UC San Diego HR Payroll and IAM

Current UCSD Identity Management

Identities

New identities are created primarily through three core mainframe applications: payroll(PPS), financial(IFIS), and student(ISIS) systems. This data may be entered via terminal emulators connected directly to the mainframe or web front ends using screen scraping, web services, etc. Some of this data is fed via file extracts into our email and Active Directory provisioning systems. All of these identities are then synchronized and merged nightly into a relational DB schema we call affiliates_db. This nightly load job also attempts to join the identities with the email and Active Directory accounts which were created separately. For certain affiliate types which are not entered into the three core systems, data can be entered from a web front end (MyAffiliates) and saved directly into affiliates_db.

Identifiers

Each of the core mainframe systems has its own internal identifier which we store in affiliates_db for cross referencing. We have an internal primary key for each person as well as a table for mapping targeted IDs to our internal ID. Our targeted IDs are UUIDs and therefore not based on any other user attributes. UCNETIDs are also loaded into affiliates_db from a UCOP file dump. SSNs are used internally for matching but not exposed to the broader campus.

SSO

In order to receive a single sign-on account, employees and students must first self register using data from the payroll or student systems. For employees, this creates a mainframe (RACF) account and links it to their (hopefully) singular affiliates_db record. Students get a kerberos account instead of a mainframe account. Either account is separate from the Active Directory system.

Shibboleth

Our Shibboleth implementation allows users to log in either with their SSO account or with Active Directory using a tabbed interface. The available tabs can be customized depending on the needs of each application(s). This allows applications to integrate with a single API (shibboleth) and later change which authentication method(s) they support. However the user authenticates we use the associated affiliates_db record to resolve attributes.

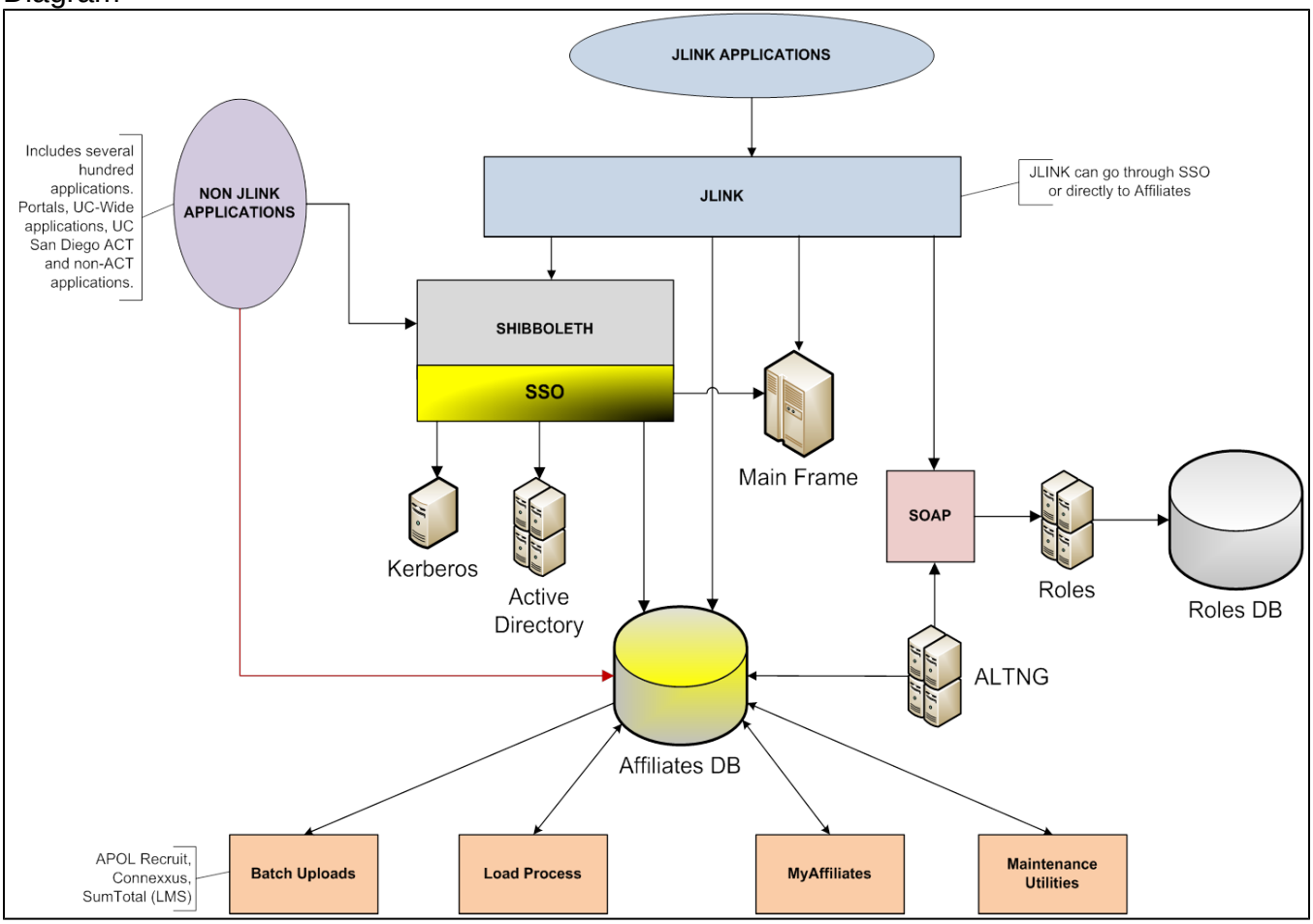
Other Provisioning

A handful of federated applications require periodic feed files for user provisioning and we have custom jobs designed to support these applications. Many local applications handle their own automated provisioning using data from our data warehouse. Other local applications provision access manually from within. Access to our core business applications, however, is provisioned through a central web front end (ALTNG) by officials designated in each department (DSAs). Department heads and their delegated DSAs are therefore responsible for all access within their own department. Access is subject to approval by the data stewards.

Enterprise Roles

Several campus wide roles have been identified which require common access provisioning across many applications. In order to improve efficiency and speed of provisioning we implemented a role based access model to store permissions which applications can consume for their own internal purposes. These enterprise roles are not in wide use yet as we have many legacy applications which would need to be rewritten to support them.

Diagram



Things we'd like to make better

1. Since data entry happens via independent applications which are completely unaware of IdM infrastructure, we have no ability to reconcile identity matches at the point of origin. Instead we rely on the sensitive SSN to match identities later. This results in a significant number of duplicate records or conflicts which must be resolved manually. In cases where we cannot collect SSN, there is little we can do to reliably match identities across systems. If the new data entry system allowed for matching across all identities we could potentially reduce the churn of bad data in the system.
2. Our current self-registration system relies on data which is not all that secret. InCommon Silver certification probably can't occur without changes to this credential issuance process. We need to issue a temporary credential to new employees at the moment their identity is vetted by an HR representative.
3. It would also be nice if we could get more timely updates for identities in the payroll system. Ideally we would receive new employee data quickly so that they can get credentials on their first day of work. In some cases we even need to issue credentials prior to the employee's start date.

Impact of the current plans on IdM

1. It's likely that our data warehouse group will continue to provide equivalent person data from the new payroll system which we can plug into our Load Process and continue on with the same model we have now. However, we'd like to take advantage of any update notifications to keep data more in sync with the payroll system, but this would take more effort.
2. We need to build out our SAML 2 endpoint(s) and ensure that the necessary attributes are resolved in order to communicate with Oracle's service provider.