Product Evaluation Questions

2. Reporting and Auditing

1. What current and historical data is maintained for reporting?
2. Is the reason (automatic and manually approved) for all provisioning decisions and actions stored? Can it be sent to an external system (Splunk, logstash, etc) data warehouse?
3. How is the data stored? Can it be read by external systems?
4. Can we export the data in its entirety?
5. Can we control how long the data is maintained?
6. Does this product provide what our auditors and compliance officers need?
7. Does this product provide what our application (target system) owners need?
8. Reporting (should we just combine audit and report?)
    a. What pre-built reports are available? Can they be customized?
    b. Can we build our own reports? Using a GUI? Non-GUI?
    c. Does the product support reports on
        i. Access for an application (target system)
        ii. All access for a user, all users in a unit, all users for a supervisor
        iii. Elevated or high-risk access
        iv. Separation of Duties
    d. What output formats are available for reports (eg, PDF, CSV, HTML)
    e. Is the data used for reports available for use by third-party reporting tools?
    f. Can reports be run on a schedule and sent by email or to a (possibly external) report repository, and/or made available via GUI? If available via GUI, what are the access controls?
9. Auditing
    a. Can we compare intended provisioning to the actual state of an application on demand?
    b. Does the product audit changes made within it (eg, who made a change to group membership logic when, and what the change was)
    c. Does the product support Separation of Duties audits?
    d. (If you do access reviews / attestations) does the product provide adequate support?
        i. review by person, unit, application
        ii. review of only manually-decided access, exceptions only, etc
        iii. Can audit results include “comments” (eg, “access being removed because …”) that become part of the record
        iv. Can the auditing work with an external ticketing system (eg, ServiceNow, Remedy)
        v. How does the product define and schedule reviews, notify and remind reviewers, etc? Can the product send emails and/or use an external ticketing system? Are reviews done within the product, or in a document sent to the reviewer?
        vi. How does the reviewer to report results? Is the effort required proportional to the number of changes?
        vii. Does the product support workflows, logic, etc. needed to implement access changes determined by a review?

1. Identity provisioning
    a. Identity matching
        i. Does your product provide an identity matching service?
            1. Describe how the identity matching service is configured, and any scoring or weighting of attributes.
            2. Describe how low quality matches are handled, and if there is a notion of matches in suspense, what are the mechanisms for making assertions about them.
            3. Can the matching service be run against an existing population seeking duplicates.
        ii. Does your product have the ability to use an external matching service?
            1. Describe the configuration of the external service.
            2. Describe how low quality matches indications are handled, and if there is a notion of matches in suspense, what are the mechanisms for making assertions about them.
            3. Describe and standards that are used in messaging or APIs for matching services.
    b. Username assignment
        i. Does your product support user selected usernames, if so, how are attempted duplicates handled.
        ii. Does your product support generated usernames, if so, describe the options and configuration.
        iii. Does your product support enrollment of new users, if so, please describe the configuration of the enrollment portal, and any support for workflow.
    c. Identifiers for services and target directories[1]
        i. Describe how your product handles identifiers or accounts which may be different from the institutional identifier.
        ii. Describe how accounts in other systems are provisioned, and any standards that are supported for provisioning.
        iii. Describe how accounts in other systems are deprovisioned, including supported workflows.
        iv. Describe your products support for deprovisioning identifiers, and any support for namespace preservation after deprovisioning.
    d. Username changes
        i. Describe how your product handles username changes, including support for namespace protection and auditing, and any workflows.
        ii. Describe how your product can communicate username changes to other systems that might need to be informed
    e. Social IDs
        i. Describe your products support for social IDs (Facebook, Google, etc.) in place of local identities[2] .
        ii. Describe your products support for social IDs that are connected to local identities.

Notes from a subsequent discussion on identity matching, need to work this in.

- where does a person start (eg, HR, or self-service from person registry)
- match methodology (scoring, etc), is match process in person registry, or one or more of the authoritative systems

- processes for handling dups, possible matches (in registry, systems of record, and possible downstream systems)
- HR, Admissions, IAM responsibilities
- are social ids involved? Do you need a match-social-id-to-person-in-registry process?
- do sponsored (loosely-affiliated) & contractors use the same processes?
- is an external identity service involved?

- Tom: Identity matching is a registry function - receiving person data from multiple sources. Do other institutions see it as a different function?
- Ethan - person registry is also a source of record at UNC
- Is the person registry/identity matching part of provisioning or something that happens before?
- One of the competencies that a person registry needs to have is identity matching, whether it's part of the provisioning infrastructure or upstream
- How centralized is the institution's ERP
- May want to survey BTAA about scoring systems and matching rules - there may be a variety
- Karen - incoming students will use a guest account in the person registry and will link it to their institutional identity
- Tom - Madison is thinking about models where there's a low bar for creating a profile (that has an institutional or social credential associated). When a user establishes a role (student enrollment, employment, etc), we can provide the role owner (registrar, HR) a mechanism to bind the role to the user's profile, or we can use the fact that the role was established via an authenticated session with the user's profile to bind the role to the user. (Tom and Jon have been pondering ways to "prove" the logged-in session, since we have historical issues with sources guessing at identifiers instead of resolving them in a login.)
- Identity matching is not 'one size fits all,' but may be part of the provisioning process. Need to add to the identity provisioning portion of the survey

---

5. Credential provisioning

a. Password rules and policies

i. Describe the password policies you support with regard to complexity, length, and any dictionary checks. Include character classes supported in complexity checks. (Call nist

ii. Does your product support flexible password policy based on password length? For example support pass phrases but requiring additional character sets for shorter passwords.

iii. Describe your support for passwords in multiple languages.

iv. Describe your products support for password expiration, including any support for flexible expiration based on grouping, assurance, or other factors such as password quality.

v. Describe how your product conveys password quality to end users.

vi. Describe how you product meets accessibility guidelines

b. Initial password setting (credential activation?, initial login?)

i. Describe how your product assures initial password setting is being done by the appropriate authority, such as invitations, one time and/or short lived tokens etc.

ii. Describe your products support for terms of use and informed consent when getting a credential.

iii. What platforms are supported for end user devices setting initial and subsequent passwords, including any required technologies.

iv. Describe any features your product has to deter attacks on unclaimed credentials.

v. Describe how your product handles multiple credential stores.

c. Assignment of additional authentication factors

i. Describe your support for certificate based authentication.

ii. Describe your support for multifactor enrollment, specifying supported technologies and products, explicitly address U2F support.

iii. Describe any support you have for challenge response questions.

iv. Describe any unlisted additional authentication factors, and any features that help user recognition such as image validation.

v. How do you handle loss of a (perhaps only) two factor device, such as one time tokens

d. Deprovisioning of credentials

i. Describe the states supported by your product for credentials, such as open, expired, disabled, locked/unlocked, security deny, etc.

ii. Describe any workflow available for deprovisioning, time based, approval based, and any attribute or membership checks that can be used for deprovisioning workflow.

iii. Describe any controls for sanity checks in your product to prevent accidental mass deprovisioning.

iv. Describe the administrative capabilities your product has for deprovisioning and deprovisioning intervention, include any delegation features.

v. Describe how your product handles deprovisioning of credentials w/r/t propagation to multiple credential stores.

In addition to the above, include documentation of your APIs related to all of the above functional areas.

6. Target Directories and Service Provisioning

a. Linking identities between directories or services
    i. Describe how your product links an identity in a source directory to the same identity in the target (and service?)
    ii. Are your user linkage attributes characterized as follows:
        1. Immutable
        2. Static
        3. Globally unique
    iii. What is the process of account matching if accounts already exist?
    iv. How flexible is customization of the IDM connector that provisions the account?
b. Communicating updates to target directories
    i. Describe the transport mechanism used for updating target directories and services
    ii. What protocols does your product support for provisioning of accounts (for example, SOAP/REST, LDAP, Messaging, JDBC)
    iii. What supported standards can your product use? (e.g., SCIM, LDIF, SPML, etc.)
    iv. Describe how your product batches or queues large quantities of updates.
c. Provisioning models: when to provision
    i. Describe how your product supports “Just-in-Time” provisioning model-- on demand provisioning when the user logs in.
    ii. Describe how you support the “Just-in-Case” provisioning model-- pre-provisioning accounts en masse
    iii. Workflow-based provisioning model
        1. Describe how your product handles automated workflows.
        2. Describe how your product handles manual intervention by an admin.
        3. Describe how your product supports end-user self-service workflows.
    iv. Do you support a threshold to alert for large quantity of updates?
d. Reconciliation
    i. How does your product ensure the target directory or service has state in sync with the source?
    ii. Does your product support rollback or transaction?
e. Fine-grained authorization
    i. Describe what authorization sources your product supports (e.g., Grouper, LDAP, Active Directory)
    ii. Is the same mechanism for account provisioning used for authorization provisioning?
    iii. What protocols for transmitting authorization does your product support? (e.g., Messaging, SOAP/REST, LDAP, JDBC)
    iv. What supported standards can your product use for authorization? (e.g., SCIM, LDIF, SPML, etc.)
    v. Does your product allow support for custom or proprietary interfaces for authorization?
    vi. How does your product links or maps internal groups and roles to external service-level fine-grained authorizations?
f. Deprovisioning and repatriation
    i. Describe how your product triggers deprovisioning of identities in a target directory or service.
    ii. Describe the process of deprovisioning identities in a target directory or service.
    iii. How is authorization removal handled for deprovisioned users?
    iv. Describe how your product supports repatriating a service account from institutional to personal.
    v. Do you support a threshold to alert for large quantity of changes?

8. Groups and roles

a. Types of groups
    i. Describe how your product supports a list of definable groups.
    ii. Describe how your product supports a hierarchy of groups (i.e., nesting and relationships between groups)
        1. What entities can be members of groups?
    iii. ?? What upstream data sources does your product readily support?
    iv. Do you support sets of groups associated together? (i.e., base, exceptions, includes/excludes)
b. Administration
    i. Describe delegated access administration features for group management.
    ii. How does your product deal with “orphaned” delegation? (When previous admins are no longer there.)
    iii. Do you provide APIs that would allow an external group and access management tool to drive your product’s groups and group memberships
    iv. Do you support attribute-based (ABAC) or role-based (RBAC) concepts to drive groups and group membership?
    v. Can groups have permissions associated with them?
    vi. What sort of attributes or metadata about groups are available?
    vii. Does your product support automatic review of roles/groups (attestation)?
c. Guidance for architecting
    i. How does your product expose or link groups or roles for fine-grained service authorizations?
    ii. How do you support Attribute-based access control?
    iii. How do you support Role-based access control?
        1. Are roles managed within the product?
    iv. How does your product define a default role or template (set of groups) for new entities?
    v. How are groups updated/kept in sync?
    vi. Describe synchronization mechanisms, i.e., changelog vs. full sync

10. Product Cost/Vendor Considerations

a. Licence

i. What is your Software licensing cost structure (Enterprise vs non)?
ii. If one of your license model is pay-per-active-account , how do you consider the following populations? :
    1. Alumni users
    2. Guest users
    3. Extended Community users (Parents, Propsect Students , Applicants, Continuing Ed students ,ec..)
    4. Social identities that are linked to Idm system
iii. Do you provide any Higher Ed discount ?

b. Vendor Support and Maintenance (On going)
    i. What is your on-goin service support contract structure ?

c. Vendor Stability
    i. How long is your product being in the market ?
    ii. How many Higher Ed clients do you have ?