Alternate Name* of the certificate proves that the computer presenting the certificate is a domain controller for the domain contained in the *Subject Alternate Name*. Domain name should also be included in the certificate in order to enable *Strict KDC Validation.*

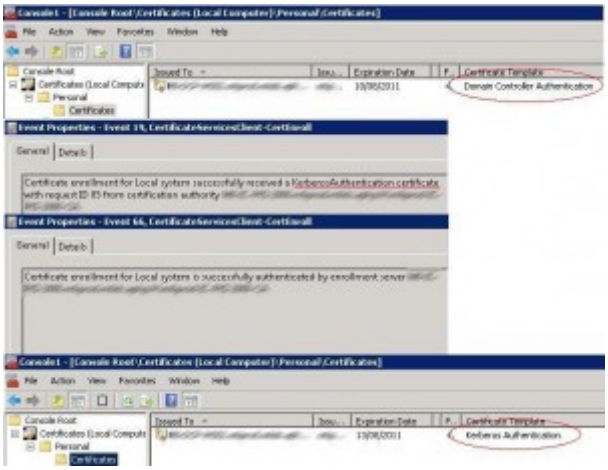
We will describe how to deploy the *Kerberos Authentication* template certificates on your domain controllers and how to revoke the old certificates issued with the *Domain Controller Authentication* template once they are useless. We distribute certificates to domain controllers using *autoenrollment* , to achieve this you need to configure your template (permissions, settings…) and setup a GPO.

If you want the new *Kerberos Authentication* template to replace the *Domain Controller Authentication* template, you need to configure it using *certtmpl.msc* by setting up the "Superseded Templates" tab. For more information you can have a look at the "Superseding Certificate Templates" chapter of this article.

Once the template is well configured and ready for autoenrollment, the new certificates will be deployed automatically, you can run the **certutil -pulse** command on the domain controllers, in order to speed up the autoenrollment process.



The new domain controller certificate is replaced in the local computer store, messages with source AutoEnrollment are displayed in the eventlog telling us that the *Kerberos Authentication* certificate is installed.



With Quest ActiveRoles Management Shell for Active Directory v1.4, you can manage certificates using PowerShell thanks to the Certificate and PKI management CmdLets. First we will check that the *Kerberos Authentication* certificates are installed on every Domain Controller:

```
Get-QADComputer -computerRole
'DomainController' | Get-
QADCertificate -Revoked:$false -
template:'*kerberos authentication*' |
format-table template,IssuedTo -
autosize
```

Once all your domain controllers have enrolled the new *Kerberos Authentication* certificates and you have checked everything is running properly, you can disable the old *Domain Controller Authentication* template with *certsrv.msc* in order to avoid installing this kind of certificate on a domain controller.

Then you can revoke the old *Domain Controller Authentication* certificates which where superseded by the *Kerberos Authentication* certificates. To achieve that we will combine the Quest CmdLets and the **Certutil -revoke** command. You just need to retrieve the *Domain Controller Authentication* certificates serial numbers and specify the reason code for the revocation of these certificates: In our case **4** for Superseded:

```
Get-QADComputer -computerRole 'DomainController' | Get-QADCertificate -
Revoked:$false -template:*domain controller authentication* | foreach {certutil
-config %SRV_CA_FQDN%\%CA_Common_Name% -revoke $_.SerialNumber 4}
```

You just need to adapt:

- %SRV_CA_FQDN%: Issuing CA server FQDN.
- %CA_Common_Name%: Certification Authority Common Name.

By combining the Certutil command line tool and Quest AD CmdLets v1.4, you can make some of your PKI management tasks automatic.

This post is also available in: French

Tags: certutil, pki, PowerShell

Filed in Public Key Infrastructure, scripts, security |