Single Sign-On and Cookies

Single Sign On (SSO) relies on cookies. Here are some FAQs about this relationship

**Single Sign On (SSO) and Cookies Frequently Asked Questions (FAQ)**

- #The basics
- #Checklist
- #Debugging this stuff
- #Debugging with SM50

**The basics**

SSO means, that you have to prove your authentication only once: to the authentication provider. Every following authentication to a SSO-enabled system will be transparent. Here, in this context SSO means that once you have authenticated to the Enterprise Portal (EP) you won’t have to re-authenticate yourself to other applications that are integrated in the portal.

For SAP EP, this is achieved with the use single sign on ticket or cookie, called Logon Ticket. This ticket created as part of the logon process and sent to your browser after successful authentication, and can then be sent from your browser to other applications. The name of the cookie is MYSAPSSO2.

**Checklist**

1. **Are you using FQDN?**

   FQDN means you should use fully qualified domain (FQDN) names *all the time* in other words, put all the “.” characters (I call this character a full stop, others call it a period, point, punkt etc) in your hostnames. So, if your portal is running on host superportal.internal.mycompany.com, then use the complete URL (http://superportal.internal.mycompany.com:port/irj) to start the portal. (Incidentally, the portal logon process makes use of the logon cookie. You can demonstrate this by logging on to your portal with the FQDN (http://superportal.internal.mycompany.com:port/irj) and then in the same browser session use a shortened URL (http://superportal:port/irj) and you should be prompted to logon again. Why? See below...)

   Why are FQDNs so important? The logon cookie is issued for the domain that you logged on to. By default, this means everything after the first full stop. Anyway, if you try to access a URL in the same domain, the cookie is sent, if it’s a different domain, it’s not sent. Simple really!

   And this is why you need to put FQDNs everywhere - especially in the hostnames in the system landscape for ITS, WAS, message servers etc. Also remember to use FQDNs for URLs in URL iViews.

   So, a couple of things become clear immediately, for example, all our hostnames must have the same number of full stops. If they don’t, we need to start making DNS entries to trick the systems - talk to your networks people about this. You might also think that we need to have everything after the first full stop consistent. This is the default, but by using domain relaxation we can overcome this. Try searching for ume.logon.security.relax_domain.level.

2. **Have you given your portal certificate to the backend systems?** This has been discussed many times, and can be found by searching for verify.der and STRUSTSSO2.

3. **Have you added an entry to the ACL table?** This is a follow on from the previous point. The certificate downloaded from the portal (verify.der) once uploaded into the backend system is only telling the backend system that this is a valid certificate to be trusted. You still need to specify which clients in the backend system can be accessed by this certificate. This is the ACL entry. The client number you enter when filling in this form has nothing to do with the client number you are logged on to! This is the client number that your portal is pretending to be and can be set through the UME parameter login.ticket_client (defaults to 000).

4. **Has your portal certificate expired?** By default it is only good for two years from the date the Java instance was installed. Look at the contents through System Administration -> System configuration -> Keystore, or double click the verify.der. If you’ve passed its expiration date, you’ve got problems and you need to rebuild the certificate and upload it again to your backend systems. If you’re lucky, you’ll get some messages in your backend system's system log as the expiration date approaches.

   Checking that your ticket certificates are consistent between the portal and the backend or use SSO2Wizard (/spnego, see SAP note 1083421 for more information)

**Debugging this stuff**

There is a tool available for download from the SAP Service Market Place as an attachment to note 701205 that allows you to see the contents of your logon cookie while logged on to the portal. Very useful. The note also describes how to debug the backend system by using SM50 and setting appropriate trace levels.

**Debugging with SM50**

The transaction SM50 is generally used to see how busy the work processes are in an ABAP instance. For SSO debugging, it has a few useful
features.

For example, it is possible to see the developer trace files for each work process, which often contain useful messages regarding SSO access. The developer traces can be viewed by selecting one of the work processes and using ctrl-shift-f8. If the trace files get too big, it is possible to delete one or more of them using the menu path Process -> Trace -> Reset. Make sure you don't want the existing trace information before you delete it!

Of course, you need to know which work process you're interested in! It will generally be one of the lower numbered ones, as the ABAP system tends to use these first. Sometimes you can see which process is giving a problem if there has been an error message generated in the system log (transaction SM21), otherwise it's trial and error. Please note that the first work process is 0.

Changing the trace level is useful for getting more detailed messages. Once again, you need to know which work process is likely to be used for the connection, but the good news is that you can select multiple work processes.

So, select a few processes and then use the menu path Process -> Trace -> Active components. When debugging SSO issues I normally set the level to 2 and activate only Security - leave all the others off.

Don't leave the trace level different from the default for too long, or else you will get a nastygram from your system administrator! More detailed traces cause more work for the system, which annoys the administrator.

Note that the work processes that have non-standard trace levels appear in a different colour - this is a reminder to reset them later on.

Anyway, now try to connect from the portal. You will start to see more useful messages, which will indicate further areas to investigate.

For example, the message:

**No entry in TWPSSO2ACL for SYS P7T and CLI 007.**

indicates that you haven't set the correct values in the ACL table, while

**Certificate not found.**

is fairly self explanatory - your portal's certificate isn't known to the SAP system.

Expired logon ticket messages will also appear, indicating the issuing date and time of the cookie plus its lifetime and the current system time. Check to make sure you haven't got a problem with timezones.