Coming SLAC Computing Changes the SDF landscape

These are some core changes to SLAC computing that projects are likely to encounter between 2022 and 2024 as SDF ramps up.

These are not official timelines and there's no official milestones for much of this, but it's my attempt at cataloging conditions that developers may need to be aware of.

Bullet cluster is deprecated, limited to no capacity for RHEL6 batch

Jobs targeting the bullet cluster/RHEL6 should be migrated soon. There would be no functional capacity for RHEL6 machines in LSF at this point.

Bubbles cluster likely to be absorbed under SDF in 2022; could be retired in 2024

On retirement of the bullet cluster, bubbles machines will represent the entire corpus of batch resources outside of SDF. Most of those machines themselves will hit 5 years of service in 2024.

When the bubbles machines are transferred to SDF or retired, whichever comes first, **there would functionally be no functional resources left managed under LSF**.

AFS is deprecated, unavailable in SDF

AFS paths are not available on SDF batch nodes. AFS will not be mountable to SDF batch nodes.

NFS paths are deprecated, unavailable in SDF

NFS paths are not available on SDF batch nodes.

Should the underlying data be available in some fashion (via GPFS, or copied over to the SDF file system) - mount paths MAY be emulated with containers at runtime.

IAM updates

The IAM system is slated for an overhaul, particularly in how identity is defined and managed at SLAC.

Information tied to your identity can include things about who you are, how you are related to Stanford/SLAC, your SLAC ID, your training information, FACTS, and information like that. This identity would be your identity at SLAC, and it would be stored by SLAC.

For SLAC/Stanford Staff, the SLAC identity is likely to be linked directly to your Stanford identity through the sunet id account. For users, it may be associated directly with an external identity (e.g. university account, Google, or ORCID, etc...)

Following from this, accounts in SDF are likely to be associated with some kind of central SLAC identity, though it is not clear if all information relevant to SDF would be stored by the SLAC IAM System. It is a possibility that SDF has it's own directory and user management system for information that's relevant to SDF in particular or more broadly unix computing in a system downstream of the SLAC IAM system. An example of that might be information like groups, and how that is translated/managed across both the web (usually through SAML/OAuth/OpenID Connect) and natively in platforms (LDAP/AD).

Web applications, for example, might be able to rely either on the SLAC IAM System or an SDF IAM System in that scenario. If there were two different IAM systems, they might have different policies around the release of information to web applications.

In light of expected changes, Crowd-only accounts are likely to be deprecated. This poses risk the the CAS server currently used at SLAC, as well as the two instances of the Group Manager application relying on it, and all Java web applications relying on CAS through the LoginFilter.

A SLAC IAM system would be expected to export a SAML interface at a minimum, and possibly an OpenID Connect or OAuth 2.0 interface. An SDF IAM system that relies on it would similarly also export one or both of those protocols. In all cases, there are applications such as dex which can consume SAML, OAuth 2, OpenID Connect, crowd, and other interfaces, while exporting an OpenID Connect interface.

uid and gid numbers may be subject to change

SDF uses Active Directory for user logins, but currently still relies on NIS groups from the unix system. This enables compatibility for mounting drives, but it is a technical legacy that may limit adoption of new IAM systems or security techniques.

uid and gid numbers might be subject to change as the IAM is updated and matures. If this forces a uid/gid update on disk storage, those disk would no longer be compatible with existing unix infrastructure.

sudo deprecated on servers in SDF

Should not expect to have sudo privileges on machines that need lustre mounted (e.g. servers for projects).

Users (admins/developers) should try to use unprivileged container technologies where possible.

container technologies

Singularity is currently available on SDF cluster. With a small amount of effort, docker images can be used by singularity, but the CLI and images are not directly compatible.

sudo deprecation has a related role here. Under vanilla installs of Docker, for example, it's usually recommended that users be added to the `docker` group, and by default they can effectively control a system through the use of privileged docker containers and mount paths. For services, there exists singularity-compose, but there is not feature parity.

There are a few new options towards unprivileged (rootless) containers, though singularity is the only solution in SDF batch nodes. That landscape will likely change over the next few years. podman is likely to emerge as a main part of that, but may need some changes which are fine on servers but might not be appropriate for batch machines.

kubernetes already exists at SLAC and SDF to some extent. Applications, especially web applications, would be encouraged to use it where possible.

RHEL 8, CentOS 7, and RHEL 7

Current SDF cluster uses CentOS 7. CentOS 7 is a community OS and does not typically get the same support of patches supported versions of RHEL get. It's likely CentOS 7 nodes will fail security audits past June 2024 (30 months) and at the very least need to be migrated to RHEL 7.

Lustre and Weka

WekaFS will be commissioned soon at SLAC (certainly it should be operational before June 2022). Weka will be able to replace parts of Lustre (`/sdf`) where performance is critical and/or in the case of small files. Use cases include home directories. Weka may be a good place to store software that is used concurrently by many batch jobs.