

# Re: ADFS Token–signing Certs Not in Trusted Root Store

---

*Source:*

[http://www.tech–archive.net/Archive/Windows/microsoft.public.windows.server.active\\_directory/2006–10/msg00792](http://www.tech–archive.net/Archive/Windows/microsoft.public.windows.server.active_directory/2006–10/msg00792)

---

- *From:* "Joe Kaplan" <[joseph.e.kaplan@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:joseph.e.kaplan@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Mon, 9 Oct 2006 13:37:43 –0500
- 

Yes, we decided to use our own token signing certs. There are really 3 approaches that I know of for this:

- Use the same SSL cert you use on the web server itself (since you'll already need this)
- Use a self–signed certificate
- Get a signing cert from a CA

We chose the 3rd option, as we wanted to have 1 cert for signing for all of the federation servers in the farm. This gives us one cert (or cert chain) to give to a resource partner. From our own internal IT processes as well, we like this better as the token signing cert is the "keys to the kingdom", and using the same cert that we use for SSL in this case places the keys to the kingdom cert in the hands of more people than we want (we have to give it to our load balancer guys, for example).

I think ADFS by default does the second option. I think this is only appropriate for testing, but there is a school of thought that says that if you can get your partners to trust that cert, it doesn't matter. In that case, you never have to worry about expiration or CRL checking, as your cert probably won't expire in any practical sense and won't have a CDP defined. Still, I'd rather provide our partners with a certificate from a well–known trusted root CA that they are likely to trust already.

The first option is actually what MS used in their first internal deployment (according to Brian Puhl's presentation at TechEd 2006). This approach worked for and has the benefit of fewer certificates.

The spec is very flexible, as the cert used for signing just needs the digital signature key usage. It doesn't need any special EKUs at all (like server authentication or client authentication). As such, you can use an SSL cert, a code signing cert, a personal email cert, or just about anything. The downside of this is that you end up having more choices to make. :)

Regarding CRL checking, you want to leave this on if you can, but practically speaking, it won't always work. CRL retrieval is still quite imperfect, in that some CRLs are retrieved via HTTP, some via LDAP and some via other stuff. Depending on all sorts of things like firewall rules and

Re: ADFS Token–signing Certs Not in Trusted Root Store

proxies, the service process may not be able to get all of the CRLs in the chain, or may not finish before it times out. As such, you sometimes have to turn it off. Brian reported that they had to disable it for one of their partners because the partner's CRL was published at an internally facing URL. There was no practical way to change this and getting a different cert didn't make sense, so they just gave up in the interests of expediency.

Certificate revocation is an important PKI concept, but unfortunately, the process of determining whether a cert chain contains any revoked certs is still impractical to the point where CRL checking often must be discarded. There is no perfect solution to this problem (yet).

(this might make a good blog posting....)

Joe K.

—  
Joe Kaplan—MS MVP Directory Services Programming  
Co—author of "The .NET Developer's Guide to Directory Services Programming"  
<http://www.directoryprogramming.net>  
—

"Susieber" <Susieber@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message  
[news:B018915F-8794-4F6E-9DA8-9F9661C3AD91@xxxxxxxxxxxxxxxxxxxx](mailto:news:B018915F-8794-4F6E-9DA8-9F9661C3AD91@xxxxxxxxxxxxxxxxxxxx)

Joe,

I think I just realized something. During ADFS Setup, you have the option to let ADFS create the token–signing certificates. That's what I've always chosen, by default. And the certs that ADFS generates itself have no trusted root. However, I just ran an additional test and found that not having a trusted root for the TS certs doesn't seem to break the claimapp from the Step–by–Step Guide.

But you're saying that you don't choose that option and instead choose to create the certs via you CA? Interesting – having tried that yet.

I read somewhere that disabling CRL checking is a bad thing, but I'm guessing you know that already. :)

Susie

"Joe Kaplan" wrote:

I have not had this problem. Our token signing certs are issued from a CA that chains up to a standard Windows trusted root though.

How is your certificate issued?

Re: ADFS Token–signing Certs Not in Trusted Root Store

The thing we can never get to work is CRL verification. For whatever reason, we cannot get all of the CA's in the chain to verify properly, so we end up having to disable CRL verification in the trust policy manually.

Joe K.

--

Joe Kaplan–MS MVP Directory Services Programming  
Co–author of "The .NET Developer's Guide to Directory Services Programming"

<http://www.directoryprogramming.net>

--

"Susieber" <Susieber@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message  
<news:50E66143-35F8-42BD-A4FD-946EB77C65AD@xxxxxxxxxxxxxxxxxxxx>

Each time I run Active Directory Federation Services setup in one of our labs I run into this problem: the token–signing certificate created by ADFS setup on each federation server is not trusted by the root. In other words, if you run the ADFS mmc, right–click on Federation Service, click Properties, and click the View button, you'll see that the certificate is not trusted.

So each time I set up ADFS, I manually import this certificate into the Trusted Root Authorities store on each federation server.

Anyone else having to do this? Looks like a bug to me but I wanted to check here first.

Susie

Re: ADFS Token–signing Certs Not in Trusted Root Store