June 1 2012 Technical Call

Attendees

Arnie

Patrick

David Moldoff

Tim Cameron

Michael Gettes

Keith Hazelton

Nate Klingenstein

Karen Hanson

Dave Moldoff

Michael Morris

Tim McGraw

Doug Falk

Ann West

Khalil

Doug Shook USC

Agenda and Discussion

1. Brief report on marketing call from Ann, Arnie
    a. What do we need to set up the next phases? Business? Letters of Commitments
    b. SIgners of Letters of Commitments become the governance body
    c. Proof of concept by October PESC meeting
        i. Draft "rules of the road"
        ii. Proof of Concept
        iii. Plan for Pilot
    d. Before next meeting:
        i. Nice web site
        ii. Demo video
        iii. Technical PoC
        iv. "Rules of the Road" or "Rules of Participation"
2. Brief update on the last use case call
3. Probably lengthy report on action items for Nate (see his email to the lists from over the weekend). Everyone should try to read Nate's email and reference links prior to the call. (see below for notes)
4. Continued preparations for initial installations of the IdPs
5. Other??

From Nate:

Discussion of stateless clustering in Shibboleth

I don't have too much to add to this article, other than to say that I don't view the dropped features as crucial, and that some of the expertise available to the collaborative may be useful in crafting an authentication solution that uses very lightweight replication or client-side state.

As an aside, if we decided that SSO is not a desirable feature, then we would not need to do any clustering at all. That makes the implementation simpler. I believe we've decided that SSO is an important benefit of CommIT from the user's perspective, though.

2) Second major decision: If stateful, how do we share sessions?

a) Single VM on virtualized hardware: This is perhaps the most appealing of all the deployment options from my perspective. By the use of virtualized hardware, we do no clustering at the IdP or authentication layers at all, but instead rely on highly virtualized hardware in order to create one "super-VM". Upgrades can be handled by bringing up a second "super-VM" in parallel, and doing a DNS switchover, followed by disabling the first VM. Jeff reported on the call that they use virtualized hardware even across data centers, giving us that additional valuable redundancy. I know of Shibboleth deployments comfortably handling a million transactions or more per day using this approach, and I'm sure it can be scaled much further.
b) Terracotta: The "officially" supported clustering mechanism for this edition of the Shibboleth IdP, and no longer the officially supported clustering mechanism in the next. The developers regret the decision to use Terracotta for replication because it introduces a massive memory footprint(doubling or tripling the memory required) and is generally unmaintainable. I would recommend we not even investigate this option, truth be told.
c) Memcached: A contributor to the project has developed a plugin that allows the IdP to use memcached as a session store. This is a widely used and extremely scalable solution. There may be some concerns over using plugins that are not part of the core codebase and thus not officially maintained or supported. If we want to run the IdP in a highly distributed architecture, I would recommend the use of memcached for session replication(over SASL, of course).

https://wiki.shibboleth.net/confluence/display/SHIB2/Contributions#Contributions-IdentityProviderExtensions

As an aside, IdPv3 will have a different clustering mechanism, probably one we write that builds on a much lighter-weight framework. I don't believe this should impact our decision-making process, primarily because of timelines and the fact that we're just looking at proof of concept architectures, other than to say that if a memcached architecture is selected, it may eventually be migrated to an architecture relying on the built-in, officially supported clustering mechanism.

3) Third major decision: Should the IdP be hosted by one organization, or as a distributed mesh?

I would prefer that a single organization solely bear the responsibility of operating the IdP for simplicity of management, clarity of responsibility, and expeditious decision-making. This could be separated from(and, I think ideally would be separated from) data curation responsibilities and policy and general administration. These roles could in turn be beholden to a governance framework including a steering committee representing key stakeholders. I'm open to other thoughts, though.

So, in conclusion, I see three architectures that are interesting enough to warrant further investigation: stateless IdP's with a clustered authentication mechanism, a behemoth VM, or a memcached-based cluster of IdP's to perhaps be migrated to an officially supported solution at some point in the future. If there's no dissent, then we can leave it to project management to conscript further volunteers.

Thanks so much for your time and attention to these details,
Nate.

p.s. Here's a cursory, general introduction, included mostly for completeness:

https://wiki.shibboleth.net/confluence/display/SHIB2/IdPClusterIntro

Conversation:

Stateless gives benefits for reliability. Stateless arch sacrifices features

Potential for Single log-out. Likely to be the most significant

We can do investigations into the scalability and reliability of Stateful.

Giant monolithic VM

Terracotta - very heavy weigh. Not recommended. Negative impact on sustainability. Much more frail

Mem.cache - highly scalable, very distributed.

Does it matter "who" will be doing this? If we bring in a contractor, it may not matter what we do now. However, our choice of contractors may depend upon sanity checks we do based on our testing. We don't need to select one mech as an anointed way to do it. We should still do our own scale testing, perhaps even for testing. Toronto University has the largest example of a Shib IdP on monolithic VM. Patrick likes memcache, or membase(?).

But, do we need stateful? Perhaps test with stateless.

Discussion around starting with stateless or stateful for proof of concept.

Agreement for Stateless for Proof of Concept.

Single log-out is a business decision.

For now, we will pursue a stateless implementation. The IdP will be hosted by a single organization. We won't try to span multiple organizations. This means we do not have to choose between mem.cache and VM.

Action items

Corporate folks: Take a look at requirements for single log-out and report back to the group.

Arnie give David Moldoff details about 4 storyboards.

Ann and Arnie try to diagram who's on which team and post on wik with call information.