UNIVERSITY OF ARKANSAS SYSTEM

SOFTWARE DEMONSTRATION

IDENTITY AND ACCESS MANAGEMENT SOFTWARE AND SERVICES

RFP 644456

DEMONSTRATION AGENDA

Executive Summary	(30 minutes)
Technical Overview	(30 minutes)
Break	(15 minutes)
User Interfaces and Experience	(60 minutes)
Multi-Campus and Inter-Campus Features	(60 minutes)
Lunch	(60 minutes)
Identity and Attribute Management	(90 minutes)
Role management	(45 minutes)
Break	(15 minutes)
Authentication	(60 minutes)
Federation and single sign-on	(60 minutes)

INSTRUCTIONS TO PRESENTERS

The software demonstration is a supplementary element to the vendor's initial RFP response. The evaluation team has prepared this demonstration outline to serve as a guide to presenters.

Presenters should follow the high-level agenda closely. Presenters should regard specific items in the outline as a general guide to the presentation, not as specific bullet points to be addressed in order. Presenters are encouraged to address multiple points in a single demonstration element, especially if doing so adds clarity or reduces presentation time. Presenters may reference items from latter portions of the agenda earlier, but those topics should still be addressed within the appropriate agenda topic.

Presenters should view the durations for each agenda item as hard deadlines. The evaluation team may elect to extend time for a topic, but presenters must not rely on this. Presenters should build in a small amount of time at the end of each high-level agenda item to address questions that arise during the presentation.

SOFTWARE DEMONSTRATION OUTLINE

EXECUTIVE SUMMARY (30 MINUTES)

1. Introductions
2. Vendor Highlights and Product Overview

TECHNICAL OVERVIEW (30 MINUTES)

3. Provide an overview of the solution and how it addresses the needs of the University System, including the stated Service Level Agreement Requirements.
   a. Describe hosting and staffing arrangements. Demonstrate 24x7 oversight and capacity to comply with the SLA.
   b. Describe software lifecycle and upgrades (frequency, technical vs. functional, relation to product release cycle). Describe the proposed approach to staying within state standard for currency of software versions.
      i. Describe change management procedure
      ii. Describe under what circumstances will the application (or any modules) be unavailable due to known/planned maintenance of the Application or any supporting hardware or infrastructure
   c. Describe the support and help desk structure and procedures.
      i. What triggers escalation
      ii. Communication process
      iii. Incident reports
4. Discuss the creation, assignment, and maintenance of internal application roles and privileges.
5. Illustrate how logging and auditing is performed, including integration with external SIEM tools.
   a. Logging of IAM-centric administrative actions
   b. Logging of user-centric actions (login success/failure, etc.)
6. Describe what users will experience in various failure modes:

a. Primary data center becomes unavailable
b. Imminent failure is detected
c. Workloads exceed system capacity

BREAK (15 MINUTES)

USER INTERFACES AND EXPERIENCE (60 MINUTES)

7. Illustrate how the software's theming and branding can be adapted to the needs of the several members of the UA System.
   a. Demonstrate a change in style or graphic element.
   b. Demonstrate the ability to change the wording of messages to end users, such as email messages, text (SMS) messages, application content, etc.
8. Demonstrate the mobile experience for both end users and administrators. If a mobile application is available, demonstrate both the application and the mobile browser experience.
9. Demonstrate the ability to preview the effects of a change before they are committed. For example, if the definition of a role changes, allow the administrator to preview how that change will affect users.
10. Demonstrate the ability to delegate one's authority to another user, such as when one is on vacation or absent.
11. Illustrate and discuss supported integration and connector protocols, such as custom APIs, LDAP, SAML, SCIM, etc.
12. Illustrate integration with proposed tools and services, such as:
    a. Help Desk/ITSM software (ServiceNow, Cherwell, etc.)
    b. ERP software (Oracle Cloud, WorkDay, Banner)
    c. Directories (Microsoft Active Directory, LDAP)
    d. Endpoint and network security tools (VPN, NAC, etc. especially Palo Alto)
    e. ID Cards, Smart Cards and Building Access Control
    f. "Internet of Things" (IoT) devices (kiosks, clickers, beacons, consumer IoT, etc.)
    g. Biometrics
13. End-user self-service. Demonstrate how end users…
    a. pick up their account for first use.
    b. change their password.
    c. reset a forgotten password.
    d. enroll two-factor authentication.
    e. manage their identity and its attributes.
14. Illustrate how workflows may be triggered both manually, and automatically (i.e. via rules). Illustrate a workflow with manual steps.
15. Illustrate a workflow that requires multiple approvers in sequence (A and B), and a workflow that requires multiple approvers from a pool (3 of 5 people must approve).

MULTI-CAMPUS AND INTER-CAMPUS FEATURES (60 MINUTES)

16. Illustrate how multiple campuses are defined within the software. Describe how campuses relate to each other, particularly any parts of the software that apply globally (to all campuses) as compared to those parts which apply to only one campus.
17. Demonstrate adding an additional campus and how it interacts with the others (if it does so).
18. Illustrate how a campus may be further divided into smaller units (e.g. college, department, etc.), and how administrative authority for those smaller units is delegated and managed.
19. Demonstrate adding and delegating a smaller administrative unit within a campus.
20. Illustrate and demonstrate how each campus's unique culture and school spirit may be preserved, even though all campuses will use the same software. Note that this prompt is not at its heart a question about branding (addressed separately); this is about making each student (and employee) feel that they are a part of the campus community where they reside.
21. Demonstrate the transfer of an individual from one campus to another (e.g. student transfers from 2-year to 4-year campus, and/or staff transfers from one campus to another).
22. Demonstrate enabling an individual to take classes at multiple campuses.
23. Demonstrate enabling an individual to work at one campus while taking classes at another.
24. Demonstrate enabling an individual to work and take classes at the same campus (assume the individual has elevated privileges as an employee that a student would not ordinarily have).
25. Demonstrate the authorization model enabling a class to accept individuals from multiple campuses.

LUNCH (60 MINUTES)

IDENTITY AND ATTRIBUTE MANAGEMENT 90 MINUTES)

26. Demonstrate how identities are created, including how identity information can come from multiple external sources.
27. Demonstrate how entity resolution is performed. That is, for a set of similar records, how does the software decide if the two (or more) records refer to the same person, or different people? How are records merged or split when necessary?
28. Demonstrate how typically static identity information, such as account name ('uid') may be changed when necessary. Describe the impact of such changes and highlight any limitations of the software regarding this capability.
29. Demonstrate how individuals are retired or removed from the registry. Can an offboarded/deprovisioned individual return to UASys and reclaim her old identity? When/how are identities permanently deleted?
30. Describe and demonstrate the impact of role changes on identity life cycle. How does holding both student and staff role/affiliation impact identity life cycle?
31. Demonstrate provisioning a user account in the following external systems:
    a. Active Directory
    b. ERP (i.e. end user access)
    c. A legacy tool (i.e. something lacking modern provisioning interfaces)
32. Demonstrate deprovisioning at least one of the previously provisioned accounts.
33. Illustrate a workflow for vetting an individual prior to receiving their identity for the first time.

34. Illustrate how more rigorous identity vetting may be performed to meet the needs of InCommon SILVER or NIST SP 800-63.
35. Demonstrate the approach to managing privileged identities and access and other shared access models, if included in the proposed solution.

ROLE MANAGEMENT 45 MINUTES)

36. Illustrate the how guest users and other short-term users are created and how roles are assigned. (e.g. vendors. conference attendees, summer camps, etc.)
37. Illustrate the ability to discover roles in existing use, absent actual Role-Based Access Controls.
38. Illustrate the definition of a hierarchy of roles, from system to campus, to business unit. Describe how roles can provide uniformity at higher levels (e.g. a Student is a Student at all campuses), while allowing for local variance when needed.
39. Illustrate how roles enforce separation of duties. That is, how do roles apply to an individual when conflicting entitlements or privileges would be granted? Highlight how separation of duties works with nested roles.
40. Demonstrate how an individual’s role may be changed, both manually, and in an automated fashion. Provide an overview of any rule engine or similar technology used to effect changes.

BREAK (15 MINUTES)

AUTHENTICATION (60 MINUTES)

41. Demonstrate how a credential can be revoked if it is deemed compromised (see also 48 below).
42. Demonstrate how a credential can be reset if the owner forgets it (see also 13.c above).
43. Illustrate and demonstrate how password quality can be enforced:
    a. Illustrate varying complexity requirements, possibly varying by attribute, role, group, or business unit.
    b. Illustrate the ability to reject known-weak passwords (e.g. the top 10,000 most common passwords, or campus-specific watch-words.
    c. Demonstrate the ability to compute the strength or complexity of a proposed password, and that such computation has meaning relative to defined complexity requirements.
    d. Illustrate how a password complexity score can inform the risk score for an individual.
    e. Illustrate how a password complexity score can be used to influence the password aging formula (i.e. better passwords live longer).
44. Illustrate and demonstrate how password aging is implemented, including how users are notified (see also 7.b above).
45. Illustrate how passwords are synchronized between the software and authentication providers such as Active Directory.
46. Illustrate and demonstrate how temporary account isolation processes may be applied for various purposes (e.g. infringing behavior, account compromise, administrative suspension):
    a. Isolate (lock) the account to prevent the actual user’s continued use.
    b. Isolate the account to prevent use by an unauthorized user.
    c. Place multiple locks on a single account; all must be removed to re-enable the account.
    d. Lock-out a single role (employee), while allowing other roles (student) to proceed.

   e. Lock-out an individual at one campus, but not others.

47. Illustrate the various Multi-Factor (MFA) credentials the software supports. (e.g. SMS, mobile app, key fob, NFC token, FIDO, biometrics, etc.). Be sure to specify if the demoed MFA technology is included in the bid or a separate option, part of the software or a third-party solution.
48. Demonstrate how an MFA credential can be revoked if it is believed stolen or compromised.
49. Illustrate how MFA can be optional or required based on various risk factors, such as:
    a. Based on an individual’s attributes or roles (i.e. individual risk score)
    b. Based on sensitive applications or data (i.e. data risk score)
    c. Based on a user’s transition from typical to sensitive activity (i.e. step-up authentication)

FEDERATION AND SINGLE SIGN-ON (60 MINUTES)

50. Demonstrate adding (connecting to) a federation, such as InCommon
51. Demonstrate connecting to a new service advertised within a subscribed federation.
52. Demonstrate accepting logins from another identity provider (IdP) within a subscribed federation.
53. Demonstrate accepting logins from social identity (i.e. Google, Facebook, etc.)
54. Illustrate how local applications and services (SAML service providers) may participate in SSO, even though not exposed to an external federation. (Alternatively, illustrate the creation of a local federation that permits campus-level apps to participate in SSO).
55. For a locally-created federation, illustrate how SAML metadata is maintained (e.g. single file, DNS, etc.)
56. Demonstrate the use an IdP discovery service, especially if each campus will expose its own IdP instance.