Shibboleth IdP v3

Information and Notes regarding the Shibboleth Identity Provider v3.x Upgrade

Shibboleth IdP v3 Upgrade						
Campus	**Current Version**	**Planning Stage**	**Upgrade Method**	**Consent**	**TOU**	**Go-Live**
UCOP	v 2.4.4	In Process	Parallel Build	no	no	September 2016
UCSC	v 3.1.2	Finished	Parallel Build	yes		Aug 2015
LBNL	v 3.2.0	Finished	Parallel Build	no	no	Dec 2015
UCSF	v 3.3.1	Finished	Parallel Build	no	no	May 2017
UCLA	v3.2.1	Finished	Parallel Build	no		June 2016

Feel free to reformat the notes section:

Campus	Problem	Solution

UCSC	Some UC Campuses have legacy SAML1 entityID names based on "urn:mace" We also use the SAML2 url standard for entities outside of InCommon, How do you apply a different entityID in the relying party?	Include the "responderId" parameter in the relying-party.xml override section.
		
		`<bean parent="RelyingParty" c:groupNames="urn:mace:incommon" p:responderId="urn:mace:uncommon:ucsc.edu">`
		
		`<property name="profileConfigurations">`
		`<list>`
		
		`<bean parent="Shibboleth.SSO" />`
		
		`<ref bean="SAML1.AttributeQuery" />`
		
		`<bean parent="SAML2.SSO" />`
		`</list>`
		`</property>`

LBNL	I wrote up a description of the mechanism we used to do a parallel upgrade, and specifically the testing. I sent that to the Shibboleth Users list. I later posted an edited, more English-like version on the InCommon wiki. In short, we used our BigIP load balancers and cookies to permit testers to select which version of the IdP they wanted to use. That way, we could leave the v2 IdP in production while testers tested their applications against v3. (As an ongoing benefit, we can individually access our IdP's now to troubleshoot individual servers, add staging servers into the production pool for final validation, roll backward or forward more readily, etc.) Other than the issue Jeffrey noted above, we had 3 issues: 1) SPs that did not support SHA-256 signing and/or encryption (covered in detail with useful examples here); 2) SPs that did not support encrypted assertions, because we had turned off encrypted assertions off for internal SPs in v2 in our earliest days for stupid reasons; and 3) a poorly converted attribute definition (more an error in understanding on my part than anything having to do with v3, though.) In the process, we also converted to a Docker-ized IdP. So far, we're pretty happy with that decision, though there was a substantial learning curve. I almost gave up in frustration at one point, but the fact that it took literally less than 10 minutes to upgrade from 3.1.2 to 3.2.0 made it all seem much more worthwhile.	
LBNL	Take note of the recent recommendations regarding the size of the InCommon metadata, now with the eduGain contents. Last week, we were bitten by this, as I had originally used the previously recommended 1GB (and increase from our 512M v2) max heap size, and we almost immediately went down. We're at 2GB now, and may increase it more.	