Consensus - Federated Logout

Points of consensus reached among participants in the first working call on this topic:

1. Some form of logout support by IdPs is strongly desirable, if not a must, given that most applications insist on offering logout links. If we could suppress those links, there would probably be support for making it optional, but participants representing SPs noted that customers expect and demand such links.
    a. IdPs that have a SAML logout endpoint but do not actually end a user's IdP session MUST return an error
2. Most applications tend to expect basic redirects for logout, but making that a profile requirement would imply additional specification work around exposing them in metadata and standardizing behavior. In lieu of that, the front-channel SAML profile is at least a standard and implemented in some form by most IdP software already. Back-channel was not deemed worth including.
    a. Commercial SPs by and large use redirects of course, but they are very unlikely candidates for this profile, so we're focusing on the requirements on services that would be metadata driven to begin with since that's going to be a key profile assumption anyway.
3. There was no support for mandating any particular **behavior** by the IdP other than accepting a request and responding. A range of organizational views dominates this space and the consensus was to explicitly leave the behavior up to the IdP.
    a. It follows that no SP can expect to regain control of the UI after a logout request and that we should be explicit about this.
    b. We should provide at least a general sense of what expected IdP outcomes might be, which we loosely characterized as "do nothing", "IdP only logout", and "single logout".
    c. The following broad strategies for handling classes of federated logout cases were identified:
        i. Do nothing: in this case, there's really no point in the IdP displaying a message to the user. IdPs that do nothing with logout reuqests should not publish logout endpoints in metadata. SPs should detect an IdP without logout endpoints and, after doing a local logout from the SP, in form the user that their IdP session could not be terminated.
        ii. IdP logout: if the IdP chooses to only terminate the IdP session, the user should be informed that they still may have actice sessions with other SPs.
        iii. Complete logout (single logout): If the organization running the IdP has some form of control over all federated SPs and can guarantee that they all support logout and are available to accept logout requests, the IdP can terminate all known SP sessions. The user should be informed that they've been logged out of everything. If possible, the user should be presented a list of the services from which they've been logged out.
        iv. Partial complete logout: if some of the SPs that the IdP federates with support logout, the IdP can log the user out of those SPs.
        v. Note that IdPs who choose options #3 or #4 might choose to first perform option #2 then give the user of proceeding to log out of more services rather than automatically logging the user out of all services. Alternatively, the IdP might choose option #1 with the option of option #3 or #4, thus not terminating the IdP session until the user chooses ot log out of everything. This latter choice, however, might allow the user to log back into the initiating SP without re-authenticating which would give the appearance of never logging out of that SP in the first place.
4. We agreed to remain silent on SOAP and stick with the HTTP-Redirect and/or the HTTP-POST binding.
5. An SP whose endpoints are based on multiple vhosts within a single entity descriptor should usually avoid SLO. A user who explicitly logs out of one vhost will not be logged out of **all** vhosts and often be unable to complete any subsequent SLO requests. Consequently, SLO works best for SPs with simple entity descriptors based on a single vhost.
6. SPs should be prepared to lose control of the UI in federated logout scenarios where the user's IdP supports an HTTP-redirect logout binding (sorry, why does the binding matter?)
7. When considering a SAML proxy, the proxy is more SP than IdP in role, and should conform to SP guidance in the profile
8. Deployers should be informed that some software will automatically issue SLO requests under some conditions, and that this should be considered if the SP doesn't intend to support the feature and has no endpoints in its metadata.
    a. In fact, an SP that issues a logout request but does not have a logout response endpoint in metadata is inviting an error conditon.