

**Identity and Access Management  
Access Services 2.0 Increment 5  
Single Sign-On – Internal  
System Design Document**



**Department of Veterans Affairs**

**March 2015**

**Version 2.7**

## Revision History

Date	Version	Description	Author
04/17/2015	1.1	Updated per anomalies	Insignia
03/27/2015	1.0	Updates for AcS 2.0 Increment 5	Insignia

## Artifact Rationale

The System Design Document (SDD) is a dual-use document that provides the conceptual design as well as the as-built design. This document will be updated as the product is built, to reflect the as-built product. Per the Project Management Accountability System (PMAS) Guide, the SDD as a conceptual design is required prior to the Milestone 1 Review. (Sections 1, 2, 3, 4, 5, 7, 9 need to be populated, as applicable.) The as-built design for each delivery must be incorporated prior to the Milestone 2 Review. (The entire document needs to be populated or updated, as applicable.)

## Table of Contents

<b>1. Introduction .....</b>	<b>1</b>
1.1. Purpose of the SDD .....	1
1.2. Identification .....	2
1.3. Scope .....	3
1.3.1. Increment 5 SSOi Scope .....	3
1.4. Constraining Policies, Directives and Procedures .....	4
1.5. User Characteristics .....	6
1.6. Relationship to Other Documents and Plans .....	6
1.7. Definitions, Acronyms, and Abbreviations .....	6
1.8. References .....	6
<b>2. Background .....</b>	<b>6</b>
2.1. Overview of the System .....	7
2.2. Overview of the Business Process .....	7
2.3. Business Benefits .....	7
2.4. Assumptions and Constraints .....	7
2.4.1. Design Assumptions .....	7
2.4.2. Design Constraints .....	8
2.4.3. Design Trade-offs .....	9
2.5. Overview of the Significant Requirements .....	9
2.5.1. Overview of Significant Functional Requirements .....	10
2.5.2. Overview of Functional Workload / Performance Requirements .....	15
2.5.3. Overview of Operational Requirements .....	16
2.5.4. Overview of the Technical Requirements .....	17
2.5.5. Overview of the Security or Privacy Requirements .....	17
2.5.6. Overview of System Criticality and High Availability Requirements .....	17
2.5.7. Single Sign-on Requirement .....	18
2.5.8. Requirement for Use of Enterprise Portals .....	18
2.5.9. Special Device Requirements .....	18
2.6. Legacy System Retirement .....	18
<b>3. Conceptual Design .....</b>	<b>18</b>
3.1. Conceptual Application Design .....	18
Application Context .....	18
3.1.1. 18 .....	
3.1.2. High-Level Application Design .....	22
3.1.3. Application Locations .....	25
3.2. Conceptual Data Design .....	26
3.2.1. Project Conceptual Data Model .....	26

3.2.2.	Database Information .....	27
3.2.3.	User Interface Data Mapping .....	30
3.2.3.1.	SSOi Screen Interface .....	30
3.2.3.1.1.	IAM CentralLogin.....	30
3.2.3.1.2.	Application Access Denied Page .....	31
3.2.3.1.3.	Failed Login Page .....	32
3.2.3.1.4.	Session Timeout Page .....	33
3.2.3.1.5.	SSOi Authenticated Landing Page.....	34
3.2.3.1.6.	IAM LoggedOff Page .....	35
3.3.	Conceptual Infrastructure Design .....	36
3.3.1.	System Criticality and High Availability.....	36
3.3.2.	Special Technology .....	36
3.3.3.	Technology Locations.....	36
3.3.4.	Conceptual Infrastructure Diagram.....	36
3.3.4.1.	Location of Environments and External Interfaces .....	39
3.3.4.2.	Conceptual Production String Diagram .....	43
4.	System Architecture .....	44
4.1.	Hardware Architecture .....	44
4.2.	Software Architecture.....	55
4.3.	Network Architecture.....	62
4.3.1.	Communication Channel Security.....	63
4.3.2.	AcS Intercomponent Communications.....	64
4.4.	Service Oriented Architecture / ESS .....	65
4.5.	Enterprise Architecture .....	65
5.	Data Design .....	66
5.1.	DBMS Files .....	66
5.2.	Non-DBMS Files .....	66
5.3.	Data View.....	66
6.	Detailed Design .....	66
6.1.	Hardware Detailed Design.....	66
6.2.	Software Detailed Design.....	67
6.2.1.	Conceptual Design .....	67
6.2.1.1.	SSOi Support for LOA 2/3 External Users.....	74
6.2.1.2.	SSOi Support for LOA 4 SAML Token/Holder of Key (HOK) .....	76
6.2.1.3.	SSOi Mobility Support.....	76
6.2.1.4.	Federation IdP and SP for Internal Users.....	79
6.2.1.5.	WS Federation for Internal Users.....	83
6.2.1.6.	SSOi Support for Attribute Service.....	85
6.2.1.7.	SSOi Proxy Authentication Request.....	87
6.2.1.8.	Session Management .....	88
6.2.1.9.	SSOi STS Architecture Flow .....	88
6.2.1.10.	Centralized Login Page .....	89
6.2.1.11.	OAuth.....	95

6.2.1.12.	Product Perspective .....	96
6.2.1.12.1.	User Interfaces .....	96
6.2.1.12.2.	Hardware Interfaces .....	96
6.2.1.12.3.	Software Interfaces .....	96
6.2.1.12.4.	Communications Interfaces.....	96
6.2.1.12.5.	Memory Constraints.....	102
6.2.1.12.6.	Special Operations .....	102
6.2.1.13.	Product Features.....	102
6.2.1.14.	User Characteristics .....	103
6.2.1.15.	Dependencies and Constraints .....	103
<b>6.2.2.</b>	<b>Specific Requirements .....</b>	<b>103</b>
6.2.2.1.	Database Repository .....	104
6.2.2.2.	System Features.....	104
6.2.2.3.	Design Element Tables.....	104
6.2.2.3.1.	Routines (Entry Points).....	104
6.2.2.3.2.	Templates.....	104
6.2.2.3.3.	Bulletins .....	104
6.2.2.3.4.	Data Entries Affected by the Design.....	104
6.2.2.3.5.	Unique Record(s) .....	104
6.2.2.3.6.	File or Global Size Changes.....	104
6.2.2.3.7.	Mail Groups .....	104
6.2.2.3.8.	Security Keys .....	104
6.2.2.3.9.	Options .....	108
6.2.2.3.10.	Protocols.....	108
6.2.2.3.11.	Remote Procedure Call (RPC) .....	108
6.2.2.3.12.	Constants Defined in Interface.....	108
6.2.2.3.13.	Variables Defined in Interface .....	108
6.2.2.3.14.	Types Defined in Interface.....	108
6.2.2.3.15.	GUI .....	109
6.2.2.3.16.	GUI Classes.....	109
6.2.2.3.17.	Current Form.....	109
6.2.2.3.18.	Modified Form .....	109
6.2.2.3.19.	Components on Form.....	109
6.2.2.3.20.	Events.....	109
6.2.2.3.21.	Methods.....	109
6.2.2.3.22.	Special References .....	109
6.2.2.3.23.	Class Events .....	109
6.2.2.3.24.	Class Methods .....	109
6.2.2.3.25.	Class Properties.....	109
6.2.2.3.26.	Uses Clause .....	109
6.2.2.3.27.	Forms .....	109
6.2.2.3.28.	Functions.....	109
6.2.2.3.29.	Dialog.....	109
6.2.2.3.30.	Help Frame.....	109
6.2.2.3.31.	HL7 Application Parameter .....	110
6.2.2.3.32.	HL7 Logical Link.....	110
6.2.2.3.33.	COTS Interface .....	110
<b>6.3.</b>	<b>Network Detailed Design.....</b>	<b>110</b>
<b>6.4.</b>	<b>Service Oriented Architecture / ESS Detailed Design .....</b>	<b>110</b>

<b>Service Description for &lt;Consumed Service Name&gt;</b> .....	<b>111</b>
<b>6.4.1.</b>	<b>111</b>
<b>6.4.2. Service Design for &lt;Provided Service Name&gt;</b> .....	<b>111</b>
6.4.2.1. Introduction.....	111
6.4.2.1.1. Purpose and Scope of Service .....	111
6.4.2.1.2. Links to Other Documents .....	111
6.4.2.2. Service Details.....	112
6.4.2.2.1. Service Identification .....	112
6.4.2.2.2. Service Versions .....	112
6.4.2.2.3. Summary of Design and Platform Details .....	112
6.4.2.2.3.1. SOA Pattern(s) Implemented .....	112
6.4.2.2.3.2. COTS Platform vendor names and versions for hosting platform.....	112
6.4.2.3. Dependencies.....	112
6.4.2.4. Service Design Details.....	112
6.4.2.4.1. Interface Technical Specs .....	112
6.4.2.4.1.1. Service Invocation Type .....	112
6.4.2.4.1.2. Service Interface Type .....	112
6.4.2.4.1.3. Service Name .....	112
6.4.2.4.1.4. Interface .....	112
6.4.2.4.1.5. End Points .....	112
6.4.2.4.1.6. Operations or Methods.....	112
6.4.2.4.1.7. Message Schemas .....	113
6.4.2.4.2. Information Model .....	113
6.4.2.4.2.1. Class Diagram and Description of Entities Involved.....	113
6.4.2.4.2.2. Mappings from ELDM to Standards Based Schemas.....	113
6.4.2.4.3. Behavior Model (AKA Use Case Realization) .....	113
6.4.2.4.3.1. Use Cases (Use Case Model).....	113
6.4.2.4.3.2. Interaction Diagrams.....	113
6.4.2.5. Gap Analysis .....	113
6.4.2.5.1. Variances from Enterprise Target Architecture .....	113
6.4.2.5.2. Variances from SLDs.....	114
6.4.2.5.3. Variances from Standards and Policies.....	114
6.4.2.5.4. Justification for Exceptions and Mitigation .....	114
<b>7. External System Interface Design</b> .....	<b>114</b>
<b>7.1. Interface Architecture</b> .....	<b>114</b>
<b>7.2. Interface Detailed Design</b> .....	<b>114</b>
<b>WebAgent Pattern</b> .....	<b>114</b>
<b>7.2.1.</b>	<b>114</b>
<b>7.2.2. Federation Pattern</b> .....	<b>115</b>
7.2.2.1. Identity Provider (IdP) – With a Valid Session Cookie .....	117
7.2.2.2. Service Provider (SP) .....	117
<b>8. Human-Machine Interface</b> .....	<b>118</b>
<b>8.1. Interface Design Rules</b> .....	<b>118</b>
<b>8.2. Inputs</b> .....	<b>120</b>
<b>8.3. Outputs</b> .....	<b>120</b>
<b>8.4. Navigation Hierarchy</b> .....	<b>120</b>

8.5. Maintenance Pages.....	125
<b>9. Security and Privacy.....</b>	<b>128</b>
9.1. Security.....	128
9.1.1. Data in Motion .....	128
9.1.2. Data at Rest .....	129
9.2. Privacy .....	129
9.2.1. Confidentiality of Sensitive Information .....	130
9.2.2. Privacy of Personal Information .....	130
9.2.3. Process Integrity.....	130
9.2.4. System Availability .....	130
<b>Attachment A – Approval Signatures .....</b>	<b>131</b>
<b>A. Additional Information.....</b>	<b>132</b>
A.1. RTM.....	132
A.2. Packaging and Installation.....	132
A.3. Design Metrics .....	132
A.4. Acronym List and Glossary .....	132
A.5. Required Technical Documents .....	132
A.6. Responses to Produce WS Security Headers .....	133
A.7. Responses to XML Encryptions, Decryptions, and Digital Signature.....	134

## List of Tables

Table 1: Scope Inclusions .....	3
Table 2: Scope Exclusion.....	3
Table 3: Policies, Directives, and Procedures .....	4
Table 4: Assumptions .....	8
Table 5: High-Level Requirements.....	10
Table 6: Functional Requirements .....	10
Table 7: Workload and Performance Requirements .....	16
Table 8: Service Availability Level 4 .....	17
Table 9: SSOi Application Context Description.....	20
Table 10: Activities in the High-Level Application Design .....	23
Table 11: Objects in the High Level Application Design .....	24
Table 12: SSOi Solution Application Locations .....	25
Table 13: Application Users .....	26
Table 14: Database Inventory .....	27
Table 15: IAM Central Login Screen Description.....	31
Table 16: Hardware Appliance .....	45
Table 17: Virtual Machines and Appliances .....	46
Table 18: Software Components.....	57
Table 19: Programming Languages .....	62
Table 20: Database File System.....	66
Table 21: Windows Authentication, User ID/Password Authentication, and Centralized Logon Page with PIV Authentication .....	71
Table 22: SSOi Support for LOA 2/3 External Users.....	74
Table 23: VAAFI IdP SAML Integration .....	75
Table 24: Access Mobile Application Using Native Apps .....	77
Table 25: Application Protected by Separate IdP (Other than SSOi) .....	80
Table 26: WS Federation for Internal Users .....	84
Table 27: SSOi Support for Attribute Service .....	85
Table 28: SSOi Proxy Authentication Request.....	87
Table 29: Port Communications and Protocols.....	96
Table 30: SSOi Products .....	102
Table 31: Pre-Production PKI Certificate List.....	104
Table 32: Production Server PKI Certificate List .....	106
Table 33: SSOi Support for Secure Token Service.....	111
Table 34: Data Points and Security.....	129
Table 35: Responses to Produce WS Security Headers .....	133
Table 36: Responses to XML Encryptions, Decryptions, and Digital Signature.....	134

## List of Tables

Figure 1: SSOi Context Diagram.....	20
Figure 2: AcS 2.0 Application Design .....	22
Figure 3: ACS SSOi High Level Design .....	23
Figure 4: AcS 2.0 Conceptual Data Mode .....	27
Figure 5: IAM Centralized Login Page.....	31
Figure 6: Application Access Denied Page.....	32
Figure 7: Failed Login Page.....	33
Figure 8: IAM SSO Session Time Out Page.....	34



Figure 9: SSOi Authenticated Landing Page .....	35
Figure 10: IAM Logged Off Page.....	36
Figure 11: AcS Production Environments .....	37
Figure 12: Sample Conceptual Networks and Environments .....	40
Figure 13: AcS Production Environments .....	41
Figure 14: Logical Network String Diagram .....	43
Figure 15: Network Communication Architecture .....	44
Figure 16: Software Architecture.....	56
Figure 17: AcS Network Security Topology .....	63
Figure 18: High-Level SSOi Service-Oriented Architecture.....	65
Figure 19: SSOi Detailed Design.....	67
Figure 20: SSOi Centralized Logon Page with Windows Authentication Sequence Diagram.....	69
Figure 21: Centralized Logon Page with PIV Authentication Sequence Diagram .....	70
Figure 22: Centralized PIV-Only Logon with PIV Authentication Sequence Diagram .....	70
Figure 23: SSOi Support for LOA 2/3 External Users Sequence Diagram .....	74
Figure 24: SSOi Mobility Support Sequence Diagram.....	76
Figure 25: Access Mobile Application Using Native Apps Sequence Diagram.....	77
Figure 26: Federation IdP and SP for Internal Users .....	79
Figure 27: Application Protected by Separate IdP (Other than SSOi) Sequence Diagram.....	80
Figure 28: WS Federation for Internal Users Sequence Diagram.....	83
Figure 29: SSOi Support for Attribute Service Sequence Diagram.....	85
Figure 30: SSOi Proxy Authentication Request Sequence Diagram .....	87
Figure 31: SSOi STS Architecture Diagram.....	89
Figure 32: Centralized Logon Page Flow .....	90
Figure 33: Centralized Logon Page Error Handling Flow .....	92
Figure 34: Centralized Login Page Supported Partial and Complete Logoff Capabilities .....	93
Figure 35: Centralized Logon Page – Logoff Flows.....	94
Figure 36: SiteMinder Policy Architecture for Core Centralized Authentication Flows.....	95
Figure 37: Secure Token Service .....	110
Figure 38: WebAgent Integration Pattern.....	114
Figure 39: SSOi as Identity provider .....	116
Figure 40: SSOi as Service Provider.....	118
Figure 41: Central Login Application Page Hierarchy (1 of 7) .....	121
Figure 42: Central Login Application Page Hierarchy (2 of 7) .....	122
Figure 43: Central Login Application Page Hierarchy (3 of 7) .....	123
Figure 44: Central Login Application Page Hierarchy (4 of 7) .....	124
Figure 45: Central Login Application Page Hierarchy (5 of 7) .....	124
Figure 46: Central Login Application Page Hierarchy (6 of 7) .....	125
Figure 47: Central Login Application Page Hierarchy (7 of 7) .....	125
Figure 48: Maintenance Page for AcS Component.....	126
Figure 49: SSOi Maintenance Page .....	127
Figure 50: Responses to Produce WS Security Headers Sequence Diagram .....	133
Figure 51: Responses to XML Encryptions, Decryptions, and Digital Signature Sequence Diagram .....	134

# 1. Introduction

The Department of Veterans Affairs (VA) currently serves Veterans, their beneficiaries, and other VA stakeholders via services across many distributed and often operationally disjoint Lines of Business (LOB). Though VA serves the stakeholders across a vast enterprise of internal and external businesses and programs, it currently lacks a single, uniform method for identifying stakeholders and applying Access Management Services to safeguard its information resources. VA also lacks the capability to harmoniously share and leverage sensitive information across its internal LOBs and external business partners. Based on this existing operating model, the Veterans Relationship Management (VRM) Program Management Office (PMO) has identified the need to establish core Access Services (AcS) to definitively and consistently identify VA stakeholders and to establish supporting processes that increase the level of security protecting the identities, information, and interests of VA stakeholders.

The enterprise-wide system as a whole is referred to as the VA AcS 2.0, which includes the applicable subcomponents. The individual subcomponents or groups are referred to as a VA AcS activity or the VA AcS activities. The VA AcS activities include the following:

Single Sign-On – Internal (SSOi)	Identity Proofing (IP)
Single Sign-On – External (SSOe)	Provisioning (PROV)
Credential Service Provider (CSP)	Specialized Access Control (SAC)
Electronic Signature (eSig)	Compliance Audit and Reporting (CAR)

Within each of the AcS activities, commercial off-the-shelf (COTS) products are used to enable the specific capabilities of the SSOi described in this document and identified by the business as referenced (where applicable) in the Business Requirements Document (BRD) and Requirements Specifications Document (RSD). The SSOi's primary customers are both internal and external user communities who need logical access to VA business applications.

An Enterprise SSOi solution is in great demand by internal VA users and application owners. Many inefficiencies and security risks result from the current process of issuing a variety of User IDs and passwords for access to VA resources. VA's Homeland Security Presidential Directive 12 (HSPD-12) Personal Identity Verification (PIV) Program currently has the responsibility to issue VA PIV cards. The current PIV process ties the cards to the VA Active Directory system and provides network authentication. By integrating the SSOi effort with Active Directory, PIV and Windows authentication, assistance is provided to VA in meeting the Office of Management and Budget guidance regarding PIV Authentication for all VA applications. Joint VA/DoD health care facilities necessitate the need to include the DoD Common Access Card (CAC) in the SSOi solution. Additionally the SSOi solution will need to support internal federation capabilities through the VAAFI architecture.

## 1.1. Purpose of the SDD

The purpose of the System Design Document (SDD) is to describe the supporting mechanics of the SSOi solution. The SDD translates the requirement specifications into a document from which the developers may create the technical solution. It identifies the top-level system architecture, as well as the supporting hardware, software, communication, and interface

components. This artifact is an evolving document and is a living artifact that is that is updated (as applicable) when modifications are incorporated and / or new capabilities are added to the solution (when appropriate).

The primary target audience is SSOi developers and teams who will assist in the establishment of the infrastructure, as well as the following stakeholders:

- VA, Department of Defense (DoD), business partners, and other federal agencies
- AcS 2.5.0 Architects
- AcS 2.5.0 Business Sponsors
- Developers and technical managers
- Senior management and mission owners who enforce decisions about the IT security budget
- IT security program managers, who implement the security program
- Information System Security Officers (ISSO) responsible for IT security
- IT application owners of software and/or hardware used to support AcS activities
- Information owners of data stored, processed, and transmitted by the IT applications
- Other technical support personnel and product vendors

This document provides the solution architecture and detailed design of the SSOi solution as well as details for understanding the specific system configurations, interfaces, workflow, Graphical User Interfaces (GUI), and data models.

## 1.2. Identification

The information contained herein is based on the CA Technologies (CA) COTS products to provide the core capabilities for access control services to VA stakeholders. This document explains the manner in which these COTS solutions will be deployed to provide the foundation system and software to be used by the AcS 2.5.0. This document applies to the following systems and software:

Name	Description	Abbreviation	Version	Release
VA AcS 2.5.0	Core set of activities to definitively and consistently identify VA stakeholders and to establish supporting processes that provide the appropriate level of security required to protect and manage the identities, information, and interests of the VA stakeholders	AcS	V 2.5.0	Release 5 (Increment 5)
Single Sign-On – Internal	Provides Single Sign-on functionality for internal users utilizing the Personal Identification Verification (PIV) card, Active Directory (AD) or Windows Authentication	SSOi	N/A	N/A

## 1.3. Scope

This SDD focuses on the technical system design to provide the foundation for the \_SSOi solution. It provides an overview of the core capabilities, architecture, and design. It does not include default COTS product design nor does it include OOTB data definitions, tables, or models except where the design alters such elements and components. The sections below provide scope inclusion and exclusion details.

**Note:** The remote proofing service is provided on another contract and supported through VAAFI.

**Table 1: Scope Inclusions**

<b>Includes</b>
Provides authentication and authorization support for VA applications
Accepts federated credentials through VA Authentication Federation Infrastructure (VAAFI) for third party providers such as: DoD Users (CAC), USAA, FCCX, and non-VA PIV
Provides VA internal users authentication and authorization support on mobile devices
Provides legacy application support for SSO
Provides support for PIV Compliant authentication (LOA 3)
Provides global log off for integrated applications/services
Provides Secure Token Service (STS) capabilities with a response message that supports the SAML format, WS-Trust protocol, and WS-Policy protocol.
Provides support for extending the SAML attributes with Provisioning data
Provide OAuth Protocol

**Table 2: Scope Exclusion**

<b>Excludes</b>
No support of biometric authentication is provided due to limitation of current products
Logging and Reporting Systems provided by the operations team (covered in the Production Operations Manual)
Backup Requirements (covered in the Production Operations Manual)
Installation and configuration instructions for software (covered in the Installation and Configuration Guide)
Details of individual partner configurations (covered in the Interface Control Documents [ICDs])

### 1.3.1. Increment 5 SSOi Scope

N/A

## 1.4. Constraining Policies, Directives and Procedures

This design complies with the following policies, directives, and procedures (as applicable). The specific requirement and sub-requirement numbers are highlighted in the individual service-specific SDDs (where appropriate).

**Table 3: Policies, Directives, and Procedures**

#	Issuing Agency	Policy, Directive, or Procedure	Purpose
1	VA	VA 6500 Handbook	<ul style="list-style-type: none"><li>• Directive Information Security Program.</li><li>• Defining overall Security Framework for VA.</li></ul>
2	VA	VA 6501 Directive	<ul style="list-style-type: none"><li>• VA Identity Verification In-Person Proofing (IPP) Process.</li><li>• Defining overall Identity Proofing Methodology for VA IAM.</li></ul>
3	VA	VA 6300 Directive	<ul style="list-style-type: none"><li>• Directive Records and Information Management.</li><li>• Defines information management framework for VA Access Services.</li></ul>
4	NIST	SP 800-53-4	<ul style="list-style-type: none"><li>• Special Publication – Recommended Security Controls for Federal Information Systems and Organizations.</li><li>• Defines the required security controls for IT systems under the Federal Information Security Management Act (FISMA).</li></ul>
5	NIST	SP 800-63-2	<ul style="list-style-type: none"><li>• Special Publication – Electronic Authentication Guideline.</li><li>• Defines levels of assurance in user identities presented to IT systems over open networks.</li><li>• Defines the data and procedural requirements for VA Access Services.</li></ul>
6	NIST	FIPS-201-2	<ul style="list-style-type: none"><li>• Federal Information Processing Standards Publication – PIV of Federal Employees and Contractors.</li><li>• Provides Identity Proofing, credentialing and chain of trust requirements and processes.</li><li>• Defines the method for secure administrative interaction and control.</li></ul>
7	NIST	FIPS-140-2	<ul style="list-style-type: none"><li>• Federal Information Processing Standards Publication (FIPS) – Security Requirements for Cryptographic Modules.</li><li>• Defines the cryptographic standards and requirements.</li></ul>

#	Issuing Agency	Policy, Directive, or Procedure	Purpose
8	NIST	SP 800-122	<ul style="list-style-type: none"> <li>• Guide to Protecting the Confidentiality of Personally Identifiable Information (PII).</li> <li>• Provides technical procedures for protecting PII in information systems.</li> <li>• Defines the information which can be used to distinguish or trace an individual's identity.</li> </ul>
9	US Congress	Section 508 Amendment to the Rehabilitation Act of 1973	<ul style="list-style-type: none"> <li>• Section 508 Electronic and information technology requirements for Federal departments and agencies.</li> <li>• Accessibility, development, procurement maintenance, or use of electronic and information technology.</li> <li>• Defines the “Human-Machine Interface” accessibility requirements.</li> </ul>
10	OMB	M-04-04	<ul style="list-style-type: none"> <li>• Memorandum to the Heads of All Department and Agencies – E-Authentication Guidance for Federal Agencies.</li> <li>• Defines the E-Authentication requirement.</li> </ul>
11	OMB	M-11-11	<ul style="list-style-type: none"> <li>• Requirements for Accepting Externally-Issued Identity Credentials.</li> <li>• FICAM architecture and procedures for federal agencies.</li> </ul>
12	GSA	FICAM	<ul style="list-style-type: none"> <li>• Federal Identity, Credentialing and Access Management (FICAM) Roadmap and Implementation Guidance.</li> <li>• Provides the common segment architecture and implementation guidance for federal ICAM programs.</li> </ul>
13	White House	NSTIC	<ul style="list-style-type: none"> <li>• National Strategy for Trusted Identities in Cyberspace (NSTIC) – Provides guidance for identity trust in cyberspace.</li> </ul>
14	US Congress	FISMA	<ul style="list-style-type: none"> <li>• FISMA of 2002, Public Law 107-347</li> </ul>
15	US Congress	E-Government Act of 2002	<ul style="list-style-type: none"> <li>• Federal Management and Promotion of Electronic Government Services.</li> <li>• Defines the requirements for electronic services.</li> </ul>
16	US Congress	The Privacy Act of 1974	<ul style="list-style-type: none"> <li>• § 552a. Records maintained on individuals.</li> <li>• Defines VA Access Services Privacy assessment and control requirements.</li> </ul>

#	Issuing Agency	Policy, Directive, or Procedure	Purpose
17	National Archives and Records Administration (NARA)	Federal Records Act	<ul style="list-style-type: none"> <li>Establishes the framework for records management programs in Federal Agencies.</li> </ul>
18	VA	VA D 0735	<ul style="list-style-type: none"> <li>Homeland Security Presidential Directive 12 (HSPD-12) Program</li> <li>Defines Department-wide policy, roles, and responsibilities for the creation and maintenance of systems and processes to implement VA's HSPD-12 Program necessary to implement Homeland Security Presidential Directive 12 (HSPD-12) program.</li> </ul>
19	OMB	M-05-24	<ul style="list-style-type: none"> <li>Implementation of HSPD 12 – Policy for a Common Identification Standard for Federal Employees and Contractors.</li> </ul>

## 1.5. User Characteristics

The user community for the SSOi activities consists of internal users including VA employees, contractors, and affiliates. SSOi also supports external business users from other government agencies like DoD for accessing VA internal business applications.

## 1.6. Relationship to Other Documents and Plans

The system design is developed based on the progressive refinement and discovery of business and functional requirements outlined and extracted from the following documents, which are located on the [AcS TSPR](#) site.

**Note:** The applicable standards and guidelines from the VA Handbook and NIST are identified in section 1.5 above.

## 1.7. Definitions, Acronyms, and Abbreviations

The abbreviations and terms used in this SDD are defined in the [Identity and Access Services Master Glossary](#).

## 1.8. References

Refer to Section 1.6 for document references.

# 2. Background

The purpose of the VA AcS Development Support task is to design, develop, implement, integrate, operationalize, and sustain an enterprise-wide VA AcS 2.5.0 for VA VRM. In order to coordinate AcS across several VRM work streams, multiple internal and external systems will need to be interconnected to provide access to these systems by facility, system and individual entities. The goal of AcS is to facilitate access transactions using an Enterprise Services

framework. The Framework should address the user account lifecycle, from identity creation through de-provisioning of the user. To accomplish these goals, the AcS should consider highly available services in an effort to minimize unintentional disruptions for the users.

This document provides the underlying design to support the SSOi activities. The system design is based on a Service Oriented Architecture (SOA) approach. The solution architecture uses accepted COTS products for each of VA AcS activity and applies the leading practices as outlined by the product vendor to the extent possible. The design of the architecture supports VA's scalability, security, extensibility, and high availability requirements to provide a flexible enterprise solution.

## **2.1. Overview of the System**

The AcS 2.5.0 is made up of several activities, which are necessary to provide identity and access management services to both internal VA employees / contractors and to external end users. It provides VA applications centralized authentication mechanism for internal users and federation capabilities to access external application. Authorization capabilities to provide coarse- and fine-grained application access while providing workflow for self-service account requests, approvals, and user life cycle management.

An Enterprise SSOi solution is in great demand by internal VA users and application owners. Many inefficiencies and security risks result from the current process of issuing a variety of User IDs and passwords for access to VA resources. VA's Homeland Security Presidential Directive 12 (HSPD-12) Personal Identity Verification (PIV) Program currently has the responsibility to issue VA PIV cards. The current PIV process ties the cards to the VA Active Directory system and provides network authentication. By integrating the SSOi effort with Active Directory, PIV and Windows authentication, assistance is provided to VA in meeting the Office of Management and Budget guidance regarding PIV Authentication for all VA applications. Joint VA/DoD health care facilities necessitate the need to include the DoD Common Access Card (CAC) in the SSOi solution. Additionally the SSOi solution will need to support internal federation capabilities through the VAAFI architecture. The internal user population that requires some form of logical access is estimated at approximately 400,000.

## **2.2. Overview of the Business Process**

Refer to the VA AcS 2.5.0 Requirements Specification Document (RSD), use case, and Requirements Traceability Matrix (RTM) documents for the business process flows.

## **2.3. Business Benefits**

This section describes the assumptions and constraints that impact the design of the SSOi solution.

## **2.4. Assumptions and Constraints**

### **2.4.1. Design Assumptions**

This section describes the assumptions that impact the design of the SSOi solution.



**Table 4: Assumptions**

Assumption
<p>SSOi Activity OOTB standard reporting will be provided for applications integrated with CA SiteMinder and CA SSO toolset using CAR Activity.</p> <p>LOA 4 “Holder Of the Key” functionality is not supported with a Federated SAML profile.</p> <p>The Identity Provider (IdP), Service Provider (SP) and STS (Security Token Store) capabilities will be developed using AcS available product capabilities.</p> <p>The SSOi Activity will use VA Active Directory (AD) as primary authentication store and thereby provide desktop SSO capability only to users in VA AD. SSOi will also leverage the attribute service provided by the Radiant Logic virtual directory to retrieve attributes about an authenticated user.</p> <p>SSOi administrator interface, similar to SiteMinder Admin UI, does not support PIV authentication due to the COTS product limitation; therefore, PIV Authentication capability will not be enabled for the SiteMinder or CA SSO Administrator Interface.</p> <p>The SSOi centralized logon page, as well as the SSOi integrated application platforms, will have similar branding capabilities amongst one another to provide for a streamlined visual and functional perspective for integrating application</p> <p>Mobile authentication will utilize SiteMinder for token issuance. Due to the larger size of the token itself, a limited number of mobile devices will be able to accept them.</p>
<p>This design assumes that Citrix Netscape Global Traffic Manager (GTM) module will be available at the time of production implementation.</p> <p>Virtual machines used for the VA AcS infrastructure will be integrated in the appropriate VA Active Directory domain for each environment.</p> <p>The AcS 2.0 is designed to have 99.99% availability, and can be failed over to the Disaster Recovery site. However, this is contingent on the availability of other components outside of the AcS 2.0 such as VAAFI and Terremark, which only support 99.99% and 99.99% availability, respectively. Therefore, if the solution components support 99.99% availability, this may not be achieved due to external dependencies which may be limited to the VAAFI 99.99% figure.</p> <p>The VA issues the necessary internal and external TLS/SSL certificates. Applications use self-signed certificates for internal server communications, and use VA issued certificates between remote servers to secure data and messages between applications.</p> <p>Virtual machines used for VA AcS infrastructure will be integrated in the appropriate VA Active Directory domain for each environment.</p>

## 2.4.2. Design Constraints

This document is developed under the schedule and cost defined in the contract for VA AcS development support. The design is constrained to features available in the tools, technologies, and frameworks defined by VA Technical Reference Model (TRM) tools list and those that have been accepted by VA.

- **CA SiteMinder:** The SiteMinder Administration Console does not support PIV authentication. As an alternative, a link to the SiteMinder Administration Console may be accessed for authorized persons through the CA Single Sign-On product.
- **CA SiteMinder:** The product may be out of compliance during the implementation / functioning if proper steps to patch are not followed. When using SiteMinder Federation

capabilities with this product, SiteMinder Federation must remain properly patched in order to mitigate known security vulnerabilities. Version Federal Information Processing Standards (FIPS 140-2) certified encryption must be used to encrypt data in transit if Personally Identifiable Information (PII), Personal Health Information (PHI), or Veteran Affairs (VA) sensitive information is involved or additional mitigating controls must be documented in an approved System Security Plan (SSP). VA users must properly protect VA sensitive data in accordance to VA 6500 Policy and the Federal Information Security Management Act (FISMA).

- **CA Secure Proxy Server:** The product must be configured to run in FIPS only mode in order to satisfy FIPS140-2 requirements.
- **DataPower XML Security Gateway:** Appliance must be operated on FIPS 140-2 compliant hardware with embedded hardware security modules (HSM).

### 2.4.3. Design Trade-offs

The following are the design trade-offs for the SSOi solution design:

- The user store and policy store have read-intensive operations. Based on the projected usage demands, the policy store and user store should be created in their own CA Directory Servers instances. Alternatively, if the stores are consolidated on common servers with failover topology, system's performance may degrade between the read and write transactions. Additionally, if the read intensive operations are occurring in the same place where the data is being written then it is likely that data mismatch may occur at time of the reading transaction.
- Since the SSOi administrative UI does not support direct PIV authentication, as an alternative, the administration console links may be provided in the CA Single Sign-On system and rely on the Desktop PIV login. However, a username and password will still be required for the administration consoles.
- Role manager uses the CA LDAP provisioning identity store as the authoritative store for user identity, as the LDAP store contains both employees and contractor information. The identity store however does not contain the manager attribute for employees who are not on-boarded via CRISP (or unless manually updated in role manager), which may impact VA's ability to perform manager-based access recertification.
- The ARX CoSign device's support for signing of web forms is indirect. It requires converting the web-based form, snippet of code or UI component to a standard, supported document type (e.g., Adobe Acrobat PDF, Microsoft Word) before being able to sign it.

## 2.5. Overview of the Significant Requirements

This section includes an overview of significant requirements. This version of the SDD meets the following high-level i5 requirements:

**Table 5: High-Level Requirements**

No.	High-Level Requirement
1.	Develop SSOi services required to support mobile credentials to include: <ul style="list-style-type: none"> <li>• Mobile Client Registration</li> <li>• Supporting management and enforcement of OAuth policies</li> <li>• Supporting OAuth from the provider perspective</li> <li>• Supporting OAuth enforcement from application perspective</li> </ul>
2.	Develop SSOi STS functionality (Phase 1) to include: <ul style="list-style-type: none"> <li>• REST interface for obtaining tokens</li> <li>• Issuance and validation of JSON Web Tokens with signature and encryption capability</li> </ul>
3.	Enhance Global log off capability to include application logout functionality
4.	Develop SSOi enhancements to include: <ul style="list-style-type: none"> <li>• Fine-grained revocation.</li> <li>• Limiting the number of access or refresh tokens</li> <li>• Self-registration of client</li> </ul>
5.	Provide SSOi configuration and development support; <ul style="list-style-type: none"> <li>• Enhance deployed functionality and SSOi integration patterns (VA IAM AcS Integration Patterns Document) required to authenticate internal users including: <ul style="list-style-type: none"> <li>▪ CA SiteMinder WebAgent</li> <li>▪ CA SiteMinder SPS</li> <li>▪ SSOi STS</li> <li>▪ IdP to SP Federation</li> </ul> </li> </ul>
6.	Enhance deployed functionality and SSOi integration patterns (VA IAM AcS Integration Patterns Document) required to perform global log off of internal users including: <ul style="list-style-type: none"> <li>• CA SiteMinder WebAgentCA</li> <li>• SiteMinder SPS</li> <li>• SSOi STS</li> <li>• IdP to SP Federation</li> </ul>
7.	Enhance SSOe activity to comply with performance specifications as detailed in RSD

## 2.5.1. Overview of Significant Functional Requirements

**Table 6: Functional Requirements**

Epic		Feature			
EPIC ID	Summary	Feature ID	Summary	Story ID	Summary

Epic		Feature			
EPIC ID	Summary	Feature ID	Summary	Story ID	Summary
159286	AcS 2.0 i5 - STS Support of JSON Web Token and Bearer Token	461870	SSOi shall provide a JSON web token within the Open Authorization Standard (OAuth) framework.	150251	As a partner application, I want to receive an OAuth JSON web token so that I can be authenticated and authorized.
		461871	SSOi shall provide a JSON web token within the STS framework.	150252	As a partner application, I want to receive a STS JSON web token so that I can communicate with other client applications.
		461872	SSOi shall be able to digitally sign and encrypt JSON web tokens.	150253	As a partner application, I want to be able to digitally sign JSON web tokens to insure integrity.
				162118	As a partner application, I want to be able to encrypt JSON web tokens so I can communicate in a secure manner with other applications.
		461873	SSOi shall be able to verify digitally signed JSON web tokens and decrypt encrypted JSON web tokens.	150254	As a partner application, I want to be able to send a signed JSON web token to insure integrity.
				162108	As a partner application, I want to send encrypted JSON Web tokens so I can communicate with other applications in a secure manner.
		461874	SSOi shall accept JSON Web tokens.	150255	As a partner application, I want to be able to send JSON Web tokens to communicate with other applications.
		461875	SSOi shall accept JSON Bearer tokens.	150256	As a partner application, I want to be able to send JSON Bearer tokens to communicate with other applications.

Epic		Feature			
EPIC ID	Summary	Feature ID	Summary	Story ID	Summary
				N/A	STS Core - Retrieve SAML token from back end
		N/A	N/A	N/A	STS Core - Enhance to meet other specific requirements
159297	AcS 2.0 i5 - Expose STS Service with REST Interface	461877	SSOi shall expose the STS service with a REST interface.	150257	As a partner application, I want to be able to use STS services with the REST interface to communicate with other applications.
		461878	SSOi shall secure the REST STS service with mutual Transport Layer Security (TLS).	150258	As a partner application, I want to be able to use REST STS services with TLS to communicate with other applications.
		N/A	N/A	N/A	SSOi STS Backend - Create Web Service
		N/A	N/A	N/A	SSOi STS Backend - Investigate/Design (how to pull in and store)
		N/A	N/A	N/A	SSOi STS Backend - Investigate/Design (sessions, tokens, VDS, Prov)
		N/A	N/A	N/A	SSO STS Front End - Create SOAP interface with transformation
		N/A	N/A	N/A	SSO STS Front End - Create JSON interface with transformation
		N/A	N/A	N/A	SSO STS Front End - Create REST interface with transformation
160071	AcS 2.0 i5 - SSOi Deferred Requirements for i5	468730	SSOi shall support mobile client registration (Internal for CAC Credential)	162174	As an SSOi user, I want to be able to register my mobile client so I can use my CAC credentials.
		468769	SSOi shall support mobile client registration (Internal for Mobile Applications).	162176	As an SSOi user, I want to be able to register my mobile client so I can use my mobile applications.

Epic		Feature			
EPIC ID	Summary	Feature ID	Summary	Story ID	Summary
		468732	SSOi shall support management and enforcement of OAuth policies (Internal for CAC Credential).	162190	As a SSOi administrator, I want to be able to create new OAuth policies to control CAC credentials.
				162195	As a SSOi administrator, I want to be able to edit existing OAuth policies so I can make changes to CAC Credential policies.
				162197	As a SSOi Administrator, I want to be able to delete OAuth policies so I can remove CAC Credential policies that are no longer required.
				162202	As a partner application, I want to be able to use OAuth CAC Credential policies to control user access to my application.
		468771	SSOi shall support management and enforcement of OAuth policies (Internal for Mobile Applications).	162314	As a SSOi administrator, I want to be able to create new OAuth policies to control mobile applications.
				162315	As a SSOi administrator, I want to be able to edit existing OAuth policies so I can make changes to mobile application policies.
				162317	As a SSOi Administrator, I want to be able to delete OAuth policies so I can remove CAC Credential policies that are no longer required.
				162319	As a partner application, I want to be able to use OAuth mobile application policies to control user access to my application.

Epic		Feature			
EPIC ID	Summary	Feature ID	Summary	Story ID	Summary
		468733	SSOi shall support fine-grained revocation (Internal for CAC Credential).	162467	As a SSOi administrator, I want to be able to revoke CAC Credentials, so that CAC Credential polices will not be enforced on an individual.
		468772	SSOi shall support fine-grained revocation (Internal for Mobile Applications).	162473	As a SSOi administrator, I want to be able to revoke mobile applications, so that mobile application polices will not be enforced on a particular mobile application.
		468734	SSOi shall support limiting the number of access or refresh tokens (Internal for CAC Credential).	162476	As a partner application I want to limit the number of access/refresh token created for an individual, to improve performance.
		468773	SSOi shall support limiting the number of access or refresh tokens (Internal for Mobile Applications).	162477	As a partner application I want to limit the number of access/refresh token created for an application, to improve performance.
		468735	SSOi shall support self-registration of clients (Internal for CAC Credential).	162478	As a client, I want to be able to self-register my CAC credentials so that I can use SSOi services.
		468774	SSOi shall support self-registration of clients (Internal for Mobile Applications).	162479	As a client, I want to be able to self-register my mobile application so that I can use SSOi services.
		468767	SSOi shall support OAuth from the provider perspective	162480	As a client, I want to be able to use OAuth services to create policies.
		468768	SSOi shall support OAuth enforcement from	162481	As a client, I want my OAuth polices to be

Epic		Feature			
EPIC ID	Summary	Feature ID	Summary	Story ID	Summary
			application perspective		enforced to provide access to my application.
					Design OAuth
					Implement OAuth Design
					Create Virtual Desktop
					Add Android/iPhone emulators to Virtual Desktop
					Add Kerberos participation to Virtual Desktop
					Add SOAP UI to Virtual Desktop
159383	AcS 2.0 i5 - Standard SSOi Traits	474250	SSOi shall provide General VA traits. See table in AcS 2.0 Increment 5 RSD.	150557	As a client application, I want a common set of general traits to setup communications between SSOi and my application.
		474251	SSOi shall provide VHA traits, which are the General VA traits plus user identity in VHA systems. See in AcS 2.0 Increment 5 RSD.	150558	As a client application, I want to have a set of VHA traits to setup communications between SSOi and my client application.
		474252	SSOi shall provide VBA traits, which are the General VA traits plus user identity in VBA system. See table in AcS 2.0 Increment 5 RSD.	150561	As a client application, I want to have a set of VBA traits to setup communications between SSOi and my client application.

## 2.5.2. Overview of Functional Workload / Performance Requirements

**User Profile:** VA Employee or Contractor who wants to gain access to an SSOi-protected application

The performance specifications for the SSOi service include the following:

- SSOi shall support 20 million authentications per day.
- VDS shall support 20 million authentications by SSOi per day.



- c. The IAM Binding Application shall support 1 million authentications per day.
- d. The IAM Binding Application shall support 1 million users.
- e. SSOi shall be able to handle an increase of 1 million users with integration to VistA.

The SSOi service for this increment shall support the following:

**Table 7: Workload and Performance Requirements**

Operation	
<b>Name</b>	SSOi User Authentication (CA SiteMinder WebAgent, CA SiteMinder SPS and IdP to SP Federation)
<b>Usage Profile (User Authentication Events)</b>	
Mean Daily volume	40000
Projected Growth	8000/year
Peak Daily volume	50000
Projected Growth	10000/year
Peak Hourly volume	8000
Days of operation	Sunday-Saturday
Hours of operation	24/7
Peak Hours	9am-7p.m.Eastern
Maximum Response Time	10 seconds

Operation	
<b>Name</b>	SSOi STS
<b>Usage Profile (Token Requests)</b>	
Mean Daily volume	0
Projected Growth	10000/year
Peak Daily volume	0
Projected Growth	25000/year
Peak Hourly volume	0
Days of operation	Sunday-Saturday
Hours of operation	24/7
Peak Hours	9am-7p.m.Eastern
Maximum Response Time	1 second

### 2.5.3. Overview of Operational Requirements

The AcS 2.0 is hosted within the Terremark environment as required by VA. Terremark is responsible for reliability and monitoring when the AcS 2.0 becomes operational. The tools, methods, and specifications for monitoring the reliability of the AcS 2.0 are at the discretion of Terremark.

**Table 8: Service Availability Level 4**

<b>*Standards adopted from specification created by Application Structure and Integration Services (ASIS)</b>	
<b>Description</b>	Mission Critical Information
<b>Minimum Availability</b>	99.99%
<b>Maximum Downtime Per Month</b>	4.4 minutes
<b>Business Value</b>	Essential to fundamental business operations – outage seriously impairs functioning of business.
<b>System Response</b>	In the absence of any system superseding requirements, the system responds to user actions in three seconds or less in 90% of the attempts, and never more than 10 seconds.
<b>Operational Hours</b>	Required 24 hours a day, every day.
<b>Significant Outage</b>	More than five minutes of downtime is considered significant at any time and requires an ANR to be sent out to the appropriate teams.
<b>Outage Impact</b>	Interruption of service may result in severe financial, regulatory, patient safety, patient health, or other business issues.
<b>Scheduled Maintenance</b>	Maintenance, including maintenance of externally developed software incorporated into the IAM system, is scheduled during off-peak hours (evenings and weekends) or in conjunction with relevant maintenance schedules.

Additional reliability specifications (response times, monitoring, maintenance periods, and operational support) may be viewed in the [IAM SLA](#).

#### **2.5.4. Overview of the Technical Requirements**

TBD will obtain RTM from Rational.

#### **2.5.5. Overview of the Security or Privacy Requirements**

N/A

#### **2.5.6. Overview of System Criticality and High Availability Requirements**

The VA AcS infrastructure supports critical business systems. The current availability requirement for mission critical systems is 99.9%. The current data centers support 99.6% availability. The Production, Preproduction, and Disaster Recovery (DR) Data Center is hosted by Terremark in Culpeper, Virginia and Miami, Florida. Terremark does not currently support an active/active geographic failover and load balancing thus failover to the DR site could take between one (1) and eight (8) hours. To mitigate the risk of not having a complete site failover, the AcS production infrastructure is intended to be scalable with limited single points of failure.

The primary production platform is virtualized with a physical servers dedicated to Oracle RAC and VDS.

The DR site is contingency site that will resume data center operations in the event of a site failure. Load balancing, fault tolerance, backups and archiving, is a function of the hosting facility, Terremark and the data center operations team. Backups are described more fully in the Production Operations Manual (POM), but essentially are the following:

- Full backups are taken of virtual machines on a weekly basis
- Backups of virtual machines must be transported off-site at least monthly
- Backups of specific databases will be taken daily between the hours of 2 a.m. and 5 a.m. Locations of the databases will be provided in the POM.

### **2.5.7. Single Sign-on Requirement**

In general, applications are meeting Single-Sign-On requirements by integrating with SSOi or other enterprise authentication solutions. SSOi is an officially approved solution designed to specifically meet these requirements..

### **2.5.8. Requirement for Use of Enterprise Portals**

SSOi is independent of any existing enterprise portal solutions. However, SSOi can be integrated with any of the enterprise portals to fulfill Single-Sign-On-related requirements

### **2.5.9. Special Device Requirements**

SSOi makes use of several WebSphere DataPower SOA appliances for token generation and end point exposure.

## **2.6. Legacy System Retirement**

This section is not applicable as no legacy systems are being retired as a result of the SSOi solution implementation.

## **3. Conceptual Design**

This section of the SDD provides details about the following topics:

- Conceptual Application Design
- Conceptual Data Design
- Conceptual Infrastructure Design

### **3.1. Conceptual Application Design**

This section provides the conceptual design of the SSOi solution.

#### **3.1.1. Application Context**

Single Sign-On – Internal (SSOi) is an authentication service designated for operations-based applications. These are typically described as business applications and not Veteran self-service applications, and are both externally and internally facing VA users and applications. This

service provides the capability to enhance the user experience by reducing time associated with multiple log-on/log-off activities, enriched password management, and reduction in help desk support. The SSOi service is client based service that allows internal VA users such as employees, contractors and partners within VA network to log on to integrated applications. The SSOi service connects to VA AD to validate user's credentials from desktop session or Kerberos token, uses Federation to support external cloud providers and accept users from SSOe while also utilizing HSPD-12 trust services to authenticate internal VA PIV users.

The primary actors interacting with the SSOi application are the following:

- Administrator (Privileged User): Responsible for control and maintenance of the SSOi (CA SSO and CA SiteMinder) and also responsible for running reports
- SSOi User: User who is using the SSOi service to log on to applications once they have logged on to their desktop successfully

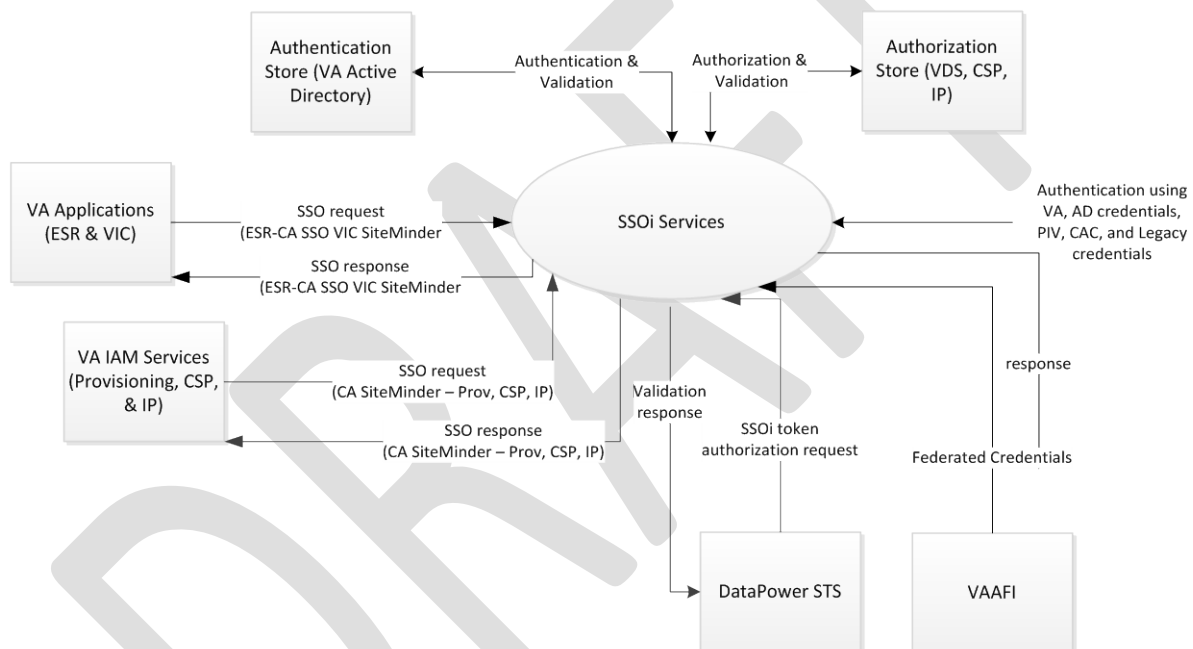
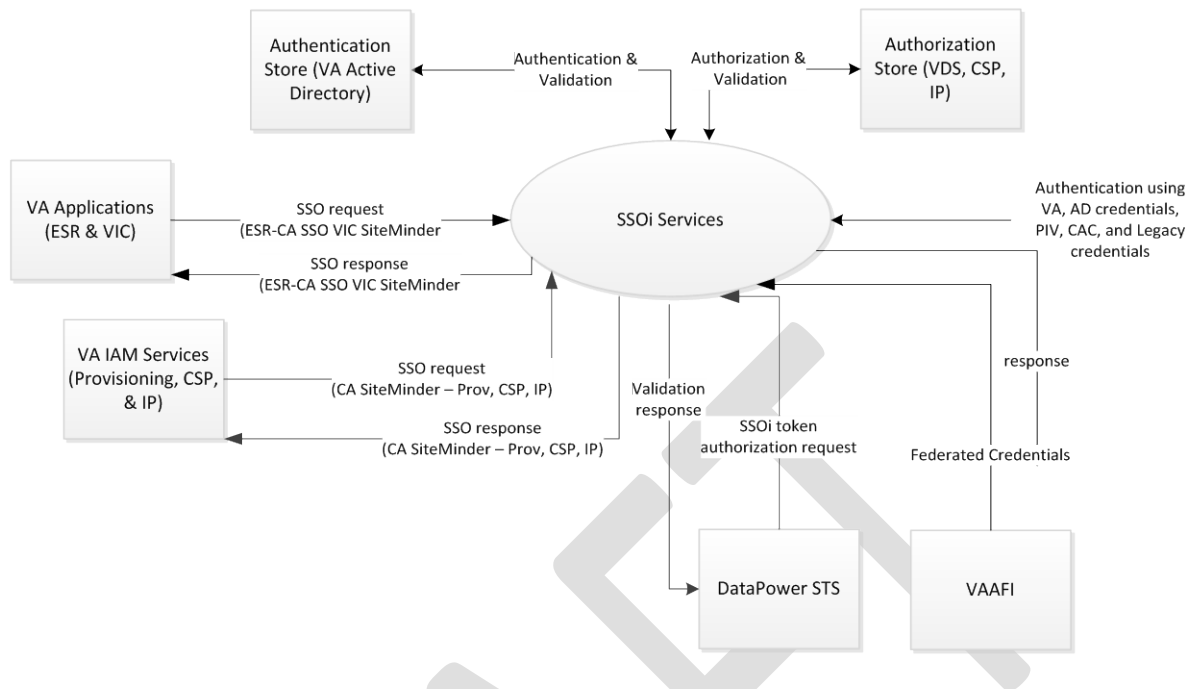


Figure 1 below is an overview of business interactions between SSOi, its clients, and supporting systems.



**Figure 1: SSOi Context Diagram**

The table below provides a description of the application context for SSOi.

**Table 9: SSOi Application Context Description**

ID	Interface Name	Relegated Object	Input Messages	Output Messages	External Party
1	VA Active Directory (AD)-SSOi	SSOi Service	LDAP queries	LDAP response / search results	LDAP Interface. VA AD is queried by SSOi Service to obtain VA internal user information. IAM (CA SiteMinder) uses the LDAP protocol to communicate with AD. AD is leveraged primarily to authenticate internal VA users.

ID	Interface Name	Relegated Object	Input Messages	Output Messages	External Party
2	Virtual Directory Service-SSOi	SSOi Service	LDAP queries	LDAP response / search results	LDAP Interface. VA VDS is queried by SSOi Service to obtain VA internal/external user information and also provide attribute authorization. IAM (CA SiteMinder) uses the LDAP protocol to communicate with VDS. VDS is leveraged primarily to authorizing VA users.
3	CSP and IP Directory Service-SSOi	SSOi Service	LDAP Queries	LDAP response / search results	LDAP Interface. VA CSP and IP Store which is CA directory instance which is queried by SSOi Service to obtain VA internal/external user information and also provide authorization response.
4	SSOi Application	SSOi Service	HTTP/HTTPS	HTTP/HTTPS	The SSOi hosted application like centralized logon pages are consumed by SSOi integrated applications
5	VA Applications-SSOi	SSOi Service	HTTP/HTTPS	HTTP/HTTPS	VA application like ESR and VIC use the CA SSO desktop native connection methods to seamlessly log in users in to their web applications.
6	VAAFI-SSOi	SSOi Service	SAML request/response	SAML request /response	VAAFI interacts with SSOi service for federation as service provider or identity provider

ID	Interface Name	Relegated Object	Input Messages	Output Messages	External Party
7	DataPower – STS-SSOi	SSOi Service	WS-Trust Token request	WS-Trust Token response	DataPower acts as the STS store that supports token translation requests from the application end and will return the standard user attributes as a part of the response specification.

### 3.1.2. High-Level Application Design

Figure 2 below provides a high-level application design for the AcS 2.0 and identifies the major AcS activities and/or relationships with VA applications.

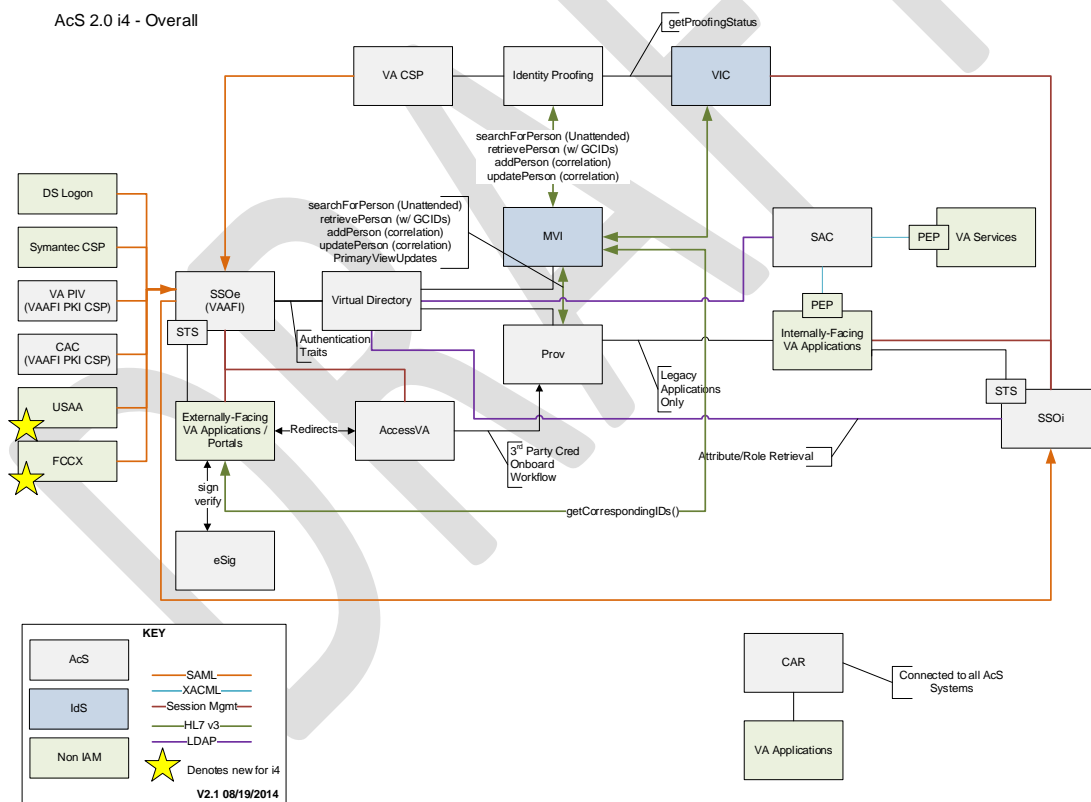
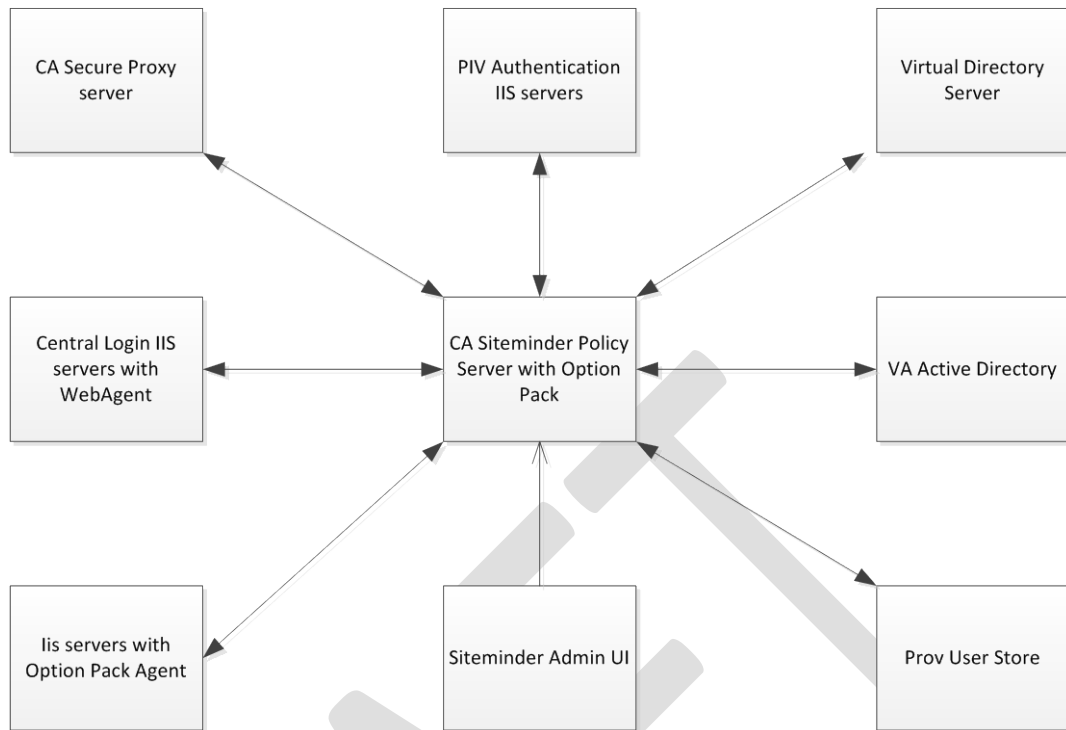


Figure 2: AcS 2.0 Application Design



**Figure 3: ACS SSOi High Level Design**

The following table provides a high-level description for each of the AcS activities. The external interfaces are interfaces for systems outside of VA and internal interfaces are interfaces for systems within VA.

**Table 10: Activities in the High-Level Application Design**

ID	Name	Description	Service or Legacy Code	External Interface Name	Internal Interface Name
1	CSP	CSP provides external user's credentials to VA applications that are not eligible for another VA approved credential.	Service	Self Service and Registration	VAAFI, IP, CAR
2	IP	IP facilitates evaluating and validating a user's identity to be true and unique to the degree (level) of confidence required by VA.	Service	N/A	MVI, CSP, CAR
3	eSig	eSig provides the ability to sign documents electronically.	Service	N/A	CAR



ID	Name	Description	Service or Legacy Code	External Interface Name	Internal Interface Name
4	SAC	SAC provides the ability to maintain and process granular access decisions based on a set of business rules and user attributes.	Service	N/A	CAR
5	Provisioning	Provisioning associates an identity to one or more application accounts and the associated entitlements to the identity. Provisioning also provides the capabilities for managing roles and certifying entitlements.	Service	TMS	AD, CAR, EDR, MVI, PIV,VDS,IP
6	SSOi	SSOi provides the desktop sign-on capability to internal VA users. SSOi also provides authentication and access to VA business applications for both internal and external user populations. External credentials are brokered by the VAAFI service and are a federated partner with SSOi.	Service	Federation	AD, IP, CSP, Provisioning, SAC
7	CAR	CAR provides the ability to proactively monitor, mitigate, and recover from potential compliance infractions and incidents.	Service	N/A	SSOi, Provisioning, CSP, IP, eSig, SAC

Table 11:

Table 11: Objects in the High Level Application Design

Objects / Components to be Built or Modified

ID	Name	Description	Service or Legacy Code	External Interface Name	External Interface ID	Internal Interface Name	Internal Interface ID	SDP Sections 1&2
N/A								

Internal Data Stores

ID	Name	Data Stored		Steward	Access
TBD	SiteMinder	TBD		TBD	TBD

### 3.1.3. Application Locations

The following table lists the application components and their locations where they will be hosted.

**Table 12: SSOi Solution Application Locations**

Application Component	Description	Location at Which Component is Run
IIS Web Server	Front end web server providing the administrative and self-service interface to CA IdentityMinder	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)
Servlet Exec	Application server for SiteMinder Federation option pack for CSP and SSOi partnerships with VAAFI. Servlet Exec is used in conjunction with CA SiteMinder Federation Option pack. All appropriate JVMs on all AcS environments were updated as part of the OIG audit findings resolution. The NewAtlanta ServletExec v6.0 from 11/30/2007 product supports a minimum of Java 1.5, as listed in the release notes, but has no specific references to restrictions about newer versions of Java. As ServletExec needs to have a waiver per TRM, this process will be initiated. Waiver approval will be reflected in the SDD	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)
CA SiteMinder	This is a set of features that provides Single Sign-On, session management, WS Security, Authentication and Authorization Policies, Policy Decision Point, and audit reporting for access controls.	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)
CA Secure Proxy Server	This is a stand-alone server that provides a proxy-based solution for access control.	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)

Application Component	Description	Location at Which Component is Run
CA SSO Server	CA desktop single sign-on solution for legacy applications.	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)
CA Directory	LDAP directory to support CA SiteMinder, CA SSO and CA IdentityMinder backend configuration and data store.	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)
IBM DataPower	COTS XML Security Gateway	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)
Oracle Database	Database to support CA IdentityMinder and audit logs from different components.	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)

**Table 13: Application Users**

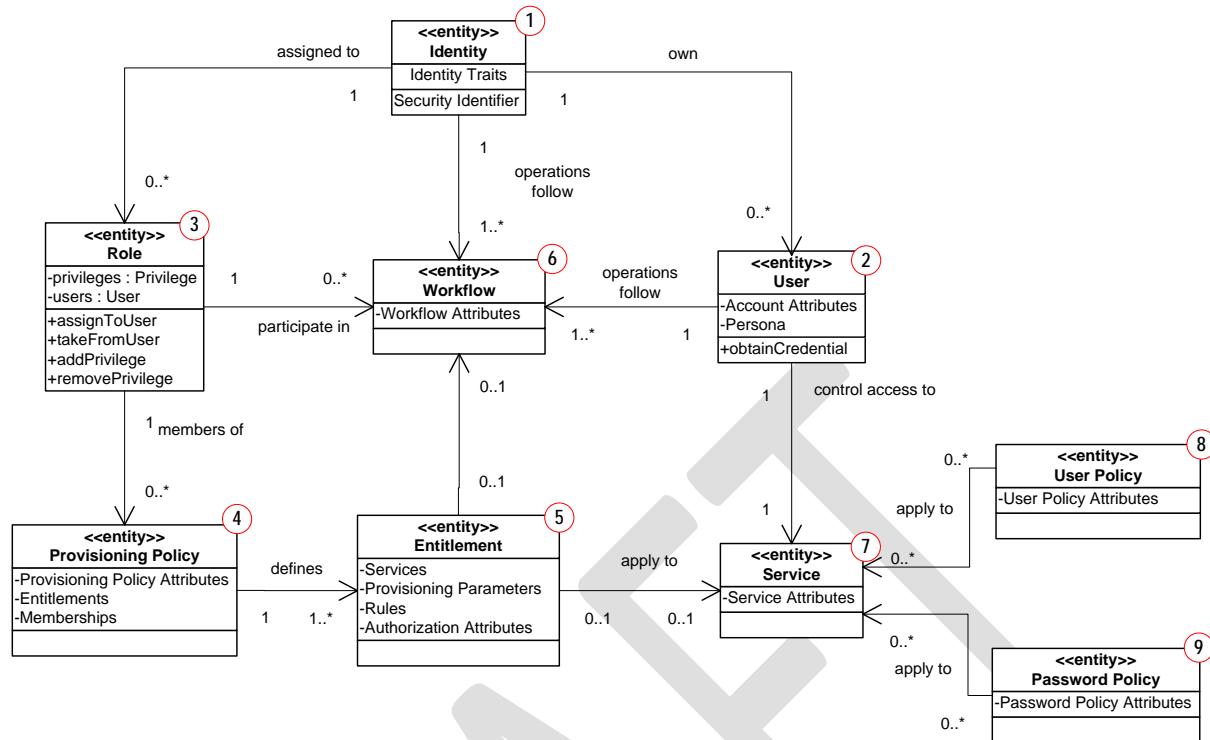
Application Component	Location	User
SSOi	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)	SSOi Administrator
SSOi	Terremark Culpeper, VA (Primary) Terremark Miami, FL (Disaster Recovery)	End User

## 3.2. Conceptual Data Design

The following sections provide the conceptual data design for the SSOi.

### 3.2.1. Project Conceptual Data Model

This section describes the conceptual data model providing high-level representation of the data entities and relationships. The data objects within the SSOi, how they are used, and how they relate to each other are provided in Figure 4. The data model is defined for CA IdentityMinder, which is used for Provisioning, CSP, and IP services for implementing VA business requirements.



**Figure 4: SSOi Conceptual Data Mode**

The SSOi uses roles, provisioning policies, entitlements and workflows to create, modify, and otherwise manage identity and account objects. These data objects are stored in repositories such as LDAP and Oracle database tables. Section 3.2.2 describes the SSOi data objects with their input and output relationships. Detailed descriptions of each data object are provided later.

### 3.2.2. Database Information

**Table 14: Database Inventory**

Ref	Object	Description	Input Relationship	Output Relationship
①	Identity (Person)	The Identity object is a set of attributes that define an identity in the VA. Identity traits are correlated and a secure identifier is assigned.	- One Identity	<ul style="list-style-type: none"> <li>- One Identity can be assigned to 0 or more Roles.</li> <li>- One Identity can own 0 or more Accounts.</li> <li>- One Identity has only one security identifier for the lifetime of the identity.</li> </ul>

Ref	Object	Description	Input Relationship	Output Relationship
②	User (Account)	The User (Account) attributes define the login information associated with the access control for a managed resource as well as information deemed necessary to perform the business processes or data synchronization requirements.	<ul style="list-style-type: none"> <li>- One Account is owned by 0 (means orphan account) or one Identity (the base identity to which other accounts are linked).</li> </ul>	<ul style="list-style-type: none"> <li>- A user account is represented by a credential which is used for authorization and access to Services.</li> <li>- Account operations (add, modify, change password, suspend, restore, delete, etc.) follow one or more workflows.</li> </ul>
③	Role	The Role attributes defines the role and the associated privileges that can be assigned to a user.	<ul style="list-style-type: none"> <li>- One Identity can be assigned 0 or more Roles.</li> </ul>	<ul style="list-style-type: none"> <li>- One Role can be members of 0 or more Provisioning Policies.</li> <li>- One Role can participate in 0 or more Entitlement Workflows.</li> </ul>
④	Provisioning Policy	The Provisioning Policy object is a definition of the level of access that may be granted to a managed resource or service to particular membership(s) or Roles. The provisioning policy defines identity reconciliation and identity feed.	<ul style="list-style-type: none"> <li>- One Role can be assigned to 0 or more Provisioning Policies.</li> <li>- Each Provisioning Policy may have 0 or more Roles.</li> </ul>	<ul style="list-style-type: none"> <li>- One Provisioning Policy may define 1 or more Entitlements.</li> </ul>

Ref	Object	Description	Input Relationship	Output Relationship
5	Entitlement	The Entitlement object is a part of the Provisioning Policy that contains the service targets and associated provisioning parameters.	<ul style="list-style-type: none"> <li>- One Provisioning Policy may have 1 or more Entitlements.</li> </ul>	<ul style="list-style-type: none"> <li>- One Entitlement can apply to 0 or more Services. It may also apply to a type of service or all services.</li> <li>- One Entitlement can start 0 or 1 Workflows to govern the creation or modification of accounts on an associated service.</li> </ul>
6	Workflow	The Workflow object represents a business process that is associated with an action or a policy. A workflow implements the steps that are required to approve or reject a request, such as a request to provision a person with a new account.	<ul style="list-style-type: none"> <li>- 0 or 1 Workflow can be started by 0 or more Entitlements.</li> <li>- 0 or more Roles can participate in workflows.</li> <li>- 1 or more Workflows can be started by Identity operations.</li> <li>- 1 or more Workflows can be started by Account operations.</li> </ul>	
7	Service	The Service object is a set of parameters that define a managed resource and associated workflows.	<ul style="list-style-type: none"> <li>- 0 or more Services can be assigned to one or more Entitlements.</li> <li>- Accounts control access to services.</li> <li>- Services can be affected by 1 Identity Policy.</li> <li>- Each Service can be affected by 0 or more password policies.</li> </ul>	

Ref	Object	Description	Input Relationship	Output Relationship
8	User Policy	The User Policy contains the rules by which a user's account is created on a managed resource.		- One user policy can be applied to 0 or more Services.
9	Password Policy	The Password Policy object sets rules that passwords must meet.		- One password policy can be applied to 0 or more Services.

### 3.2.3. User Interface Data Mapping

This section describes and defines the data that will be available for users of the SSOi solution via the user interfaces and stored / retrieved from the database, if applicable. Out-of-the-box screens are not shown.

#### Data Tier:

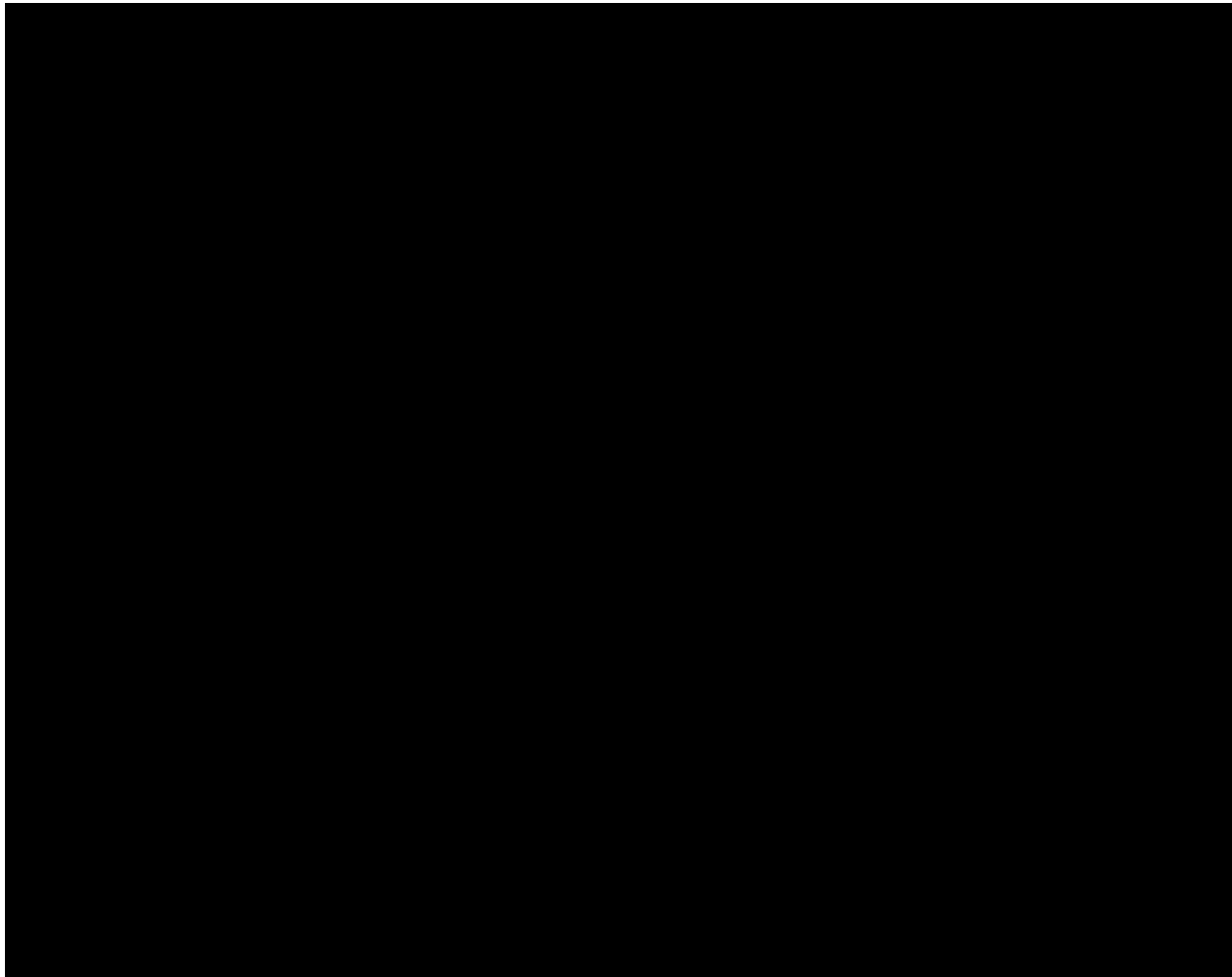
The data tier consists of user stores and a policy store. The policy server uses directory plugins to connect to each user directory for authentication and authorization. Currently SSOi supports Active Directory (AD), Provisioning Store, CSP, IP, and VDS as a user authentication and authorization store. The policy store contains all the policies used to enforce the authentication and authorization requests.

The sections below provide detailed technical flows for the SSOi activity and the associated interactions amongst the system components. The functionality and features provided below focus solely on the requirements directly related to the SSOi activity.

#### 3.2.3.1. SSOi Screen Interface

##### 3.2.3.1.1. IAM CentralLogin

Figure 5 represents the screen that user uses to login to SSOI applications. It has three authentication options userid/password, windows, and PIV authentication. Table 15 describes it.



[Department of Veterans Affairs](#) | [Privacy Policy](#)

Figure 5: IAM Centralized Login Page

Table 15: IAM Central Login Screen Description

Graphical User Interface (GUI) Field	Table (Database Table that field connects to)	Field (Field in Table that the GUI field connects to)	Comments
Userid	AD	sAMAccountName	
password	AD	Password	

#### 3.2.3.1.2. Application Access Denied Page

Figure 6 represents the screen that displays when the user logs in with correct credentials but is not authorized to access the application.





**Figure 6: Application Access Denied Page**

#### **3.2.3.1.3. Failed Login Page**

Figure 7 represents the screen that displays when the user enters invalid credentials.

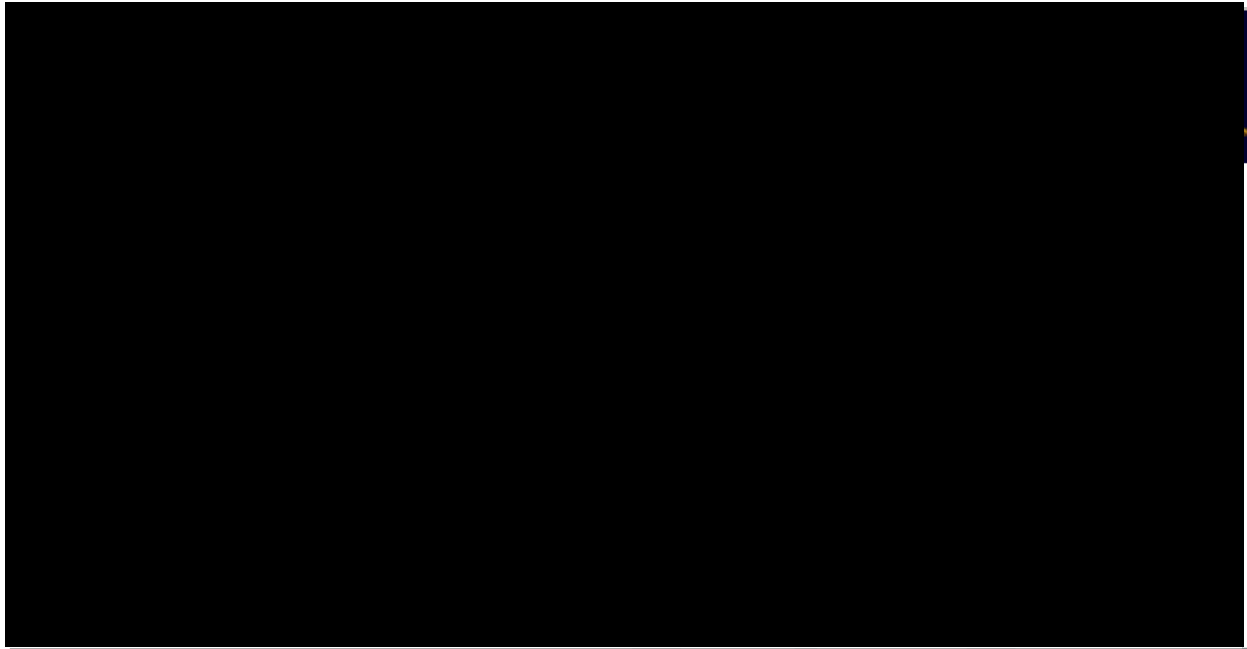


[Department of Veterans Affairs](#) | [Privacy Policy](#)

**Figure 7: Failed Login Page**

#### **3.2.3.1.4. Session Timeout Page**

Figure 8 represents the screen that displays when the user logs into application and leave the browser idle until the timeout.



[Department of Veterans Affairs](#) | [Privacy Policy](#)

**Figure 8: IAM SSO Session Time Out Page**

#### **3.2.3.1.5. SSOi Authenticated Landing Page**

Figure 9 represents the screen that displays when the user clicks on the **Logout** link/button when logged into an SSOi application.



---

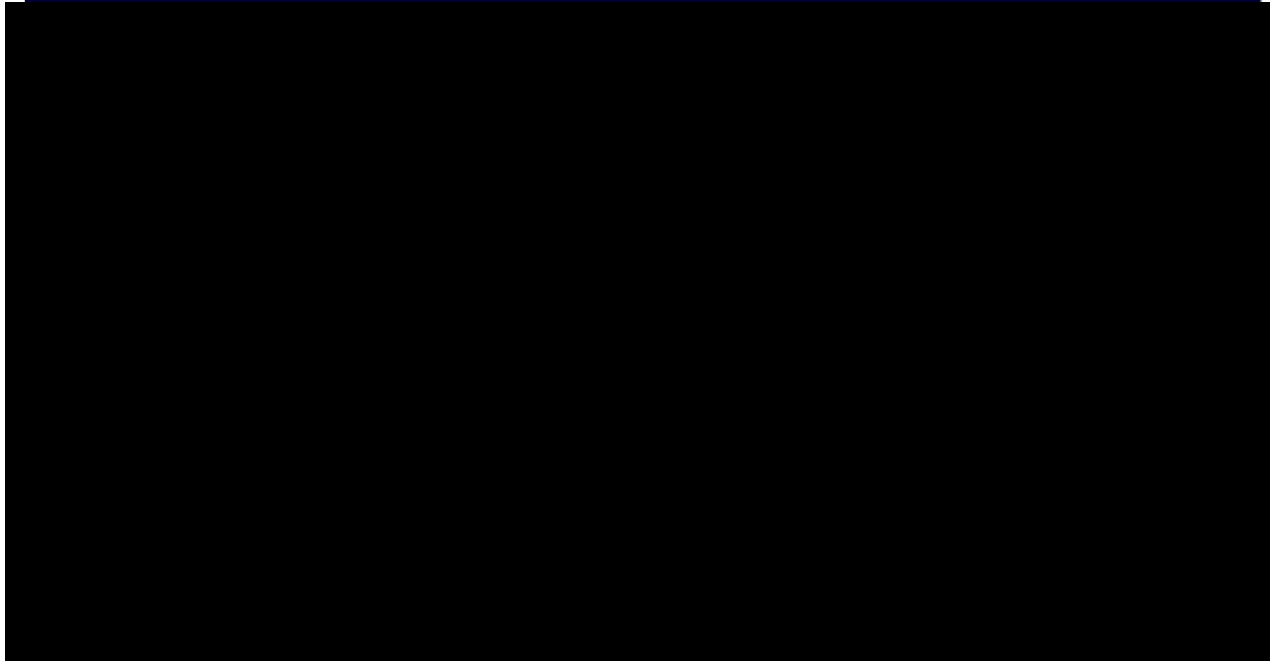
[Department of Veterans Affairs](#) | [Privacy Policy](#)

---

**Figure 9: SSOi Authenticated Landing Page**

#### **3.2.3.1.6. IAM LoggedOff Page**

Figure 10 represents the screen that displays when the user clicks on the **Logout** button on the SSOi Authenticated Landing page.



**Figure 10: IAM Logged Off Page**

### **3.3. Conceptual Infrastructure Design**

The communication between the Terremark data centers in Culpeper, Virginia and Miami, Florida. The VA AcS infrastructure environment is set up at the Terremark data center in Culpeper, Virginia. The alternate site or disaster recovery site for VA AcS operations is the Terremark data center in Miami, Florida.

#### **3.3.1. System Criticality and High Availability**

N/A

#### **3.3.2. Special Technology**

N/A

#### **3.3.3. Technology Locations**

#### **3.3.4. Conceptual Infrastructure Diagram**

The high-level conceptual infrastructure diagram for the VA AcS infrastructure is shown in Figure 11 below.

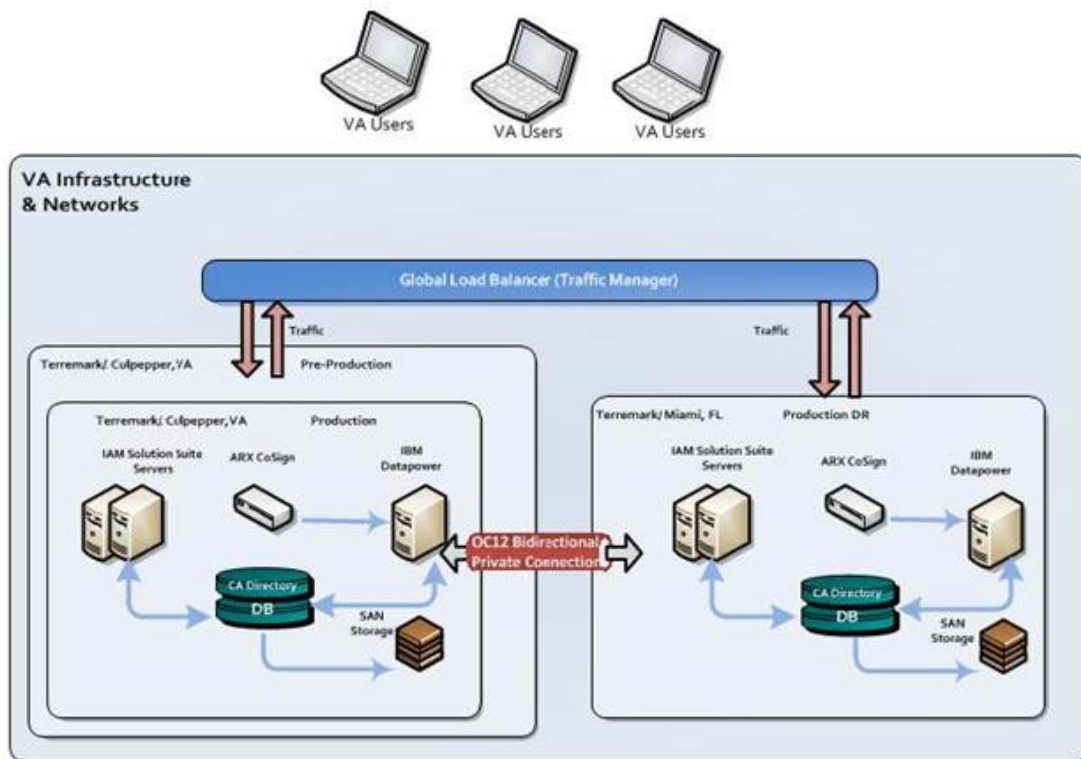


Figure 11: AcS Production Environments

### **Development Environment (DEV) AITC – Austin, TX**

- This environment is utilized by the Development team for initial development of service enhancements, integrations with consuming applications, defect resolution, and unit testing.
- This is a loosely controlled environment for the AcS developers to use. The development team implements and maintains the COTS products, COTS patches, and code.
- System administrators maintain the operating systems and operating system patches.
- Code and configuration is stored in Subversion source control and exported as a build when moving to the next environment.
- The initial setup instructions are fine-tuned; the migration instructions are provided to migrate the code and configuration to the subsequent environments.

### **Software Quality Assurance (SQA) AITC – Austin, TX**

- This environment is utilized by the Development team for integration testing, load, configuration, and quality tests.
- System Administrators install, configure, and operate applications as testing is performed.
- This is a tightly controlled environment and closely resembles the Production architecture. Issues with performance or the setup instructions are performed between Developers and the Administrators responsible for the environment.
- The setup instructions are fine-tuned.

### **Pre-Production – Terremark Culpeper, VA**

- The User Acceptance Test (UAT) for the AcS is performed in this environment.
- This is where performance testing occurs.
- System Administrators install, configure, and operate applications per the fine-tuned setup instructions and provide support as testing is performed.
- Any remaining issues with performance or the setup instructions are worked out with the System Administrators.
- The setup instructions are finalized.
- This is a tightly controlled environment and is as close to identical as possible to the Production environment.

### **Production – Terremark Culpeper, VA**

- The finalized setup instructions are installed.
- The environment is closely monitored.

### **Production Disaster Recovery (DR) – Terremark Miami, FL**

- This site provides hot failover capability so that services and data are maintained in the event of a failure in Production.
- This environment is identical to the Production environment.

- Once the change to Production is verified, the change is implemented in the DR environment.
- The DR environment is in the Terremark Miami, FL data center. The environment is configured with an Active-Passive topology.
- The identity services components like CA IdentityMinder, CA SiteMinder, Provisioning Manager, CA report server, CA UARM would be configured to be on software load balanced on their local site.
- There will be a directory and database synchronized across a private OC-12 connection between both sites. Multiple instances of CA Directory are deployed locally at Terremark Culpeper, VA and remotely at Terremark Miami, FL data centers in a multi-write replication mode. Multi-write replication is a mechanism for replicating updates to a number of instances to maintain that the user stores are synchronized for internal and external users.
- Oracle Data Guard is utilized for database replication from the Production data center at Terremark Culpeper, VA to the disaster recovery data center at Terremark Miami, FL sending the archive logs at an incremental time span asynchronously down to as low as 1 second.

#### **3.3.4.1. Location of Environments and External Interfaces**

This section depicts the AcS 2.0 with many of its internal and external connections exposed. Each subsystem of the infrastructure will be described in the next sections of this document. In each section, these connections will be described and an internal breakdown of the components will also be shown.

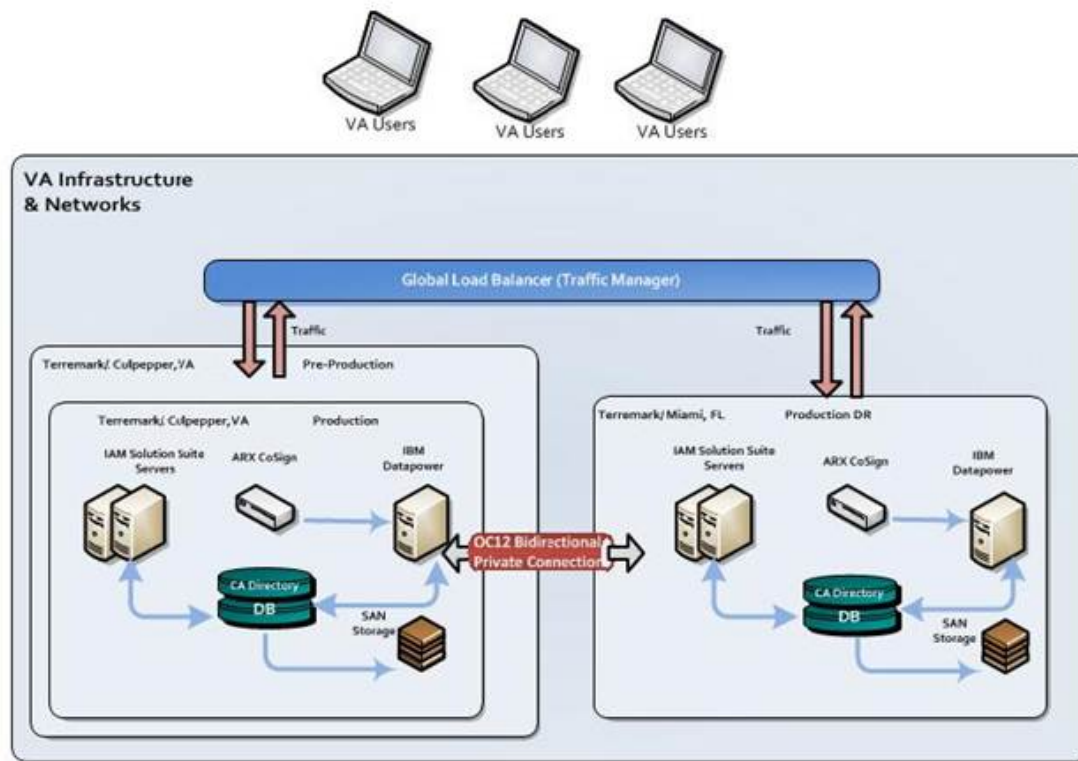




**Figure 12: Sample Conceptual Networks and Environments**

The high-level conceptual infrastructure diagram for the VA AcS infrastructure is shown in Figure 13 below. The diagram also depicts the communication between the Terremark data centers in Culpeper, Virginia and Miami, Florida. The VA AcS infrastructure environment is set

up at the Terremark data center in Culpeper, Virginia. The alternate site or disaster recovery site for VA AcS operations is the Terremark data center in Miami, Florida.



**Figure 13: AcS Production Environments**

### **Development Environment (DEV) AITC – Austin, TX**

- This environment is utilized by the Development team for initial development of service enhancements, integrations with consuming applications, defect resolution, and unit testing.
- This is a loosely controlled environment for the AcS developers to use. The development team implements and maintains the COTS products, COTS patches, and code.
- System administrators maintain the operating systems and operating system patches.
- Code and configuration is stored in Subversion source control and exported as a build when moving to the next environment.
- The initial setup instructions are fine-tuned; the migration instructions are provided to migrate the code and configuration to the subsequent environments.

### **Software Quality Assurance (SQA) AITC – Austin, TX**

- This environment is utilized by the Development team for integration testing, load, configuration, and quality tests.
- System Administrators install, configure, and operate applications as testing is performed.

- This is a tightly controlled environment and closely resembles the Production architecture. Issues with performance or the setup instructions are performed between Developers and the Administrators responsible for the environment.
- The setup instructions are fine-tuned.

### **Pre-Production – Terremark Culpeper, VA**

- The User Acceptance Test (UAT) for the AcS is performed in this environment.
- This is where performance testing occurs.
- System Administrators install, configure, and operate applications per the fine-tuned setup instructions and provide support as testing is performed.
- Any remaining issues with performance or the setup instructions are worked out with the System Administrators.
- The setup instructions are finalized.
- This is a tightly controlled environment and is as close to identical as possible to the Production environment.

### **Production – Terremark Culpeper, VA**

- The finalized setup instructions are installed.
- The environment is closely monitored.

### **Production Disaster Recovery (DR) – Terremark Miami, FL**

- This site provides hot failover capability so that services and data are maintained in the event of a failure in Production.
- This environment is identical to the Production environment.
- Once the change to Production is verified, the change is implemented in the DR environment.
- The DR environment is in the Terremark Miami, FL data center. The environment is configured with an Active-Passive topology.
- The identity services components like CA IdentityMinder, CA SiteMinder, Provisioning Manager, CA report server, CA UARM would be configured to be on software load balanced on their local site.
- There will be a directory and database synchronized across a private OC-12 connection between both sites. Multiple instances of CA Directory are deployed locally at Terremark Culpeper, VA and remotely at Terremark Miami, FL data centers in a multi-write replication mode. Multi-write replication is a mechanism for replicating updates to a number of instances to maintain that the user stores are synchronized for internal and external users.
- Oracle Data Guard is utilized for database replication from the Production data center at Terremark Culpeper, VA to the disaster recovery data center at Terremark Miami, FL sending the archive logs at an incremental time span asynchronously down to as low as 1 second.

### 3.3.4.2. Conceptual Production String Diagram

Figure 14 provides a logical view of the AcS 2.0 components.

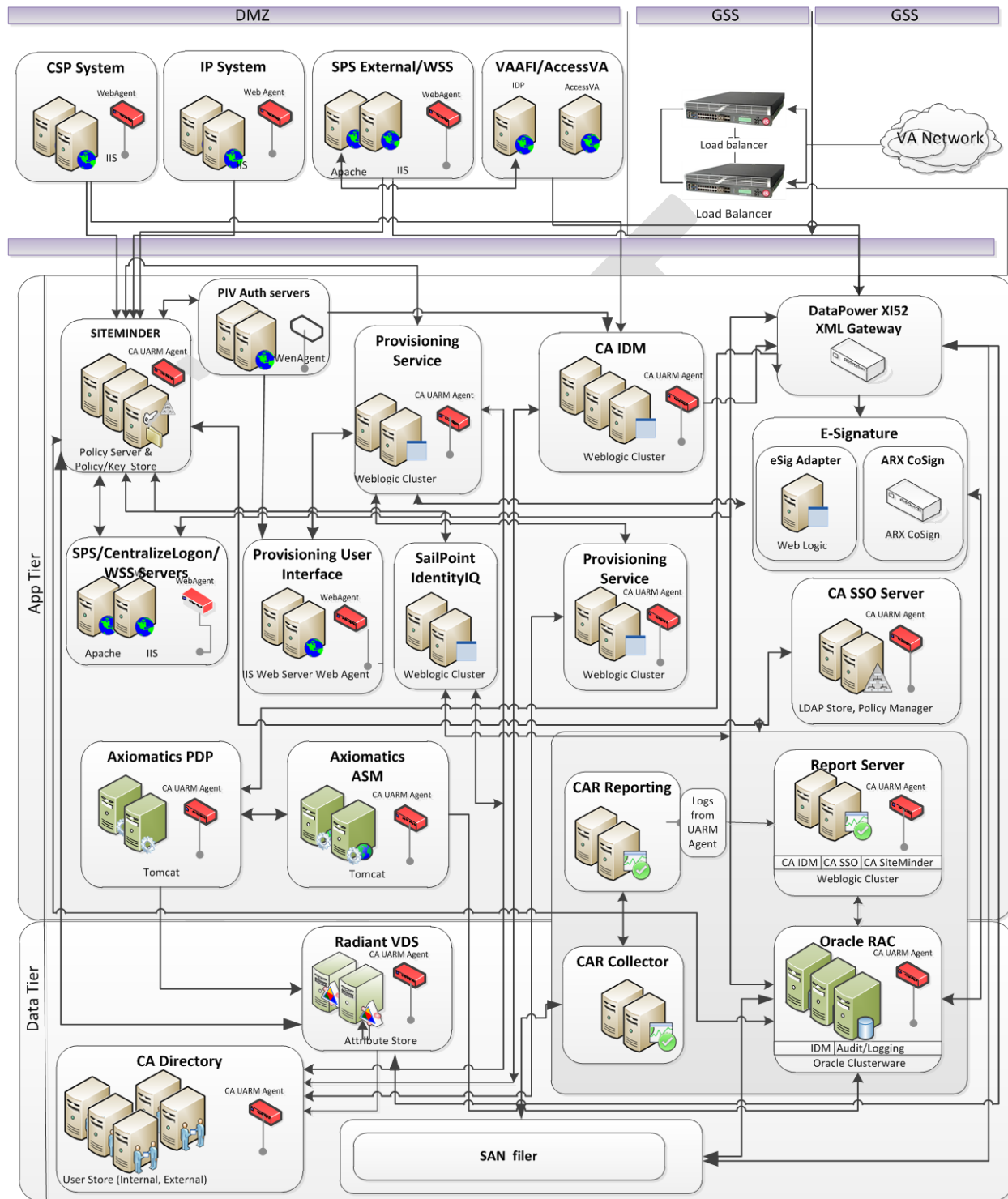


Figure 14: Logical Network String Diagram

## 4. System Architecture

The AcS 2.0 system architecture includes the hardware, software, and communication architectures. The hardware architecture describes the physical components needed in the system and their relationship to one another. The software architecture describes the software products, components, and code needed to provide the AcS 2.0. The communication architecture describes the connection and security requirements needed between the hardware components.

### 4.1. Hardware Architecture

The following diagram, Figure 15, shows the AcS 2.0 hardware architecture and network topology.

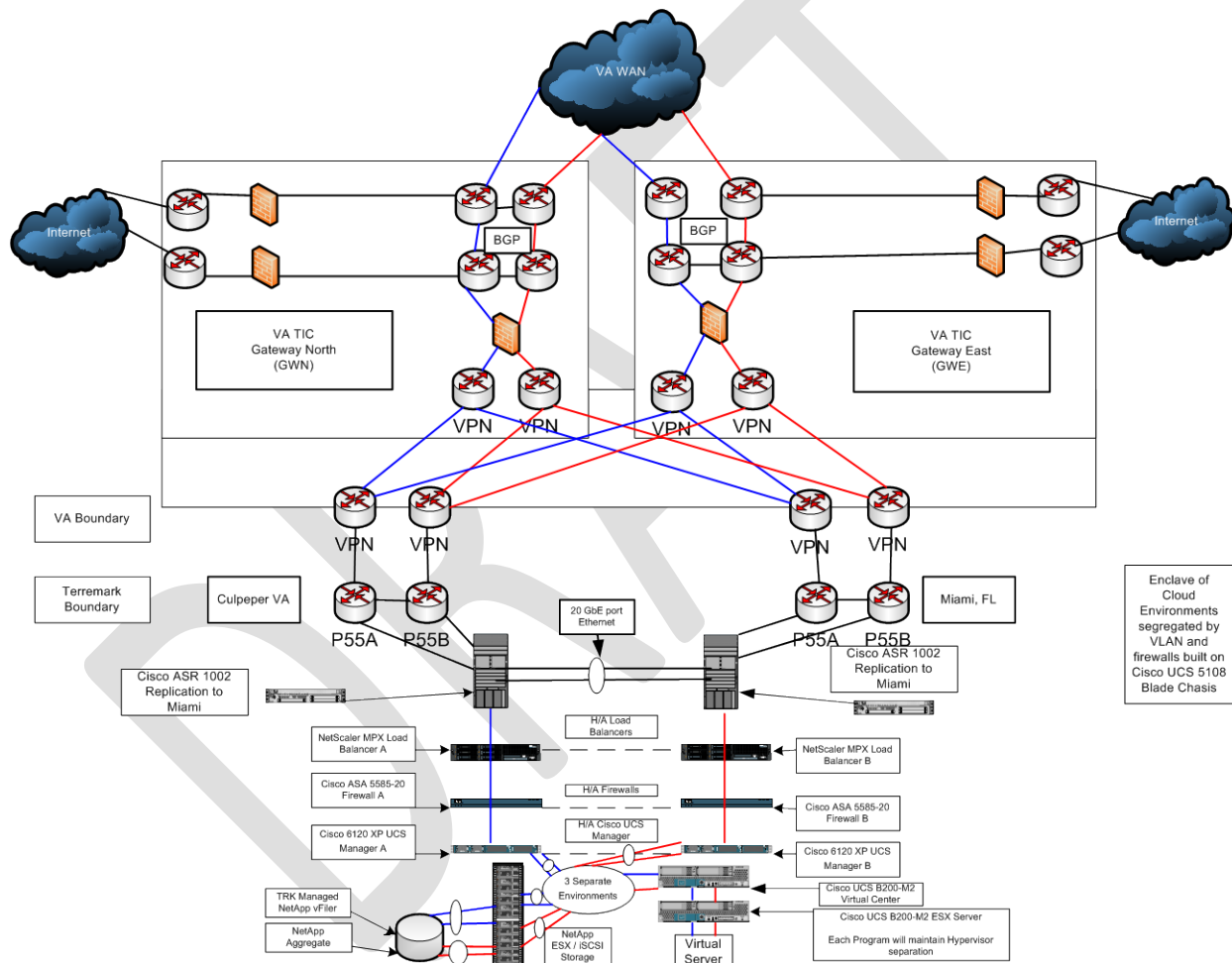


Figure 15: Network Communication Architecture

The following table provides information for the hardware appliances used for the VA SSOi solution.

**Notes:**

- All environments currently use X152 DataPower
- Production and DR are using X152 DataPower.

**Table 16: Hardware Appliance**

Hardware Appliance	Descriptions	High Availability (HA)
IBM DataPower	A critical component of AcS infrastructure to securing web service message flows as a proxy using IBM DataPower Appliance	For High Availability configuration, the DataPower X152 appliances will reside behind a Citrix Netscaler. This setup will have no effect on the existing DataPower configurations, as each transaction will be independent and processed separately by each DataPower appliance. The load balancer will serve as a reverse-proxy to distribute network traffic. The goal is to improve the overall burden of a single machine by enabling an industry standard algorithm.

The uniform resource locators (URLs) for SSOi for production, pre-production and SQA are provided in the table below. The AcS components residing in the DMZ are the external facing web servers that contain the CSP pages and federation components. These components will be load balanced by the Citrix Netscalers located in the Terremark GSS. DataPower, along with the remaining AcS application components, will reside in the GSS. The following table provides details on the AcS 2.0 machines such as ports, URLs, protocols hostnames for each application in every environment.

**Table 17: Virtual Machines and Appliances  
SQA (AITC)**

Application	Number of VMs	Number of Physical Servers	Hostname
CSP, IP, Federation Services WebUI, SPS, WSS (IIS- Single instance on each, Tomcat)	5	N/A	
IdentityMinder supporting (Credential Service Provider and Identity Proofing) WebLogic cluster Admin service on primary node	3	N/A	
Centralized Logon page, SPS, WSS (IIS- Single instance on each)	1	N/A	
PIV Authentication Handler (IIS ) Single instance on each No OCSP responder or CRL configuration	1	N/A	
IdentityMinder support (Provisioning Service) (WebLogic) Admin service on primary node	2	N/A	
Provisioning WebUI (IIS) Single instance on each	2	N/A	
Provisioning Server	2	N/A	

Application	Number of VMs	Number of Physical Servers	Hostname
CA Directory (CSP and IP)	3	N/A	
CA Directory (Provisioning)	2	N/A	
CA SSO Server	2	N/A	
CA SSO	2	N/A	
CA UARM (Tomcat)	4	N/A	
CA Report Server (WebLogic)	2	N/A	
CA SiteMinder (WebLogic) includes CA Directory instance for SiteMinder Admin service on primary node Admin UI on primary node	3	N/A	
Axiomatics PDP (Tomcat)	1	N/A	



Application	Number of VMs	Number of Physical Servers	Hostname
Axiomatics ASM/PAP (Tomcat)	1	N/A	
Axiomatics Policy Auditor	1	N/A	
Radiant Logic VDS	N/A	1	
Oracle RAC	N/A	2	
DataPower XI50 (Appliance)	N/A	2	
ARX CoSign (Appliance)	N/A	1	
eSig WebLogic Servers Admin service on primary node	2	N/A	
Role manager (SailPoint) servers (WebLogic) Admin service on primary node	2	NA	

**Pre-Production (Terremark Culpeper, VA)**

Application	Number of VMs	Number of Physical Servers
CSP, IP ,Federation Services WebUI/SPS/WSS (IIS, Tomcat) Single IIS instance on each	4	N/A

Application	Number of VMs	Number of Physical Servers	Hostname
Centralized Logon page, SPS, WSS (IIS) Single IIS instance on each Web403/404 will replace 413/414	2	N/A	
PIV Authentication Handler (IIS) Single IIS instance on each	2	N/A	
IdentityMinder supporting (Credential Service Provider and Identity Proofing) (WebLogic) Admin service on primary node	2	N/A	
IdentityMinder support (Provisioning Service) (WebLogic) Admin service on primary node	3	N/A	
Provisioning WebUI (IIS ) Single IIS instance on each	2	N/A	
Provisioning Server	2	N/A	
CA Directory (CSP and IP)	2	N/A	
CA Directory (Provisioning)	2	N/A	

Application	Number of VMs	Number of Physical Servers	Hostname
CA SSO Server	2	N/A	
CA UARM	3	N/A	
CA Report Server	1	N/A	
CA SiteMinder (WebLogic) includes CA Directory instance for SiteMinder Admin service on primary node Admin UI on primary node	3	N/A	
Axiomatics PDP (Tomcat)	2	N/A	
Axiomatics ASM/PAP (Tomcat)	1	N/A	
Radiant Logic VDS	N/A	1	
Oracle Database	N/A	2	
DataPower XI52 (Appliance)	N/A	N/A	
ARX CoSign (Appliance)	N/A	N/A	
eSig WebLogic Servers Admin service on primary node	2	N/A	

Application	Number of VMs	Number of Physical Servers	Hostname
Role manager (SailPoint) servers (WebLogic) Admin service on primary node	2	N/A	

**Production (Terremark Culpeper, VA)**

Application	Number of VMs	Number of Physical Servers	
CSP, IP, Federation Services WebUI,SPS,WSS (IIS ) Single IIS instance on each	4	N/A	v i v i v i v i
Centralized Logon page, SPS, WSS (IIS , Tomcat) Single IIS instance on each	2	N/A	v i v i
PIV Authentication Handler (IIS ) Single IIS instance on each No OCSP or CRL	2	N/A	v i v i
IdentityMinder (CSP and IP) (WebLogic) Admin service on primary node	2	N/A	v v
IdentityMinder (Provisioning) (WebLogic) Admin service on primary node	3	N/A	v v v c

Application	Number of VMs	Number of Physical Servers	Hostname
Provisioning WebUI (IIS) Single IIS instance on each	2	N/A	
Provisioning Server	2	N/A	
CA Directory (CSP,IP)	2	N/A	
CA Directory (Provisioning)	2	N/A	
CA SSO Server	2	N/A	
CA UARM	3	N/A	
CA Report Server (WebLogic)	1	N/A	
CA SiteMinder (WebLogic) includes CA Directory instance for SiteMinder Admin service on primary node Admin UI on primary node	3	N/A	
Axiomatics PDP (Tomcat)	2	N/A	

Application	Number of VMs	Number of Physical Servers	Hostname
Axiomatics ASM/PAP (Tomcat)	1	N/A	
Radiant Logic VDS	N/A	2	
Oracle Database	N/A	2	
DataPower XI52	N/A	N/A	
ARX CoSign	N/A	N/A	
eSig WebLogic Servers	2	N/A	
Role manager (SailPoint) servers (WebLogic)	2	N/A	

**DR (Terremark Miami, FL)**

Application	Number of VMs	Number of Physical Servers
CSP,IP,Federation Services WebUI (IIS)	4	N/A
Centralized Logon page, SPS, WSS (IIS, Tomcat)	2	N/A

Application	Number of VMs	Number of Physical Servers	Hostname
PIV Authentication Handler (IIS)	2	N/A	
IdentityMinder (CSP) (WebLogic)	2	N/A	
IdentityMinder (Provisioning) (WebLogic)	3	N/A	
Provisioning WebUI (IIS)	2	N/A	
Provisioning Server	2	N/A	
CA Directory (CSP and IP)	2	N/A	
CA Directory (Provisioning)	2	N/A	
CA SSO Server	2	N/A	
CA UARM	3	N/A	

Application	Number of VMs	Number of Physical Servers	Hostname
CA Report Server (WebLogic)	1	N/A	
CA SiteMinder (WebLogic) includes CA Directory instance for SiteMinder	2	N/A	
Axiomatics PDP (Tomcat)	2	N/A	
Axiomatics ASM/PAP (Tomcat)	1	N/A	
Radiant Logic VDS	N/A	2	
Oracle Database	N/A	2	
DataPower XI52 (Appliance)	N/A	N/A	
ARX CoSign (Appliance)	N/A	N/A	
eSig WebLogic Servers	2	N/A	

## 4.2. Software Architecture

The following diagram shows the complete software architecture of the VA AcS 2.0.



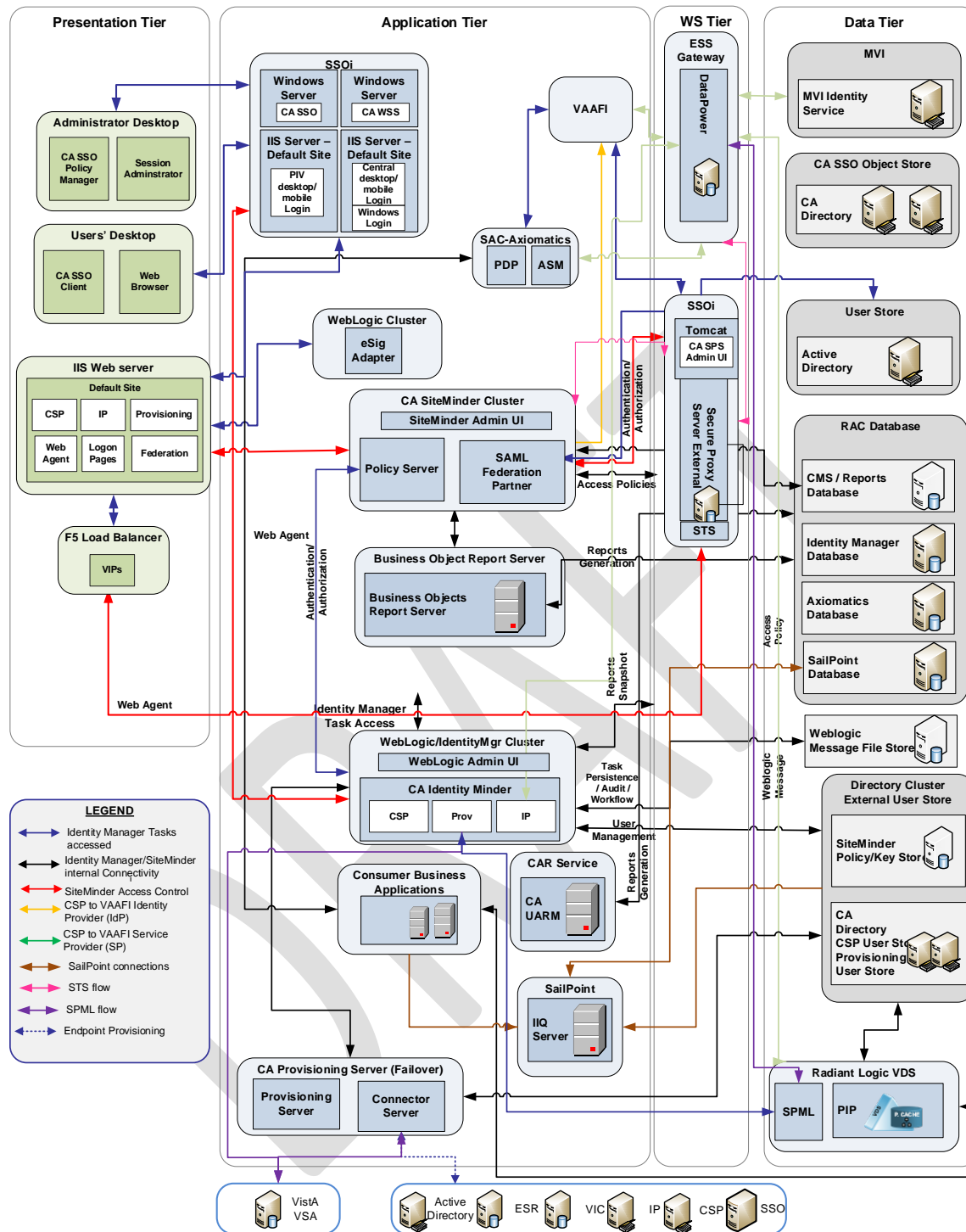


Figure 16: Software Architecture

The following table describes the AcS 2.0 products and versions for SSOi.

**Table 17: SSOi Products and Versions**

Products	Abbreviation	Product Version/Release
CA Single Sign On	CA SSO	12.1 / R12.1
Oracle Database	-	11gR2
CA Directory	CA LDAP	12.0 SP7
CA SiteMinder	CA SM	SM r12.51
CA Option Pack for Federation	-	SM r12.51
Login Page	LP	ASP.NET 4
CA Secure Proxy Server	CASPS	SM r12.51
New Atlanta ServletExec	ServletExec	ServletExec 6.0.0.2_39

The following table provides information about the software components.

**Table 18: Software Components**  
**Oracle Database 11gR2**

The shared database environment will maintain the following table spaces required for the components of the AcS implementation. Database High Availability and Data Guard to synchronize and replicate a HOT Oracle database environment to TerreMark Miami, FL.

Characteristic	Description
<b>Database Table spaces</b>	<p>4 Data Table spaces: PROVIDM_DATA, CSPIPIDM_DATA, CASMAUDIT_DATA, ESIGAUDIT_DATA, VDSAUDIT_DATA, SACASM_DATA</p> <p>3 Index Table spaces: PROVIDM_INDX, CSPIPIDM_INDX, CASMAUDIT_INDX</p> <p>Users</p> <p>Temp</p> <p>Rollback</p> <p>Undo</p>

Characteristic	Description
<b>High Availability</b>	For the AcS 2.0, database high availability is critical. A database outage can cause a multitude of errors to occur on the application side, thereby nullifying the high availability configurations on the application itself. It was planned for Raw Devices to be utilized by Oracle Automatic Storage Management (ASM) file system, working as the volume manager, overseeing the clusterware file systems. ASM, attached by each node, exposes the existing pool of storage and makes it available as an interface for the Oracle database files. The ASM is supported by Oracle Clusterware. If a single Oracle instance on a node fails, the ASM and database instances on the surviving nodes are designed to automatically failover. Due to the load dependency on the ASM file system storage, mirroring is needed to provide high availability.

### CA Directory

The CA Directory servers are a shared resource for the AcS 2.0. The CA Directory infrastructure will be configured in a multi-master replication configuration. The CA Directory comprises of various instances elaborated as follows.

**Note:** CA Directory structure as applicable for each of the directory instance specific to a release and will be provided in each release. The holistic view of the CA Directory structure is provided in Software Detail Design Sections.

Characteristic	Description
<b>Directory Instances</b>	User store CA IdentityMinder for CSP solution and Provisioning services, Policy and Key store for CA SiteMinder for CSP service Object/policy store for CA SSO for SSOi services.

Characteristic	Description
<b>High Availability</b>	<p>There will be a master write server for each directory. The other supporting directories will be read directories.</p> <p>The CA Directory will provide intelligent and transparent chaining of queries to distributed servers. It performs transparent routing to re-route requests in the event of failure on a particular CA Directory server. The CA Directory router DSA distributes incoming requests evenly among DSAs in the same site. The clients accessing router dsa are configured to maintain the list of AcS CA Directory router DSA's and the failover occurs from the client's end. This improves performance, allowing CA Directory's replication mirroring to provide synchronized in real-time and consistent servers.</p> <p>CA IdentityMinder, CA SSO, and CA SiteMinder will leverage the directories through a round robin load balancing configuration. Multiwrite-DISP replication is a replication scheme that uses multiwrite replication for real-time updates and DISP for recovery. By default, the Directory System Agent (DSA) is configured for multiwrite-DISP replication. This replication scheme combines the efficiency of multiwrite when DSAs are online (real-time updates), with the robustness of DISP to allow DSAs to recover after being offline (recovery).</p> <p>The DSA uses its routing capabilities to distribute requests evenly between systems while data replication keeps the data synchronized.</p>

#### **Web Tier – IIS Web Server**

The Web Tier consists of IIS web servers that provide reverse proxy and federation to the applications.

Characteristic	Description
<b>IIS Web Server Instances</b>	Central Login /PIV Login application is hosted on IIS server
<b>High Availability</b>	<p>IIS Web Servers are used by the centralized logon, PIV Auth and Federation servers to support multiple services. They will be protected by the SiteMinder Option Pack (Federation), PIV Authentication Servers, and Centralized Logon Server Page.</p> <p>There are two IIS web servers required for PIV, Federation, and Centralized logon.</p>

#### **Application Tier – WebLogic Application Server**

The application tier for the Provisioning service is made up of a cluster of WebLogic application servers. The Application Tier is a shared environment for hosting application components. The AcS-related applications hosted are listed below. The Report Server instance is a Business Objects environment that provides reporting services for Access Services. The CA Report server

(SAP Business Objects XI R3.1 SP3) that constitutes the Reporting Infrastructure is hosted on a WebLogic cluster.

Characteristic	Description
<b>WebLogic Instances</b>	CA SiteMinder Admin UI
<b>High Availability</b>	<ul style="list-style-type: none"> <li>The SiteMinder Admin UI consists one local Single node WebLogic instance available in primary SiteMinder policy server. CA product has a limitation that Admin UI cannot automatically failover. But the High availability is achieved by configuring it to manage multiple Policy Servers including Primary and secondary servers so that alternate server can be used in case of unavailability of the primary server.</li> </ul>

### CA SiteMinder

CA SiteMinder is an integral component of Access Services solution, providing CSP solution federation capabilities to integrate with VAAFI. CA SiteMinder is also utilized to protect the CA IdentityMinder application. CA SiteMinder is comprised of the following components.

Characteristic	Description
<b>Subcomponents</b>	<p><b>SiteMinder Policy Server:</b> The Policy Server provides advanced authentication and password services to protected applications such CA IdentityMinder. The policy server communicates with the CA Directory, which stores the required policy objects, key objects and user data to provide federation services as well.</p> <p><b>Secure Proxy Server:</b> The Secure Proxy Server provides agentless web based integration as well as provides secure web services calls supported by centralized policies defined in SiteMinder.</p> <p><b>Policy/Key Store:</b> The policy store / key store is CA Directory instance which stores configured policies, objects and keys required by CA SiteMinder.</p> <p><b>Web Agents:</b> The agents to be installed on the web server protect the resources.</p> <p><b>Admin User Interfaces:</b> The Admin UI hosted in admin VLAN to manage CA SiteMinder and policies.</p> <p><b>FSS Administrative UI:</b> The Federation Admin UI is hosted on same VM as CA SiteMinder to manage CA SiteMinder for federation configuration. It requires a web server as provided in the web tier above.</p> <p><b>Audit:</b> The SiteMinder Audit sub-system stores audit events for SiteMinder authentication and authorization transactions. The data is stored in the oracle database and is secured from modifications.</p>

Characteristic	Description
<b>High Availability</b>	<p>CA SiteMinder will be installed on three (3) servers. These servers will be load balanced using the native CA SiteMinder software configuration.</p> <p>The CA SiteMinder web agent HA is depending on Application Web server HA. If there are multiple IIS instance for the protected application, webagent is also on HA as it is installed on individual Web server. Webagent configured to talk to all the SiteMinder Policy Server available and internally it load balance the request in a round robin mode.</p>

### CA SSO

The CA SSO server, Authentication services, and CA SSO desktop client enable the SSOi services for desktop single sign on usage. The authentication services communicate with the user store to provide credentials and authenticate the user to the SSOi solution. It also interacts with the CA Directory to maintain user logon information. The CA SSO is installed and configured in FIPS only mode as approved by TRM.

Characteristic	Description
<b>Subcomponents</b>	<p><b>CA SSO Server:</b> The CA SSO Server is the main component of the CA SSO suite. It manages resources and provides services to the CA SSO Client. A CA SSO server farm will be created for clustering. The data on each server can then be replicated to the servers contained within the farm.</p> <p><b>CA Policy Manager:</b> The Policy Manager is the user interface to manage the SSO Server and the data stores (CA Access Control and CA Directory). It is usually installed on an administrator's workstation for remote management of SSO Servers using TCP/IP.</p> <p><b>CA SSO Desktop Client:</b> The CA SSO Client is the desktop component of CA SSO must be installed on every end-user workstation that requires SSOi solution.</p>
<b>High Availability</b>	<p>A number of components for CA SSO that will be configured for High Availability.</p> <p><b>CA SSO Server:</b> A CA SSO server farm will be created. It is a system of two networked CA SSO Servers. The data on each server will be replicated to servers in the farm. The Terremark Culpeper, VA site and Terremark Miami, FL site will contain two (2) servers in each site to create the server farm. One server in each server farm is assigned as hub. The hub server is the server that receives the incoming updates from external server farms, and propagates the data to its peers within its own server farm to achieve failover between sites. The load is be balanced between two servers in the server farm using the Citrix NetScaler load balancer.</p> <p><b>CA Policy Manager:</b> The Policy Manager is the user</p>

Characteristic	Description
	<p>interface that enables the management of the SSO Server and the data stores (CA Access Control and CA Directory). It is installed on an administrator's workstation for remote management of SSO Servers using TCP/IP.</p> <p><b>CA SSO Desktop Client:</b> The CA SSO Client is the desktop component of CA SSO. It must be installed on every end-user workstation that requires SSOi solution. The CA SSO Client has built-in failover between the CA SSO Client and the authentication host, and between the CA SSO Client and CA SSO Server. The fully qualified domain name (FQDN) for both Terremark Culpeper, VA server farm and Terremark Miami, FL server farm is defined in the CA SSO client configuration for built-in failover.</p>

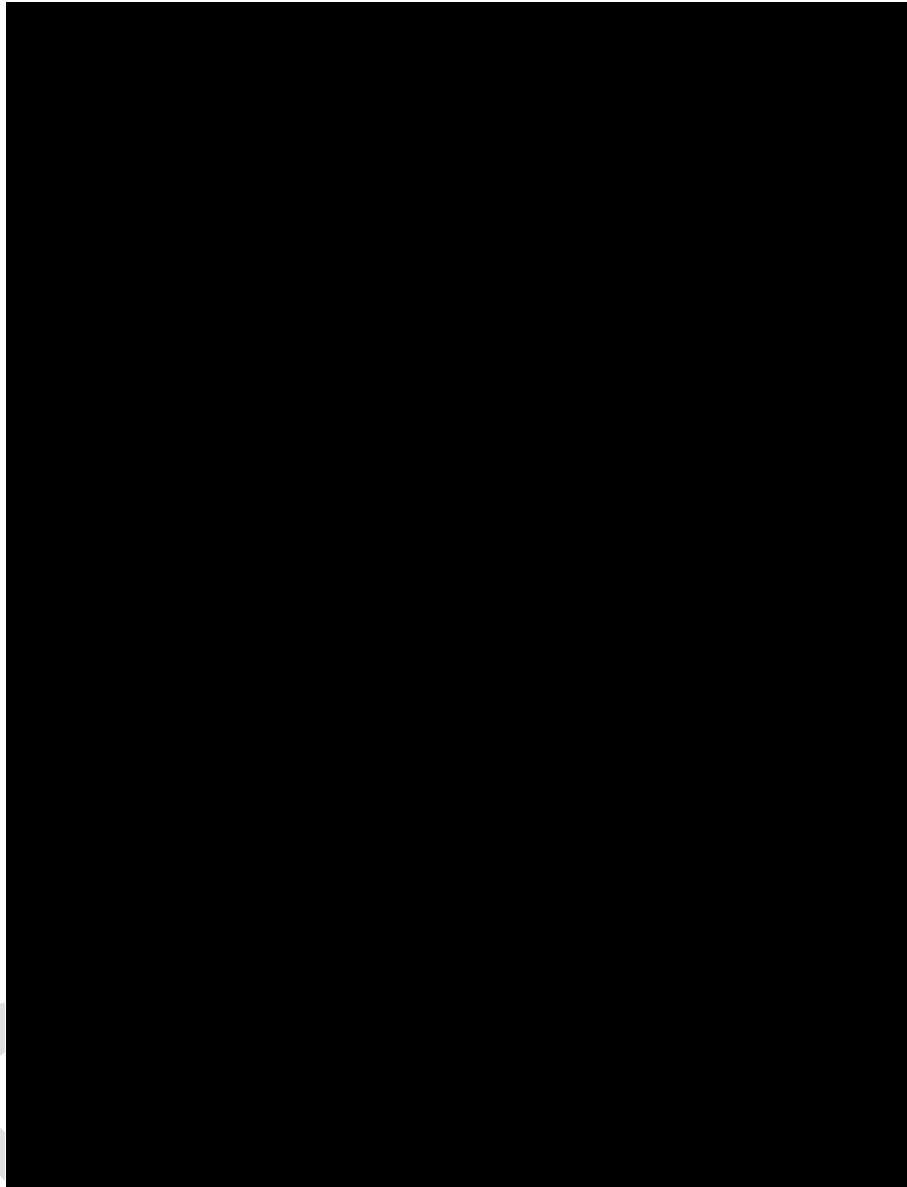
The following table defines the programming languages used for development within the VA AcS 2.0.

**Table 19: Programming Languages**

Programming Languages	Definition/Description
Java	Java language was used to develop custom class/jar file for IdentityMinder Business Logic Task Handler BLTH.
C#/.NET	C#/.NET for development of custom applications.
HTML / DHTML	Provides basic web page language.
ASP.NET	Active Server Pages for development of web pages. The SiteMinder login.fcc page was customized using this language.
XML	Common configurations are stored as XML files.
JavaScript	Scripting language.
RegEX	Regular Expression.

### 4.3. Network Architecture

The following diagram depicts the communication channels between the different AcS components and protocols used.



**Figure 17: AcS Network Security Topology**

### **4.3.1. Communication Channel Security**

In order for AcS system components to communicate internally (within the boundaries of AcS) or externally in a secure manner, the supporting software PKI infrastructure components need to be configured. Every Hypervisor Virtual machine, physical server, hardware or software appliance, and applicable other AcS-exposed service is issued a VA internal or commercial (publicly trusted) CA signed server certificate and configured for runtime use. If auto-enrollment service for PKI certificates is not available for any of the AcS' virtual or physical system components, certificate signing requests (CSR) (in the form of Certificate Signing Request [CSR] file) will be generated for each component and sent to the VA PKI helpdesk at [\[redacted\]pki\[redacted\]sslts/](#). The following lists the server certificates for the AcS components:



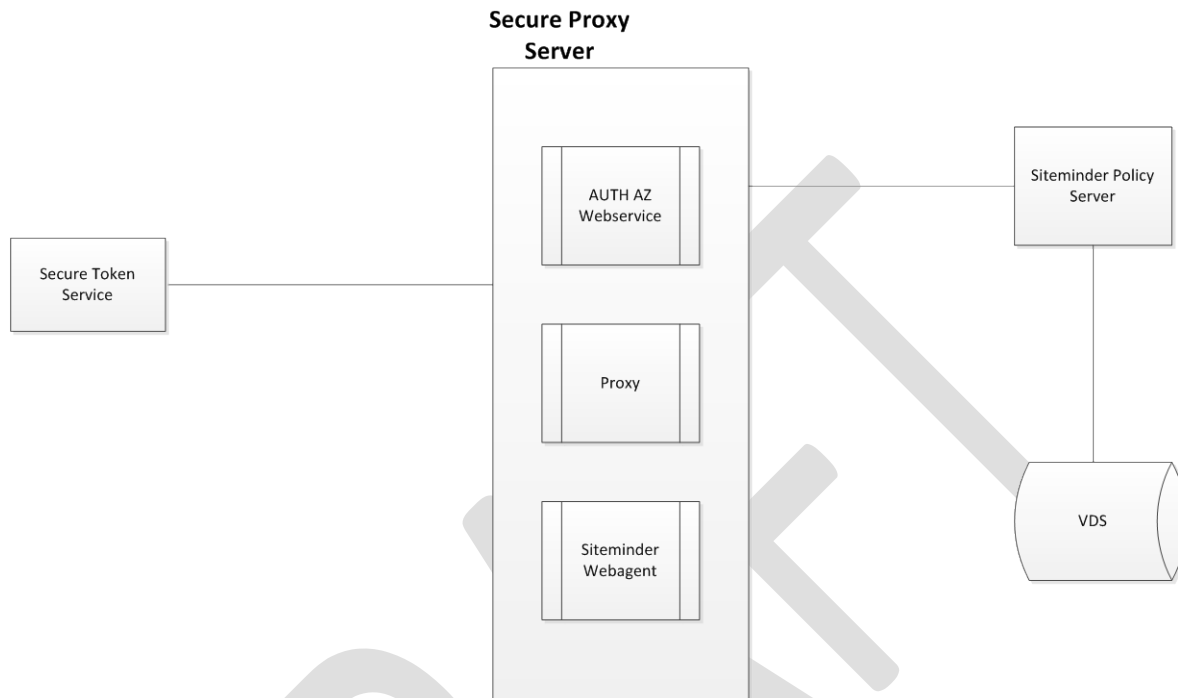
- The publicly accessible AcS URLs requiring user authentication are protected by SSL/TLS encryption. The client SSL/TLS connections will be terminated at the Citrix NetScaler load balancer and subsequently proxied to the appropriate AcS DMZ component.
- The SSL/TLS certificates assigned to the AcS' external access URLs were requested from and issued by VAs commercial (publicly trusted) certificate authority - GTE Cybertrust
- The AcS native components communicating TCP/IP layer secured FIPS mode of encryption.
- VA Internal User Access
- AcS Infrastructure Security

#### **4.3.2. AcS Intercomponent Communications**

Please refer to Section 6.2.1.12.4.

## 4.4. Service Oriented Architecture / ESS

SSOi Service-oriented architecture delivers login and validation of token services to the applications as a service to either end-user applications or other services. The figure below shows the high-level SSOi service-oriented architecture.



**Figure 18: High-Level SSOi Service-Oriented Architecture**

- **Secure Proxy Server (SPS):** The SPS provides proxy services for application authentication and authorizations. SPS enables mobile applications in a similar way it does for web applications by issuing mini cookies. These cookies are compliant with native mobile applications and browsers. A web application can also call the SSOi Authentication and Authorization web service interface to authenticate and validate the SSOi sessions via SOAP and REST messages.
- **Security Token Store Service (STS):** DataPower acts as the STS store that supports token translation requests from application end, where it supports WS-Trust token as input request having user's SiteMinder session as part of request. STS store validates Token request and will returns the standard user attributes as a part of response specification
- **Virtual Directory Server (VDS):** Virtual Directory service produces single view of data for any given user, by retrieving the attributes from multiple sources like AD, LDAP.

## 4.5. Enterprise Architecture

Please refer to the COTS Product Roadmap on the [AcS TSPR](#) site.

## 5. Data Design

This section outlines the design of the database management system (DBMS) and non-DBMS files associated with the SSOi solution as well as the data security implementation.

### 5.1. DBMS Files

SSOi uses Oracle 11gR2 Database and CA Directory for persistent data storage. The Oracle database “ACSDb” is created and used for the following purposes:

- CA SiteMinder audit schema is built during the installation via COTs pre-bundled scripts to store audit information
- Similarly, CA Directory will be used for the following purposes:
  - CSP User Store is built to store user attributes for external VA users.
  - Provisioning User Store is built to store user attributes for users who are requesting access.
  - SiteMinder Policy Store is built to store policy and configurations of SiteMinder.

**Table 20: Database File System**

Table Spaces	Data Files
CASM_DATA	+ORADATA/acsdbs/datafile/casm_data
CASM_INDX	+ORADATA/acsdbs/datafile/casm_indx

### 5.2. Non-DBMS Files

This section is not applicable.

### 5.3. Data View

This section is not applicable.

## 6. Detailed Design

This section describes the design for the SSOi solution and its activities in detail.

### 6.1. Hardware Detailed Design

The sections below provide the hardware information for each activity in the VA SSOi. The following table displays the sizing, network, Operating System, and number of Virtual Machines required to be deployed across AcS activities:

**Note:** Applications will be deployed on virtual machines except Oracle (SQA), IBM DataPower, and ARX CoSign.



## 6.2. Software Detailed Design

This section provides final detailed information associated with the design of SSOi solution activity and the associated functionality.

### 6.2.1. Conceptual Design

The existence of multiple applications accessed by the VA user community creates a problem where users have to remember multiple passwords for multiple applications. Each application is using disparate login capabilities that commensurate with the risk-level associated with the specific application security requirements. The SSOi activity addresses these identified issues of multiple passwords, re-authentication and security challenges by providing seamless authentication from application to application without prompting user's for their credentials again. To simplify users' experience, the SSOi activity will provide a common entry point for SSOi enabled applications. The authentication events for users will be logged and audited as required to produce necessary reports.

A detailed view of the SSOi activity is depicted in the following diagram.

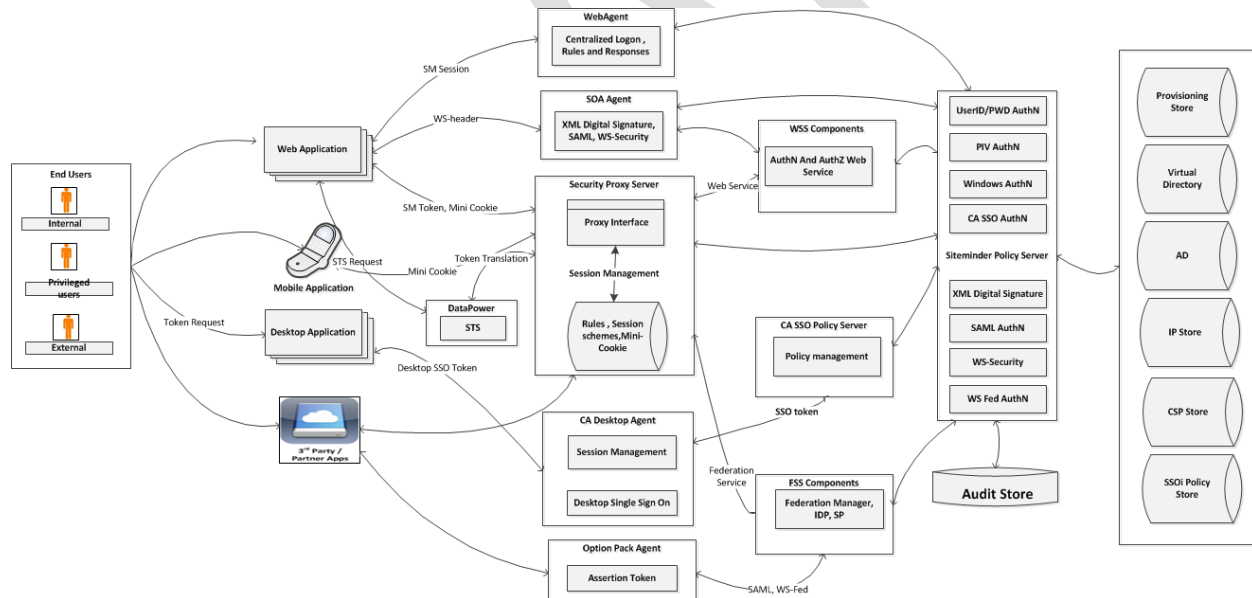


Figure 19: SSOi Detailed Design

#### Enforcement Agents:

The enforcement agent enforces policies and protects the end applications.

- **Web Agent:** A Web agent protects web and application containers. Agents are installed on application web servers to intercept authentication requests to determine authorization permissions defined by the access policies.

- **SOA Agent:** This Agent protects the web service endpoints and enforces necessary policies to secure access. SOA Agents provide the capability to read SOAP/REST messages and add / update security headers with a user's SSO session.
- **Secure Proxy Server (SPS):** The SPS provides proxy services for application authentication and authorizations. SPS enables mobile applications in a similar way it does for web applications by issuing mini cookies. These cookies are compliant with native mobile applications and browsers. A web application can also call the SSOi Authentication and Authorization web service interface to authenticate and validate the SSOi sessions via SOAP and REST messages.
- **SiteMinder Admin UI:** The SiteMinder Admin UI is the application that runs on WebLogic. It is used to create/modify the SiteMinder policy server configurations. It writes all the updates to the policy store.
- **Desktop Agent:** Desktop agent provides SSO functionality to desktops / thick clients and provides user access by validating their internal desktop session. The desktop agent is also used to support web application SSO capability (protected by web agents) by redirecting the request to the web application.
- **Option Pack Agent:** This agent specifically enforces policies for federation application. It has the ability to generate and consume SAML assertion as well as WS Trust. The option pack agent communicates with the Federation Security Service (FSS) to manage federation partners.

#### Service Endpoints:

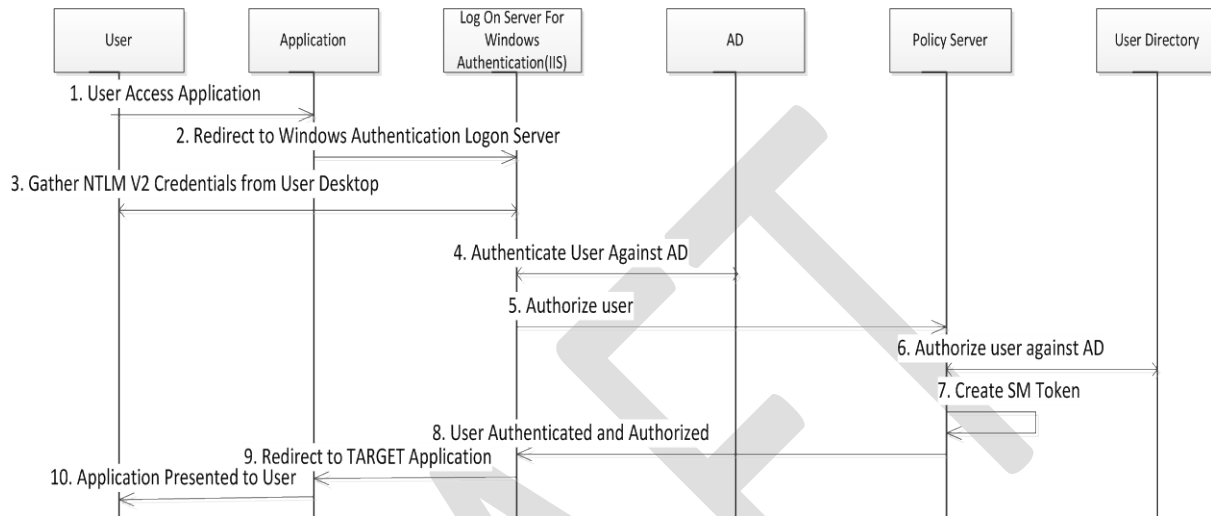
- **Web Service Security (WSS):** SSOi supports WS-Security tokens through WSS for various web service methods such as SOAP and REST. WSS also provides authentication and authorization web services to validate XML requests from client and generate sessions through XML response.
- **Federation Security Service (FSS):** FSS supports legacy through option pack agent and partnership federation through federation manager. It supports various federation standards such as SAML and WS-Federation. This provides the Identity Provider (IdP) and Service Provider (SP) objects for application integration.
- **Security Token Store Service (STS):** DataPower acts as the STS store that supports token translation requests from application end, where it supports WS-Trust token as input request having user's SiteMinder session as part of request. STS store validates Token request and will returns the standard user attributes as a part of response specification

#### Centralized Policy Engine:

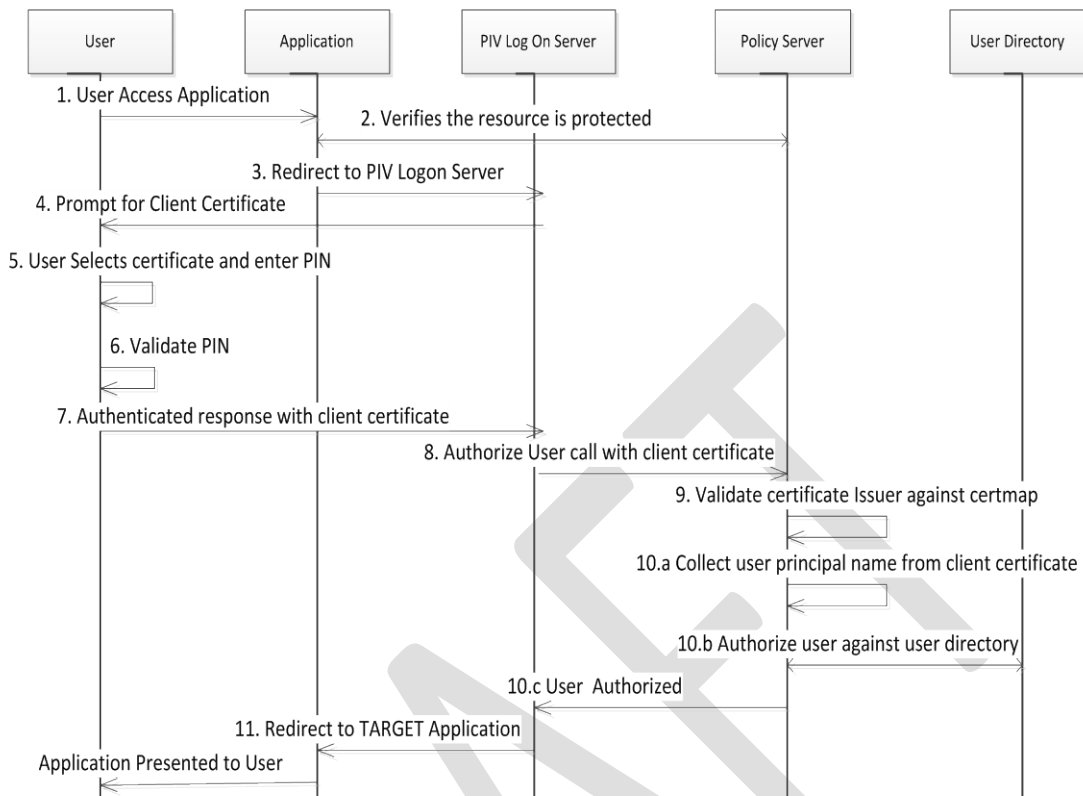
The SSOi policy engine is made up of CA SiteMinder and SSO policy server. All policy configuration, administration, and evaluation are managed through a centralized policy engine. The policy engine receives the requests from the different enforcement agents and service components. It then evaluates and takes action on the requests by providing an appropriate response back to the integrated application. The centralized policy engine provides various ways to authenticate a user such as user ID/password, Microsoft Windows authentication using Kerberos and NTLM token, PIV and PKI authentication, conversion of desktop token to a Web token, XML digital signature, SAML, and WS-Federation. SSOi validates credentials against the

back end user store and provides the SSO token as well as the user attributes to enforcement point for response back to the application.

Refer to Section 3.2.3.1 for screen interface.

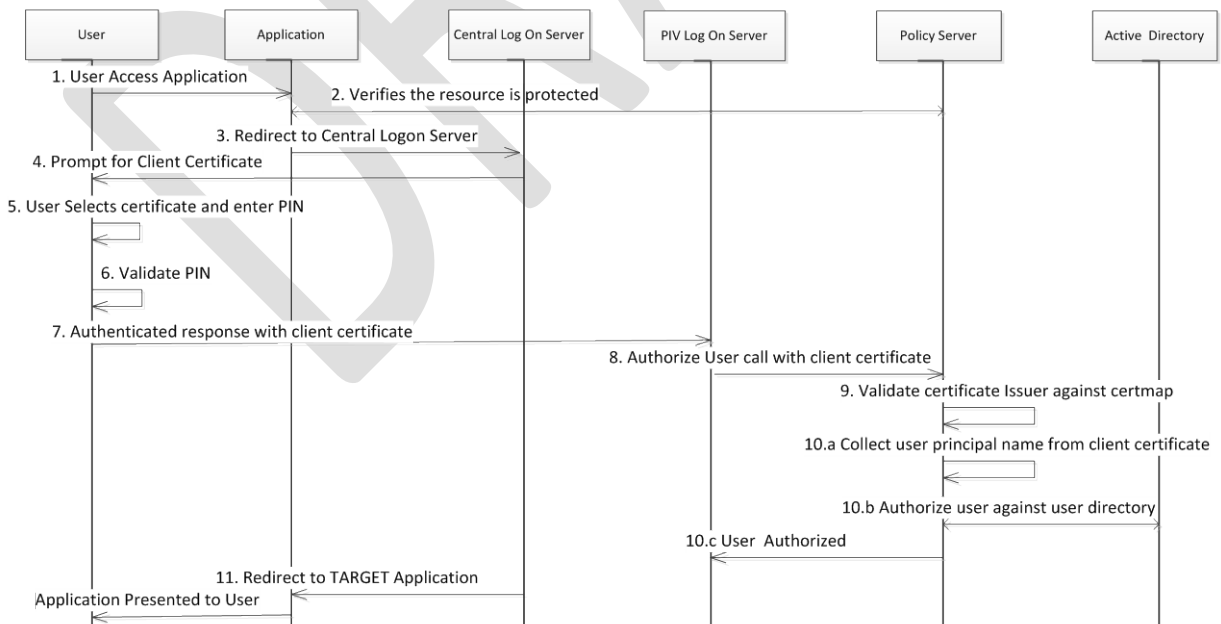


**Figure 20: SSOi Centralized Logon Page with Windows Authentication Sequence Diagram**



**Figure 21: Centralized Logon Page with PIV Authentication Sequence Diagram**

**Centralized PIV only Log on page with PIV authentication**



**Figure 22: Centralized PIV-Only Logon with PIV Authentication Sequence Diagram**

**Table 21: Windows Authentication, User ID/Password Authentication, and Centralized Logon Page with PIV Authentication**

Field	Description
Use Case Name	Authentication Support for Level of Assurance (LOA 2/3)
Description	This use case describes the process through which a user authenticates to the SSOi service using approved LOA 2/3
Actors	<ol style="list-style-type: none"> <li>1. Internal Users</li> <li>2. SSOi</li> <li>3. Centralized Logon Page</li> <li>4. SSOi Integrated Application(s)</li> <li>5. User Directory</li> </ol>
Pre-Conditions	<p>User has valid credential for each type of authentication method and tries to access the protected application.</p> <p>Invalid credentials are supported through error flows.</p>
Trigger	The internal user tries to access the application protected by SSOi.
Actions	<p><b>Centralized Logon Page with Windows Authentication</b></p> <ol style="list-style-type: none"> <li>1. SSOi Web Agent Intercepts the request to access integrated application and verifies it with policy server</li> <li>2. Web Agent redirects the request to centralized log on page with Windows authentication.</li> <li>3. The IIS Windows Authentication Logon Server</li> <li>4. The logon server collects the Kerberos credentials.</li> <li>5. SSOi authenticates the user against the Active Directory.</li> <li>6. The Logon Server passes the control to SiteMinder Policy server to authorize the user</li> <li>7. If the user is authorized to access the resource then a token is generated by SSOi (SiteMinder Policy Server)</li> <li>8. SSOi creates SiteMinder Token and then Notifies Windows Authentication Logon Server</li> <li>9. The Logon server redirects the user to application</li> <li>10. The Application is presented to the user.</li> </ol> <p><b>Centralized Logon Page with User ID/Password Authentication</b></p> <ol style="list-style-type: none"> <li>1. SSOi Web Agent Intercepts the user request to access integrated application</li> <li>2. The Web Agent verifies it with policy server if the application is protected</li> <li>3. If the resource is protected Web Agent redirects to Logon server</li> <li>4. Logon Server prompts for credentials</li> <li>5. The user enters the credentials</li> <li>6. The Logon Server passes the control to SiteMinder Policy server to authorize the user</li> <li>7. SiteMinder Policy Server authenticates and authorizes user against Active</li> </ol>



Field	Description
	<p>Directory</p> <ol style="list-style-type: none"> <li>SiteMinder Policy Server create SiteMinder Token</li> <li>User is authenticated and authorized to access the resource by SiteMinder Policy Server</li> <li>The Logon server redirects the user to application</li> </ol> <p><b>Centralized Logon Page with PIV Authentication</b></p> <ol style="list-style-type: none"> <li>SSOi Web Agent Intercepts the user request to access integrated application</li> <li>The Web Agent verifies it with Policy Server if the application is protected</li> <li>If the resource is protected, Web Agent redirects to centralized log on page where a user can select PIV Logon from the list of supported authentication methods.</li> <li>PIV Logon prompts to select Client certificate</li> <li>The user selects the client certificate and enters the PIN</li> <li>The SSL server maps the user's certificate to the server.</li> <li>CA SiteMinder verifies the user exists.</li> <li>CA SiteMinder verifies the user's basic credentials.</li> <li>CA SiteMinder verifies that the certificate credentials and the basic credentials represent the same user.</li> <li>If the user lookup failed on AD by SiteMinder then it generates user OnAuthattempt rule which redirects the user back to failed logon page or else it authorizes the access to the resource and redirects the user back to application with valid SiteMinder session cookie.</li> </ol> <p><b>Centralized Logon Page with PIV-Only Authentication (LOA 3)</b></p> <ol style="list-style-type: none"> <li>SSOi Web Agent Intercepts the user request to access integrated application</li> <li>The Web Agent verifies it with Policy Server if the application is protected by higher level 10 authentication.</li> <li>If the resource is protected, Web Agent redirects to centralized PIV log on page where a user can hit login button to login using PIV card.</li> <li>Central logon server sends the requests to PIV logon server.</li> <li>PIV Logon prompts to select Client certificate</li> <li>The user selects the client certificate and enters the PIN</li> <li>The SSL server maps the user's certificate to the server</li> <li>PIV certificate authentication happens at the TLS layer, higher authentication level (10) configured on policy server</li> <li>CA SiteMinder verifies the user exists.</li> <li>CA SiteMinder verifies the user's basic credentials.</li> <li>CA SiteMinder verifies that the certificate credentials and the basic credentials represent the same user.</li> <li>If the user is authorized to access the resource, the PIV Logon server redirects the user to application</li> </ol>
Main	User is authenticated successfully and application is presented to the user.

Field	Description
Success Scenarios	
Main Failure Scenarios	<p>No failed authorization may occur but system had been designed to handle to scenario if at all it may</p> <ul style="list-style-type: none"> <li>• Default failed Kerberos authentication will redirect the user to the centralized log Username/Password page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed Kerberos authorization will redirect the user to centralized login page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed UserId/Password authentication will redirect the user to the centralized log Username/Password page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed UserID/Password authorization will redirect the user to centralized login page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed PIV authentication will redirect the user to the centralized log Username/Password page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed PIV authorization will redirect the user to centralized login page but can be customized to redirect to any application page based on the policies.</li> <li>• Default Session Timeout redirects the users back to the centralize logon page</li> <li>• Default Authorization Failure to the application will redirect the user to the centralize failedlogin page</li> <li>• Default Application Logout Page will redirect the user back to the centralize logon page</li> </ul>

### 6.2.1.1. SSOi Support for LOA 2/3 External Users

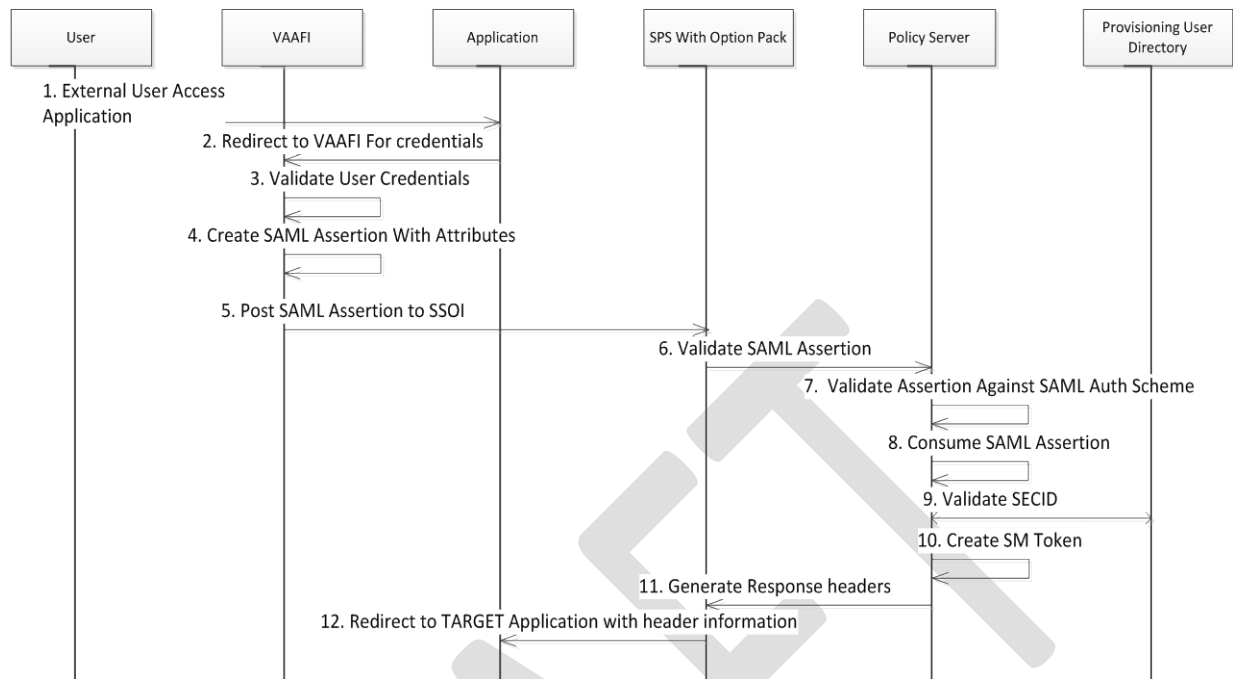


Figure 23: SSOi Support for LOA 2/3 External Users Sequence Diagram

Table 22: SSOi Support for LOA 2/3 External Users

Field	Description
Use Case Name	LOA 2/3-SSOi Integration Flow
Description	This use case describes the process by which an SSOi User performs SSO to one or more integrated application.
Actors	1. External User 2. VAAFI 3. SSOi Integrated Application(s) 4. User Directory
Pre-Conditions	1. The SECID which will be received from VAAFI and will be available as correlated attribute in Provisioning store 2. A valid external card user access the SSOi protected application through Access VA
Trigger	External User initiates the application session by clicking on the target application from Access VA
Actions	1. An external user accesses an application which is SSOi service provider via public URL. 2. The user will be redirected to IdP (VAAFI) for credentials 3. The IdP will authenticate the external user and generate the SAML assertion with user attributes as defined in integration RSD/SDD.

Field	Description
	<ol style="list-style-type: none"> <li>4. VAAFI SAML service posts the generate SAML Assertion to the SSOI SAML Assertion Consumer URL. SPS will proxy the internal URL access for external users.</li> <li>5. SSOI SAML Consumer server makes a call to the SiteMinder Policy server to validate SAML assertion.</li> <li>6. SM Policy server validates SAML assertion against SAML auth scheme and by verifying the digital signature. It consumes the assertion and after that decrypts the SAML Assertion.</li> <li>7. SM Policy server validates the user retrieved from SAML assertion against Provisioning User directory by validating SECID</li> <li>8. If the SECID is valid, Policy server creates the SM token and redirect to the target application with the required header variables such as Firstname, csid, icn, Lastname, EDIPI, email address, assurance level, and other attributes received as a part of assertion mentioned in the following table</li> <li>9. If the user is valid, Policy server creates the SM token and redirect to the target application with required header variables based on the policy configured for the integrated application and SPID</li> </ol>
Main Success Scenarios	User is authenticated and Application is presented to the user.
Main Failure Scenarios	<p>In the event of an exception or error during attribute consumption default SAML assertion error will be generated and returned it to VAAFI</p> <p>For each integration using SPS, the scenarios for Session timeout, logout and authentication/authorization failures will be implemented in a similar fashion as documented in <a href="#">6.2.4.10 Centralized Login page</a></p>

**Table 23: VAAFI IdP SAML Integration**

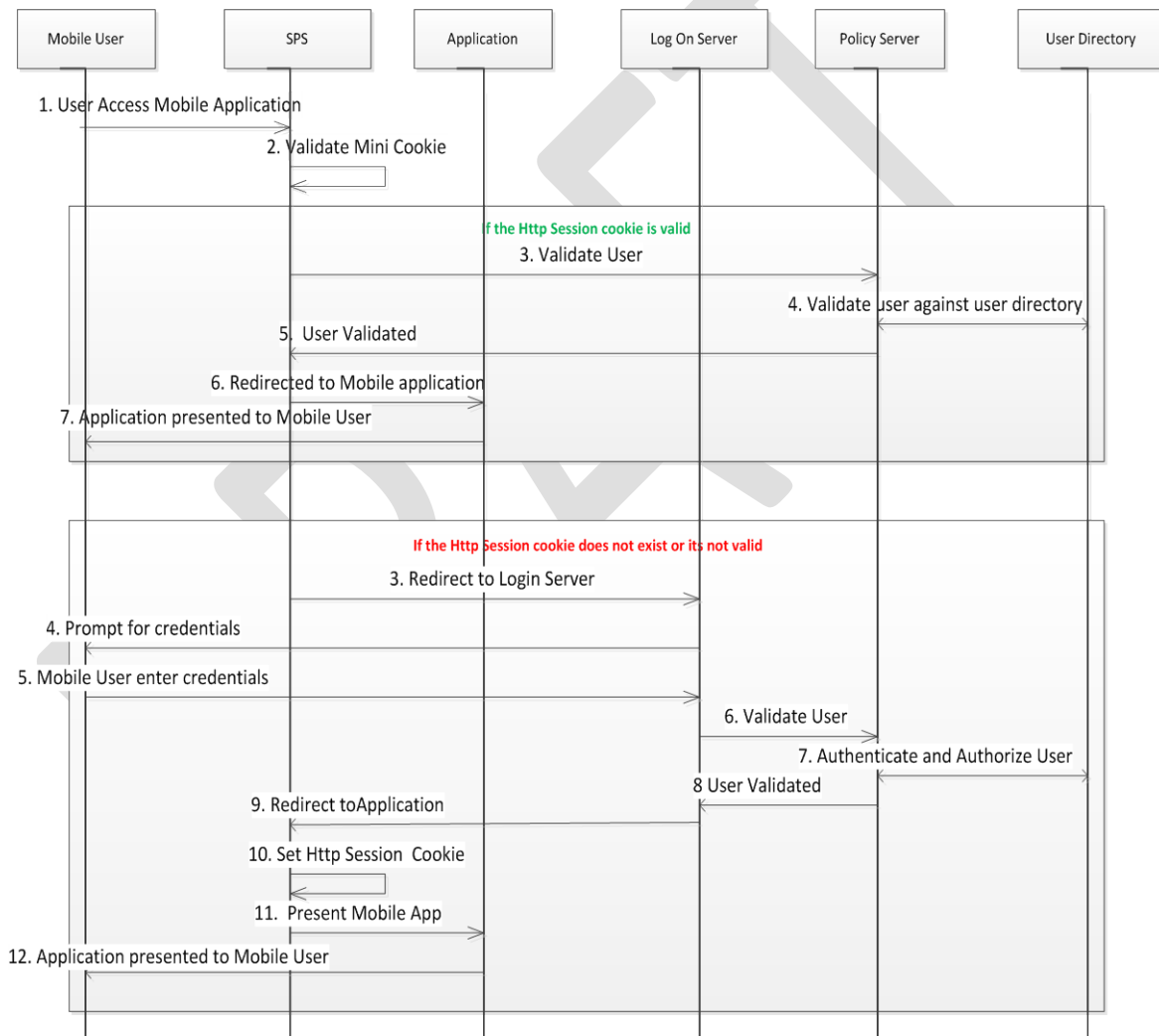
Field	Description
IDPID:	
SiteMinder Affiliate Domain	N/A
NameID	SubjectDN
AuthN Director	N/A
Encryption Algorithm	AES128
SLO	N/A
Attribute Details	va_eauth_secid , va_eauth_csid, va_eauth_birthdate, va_eauth_pnidtype, va_eauth_pnid, va_eauth_credentialassurancelevel, va_eauth_dodedipnid, va_eauth_birlsfilenumber, va_eauth_middlename, va_eauth_lastname, va_eauth_state, va_eauth_icn, va_eauth_suffix

Field	Description
Signature Algorithm	Signing Algorithm: RSA with SHA1 RSA Key Size: 256 SAML Token: Signed Attribute Encryption: AES-128

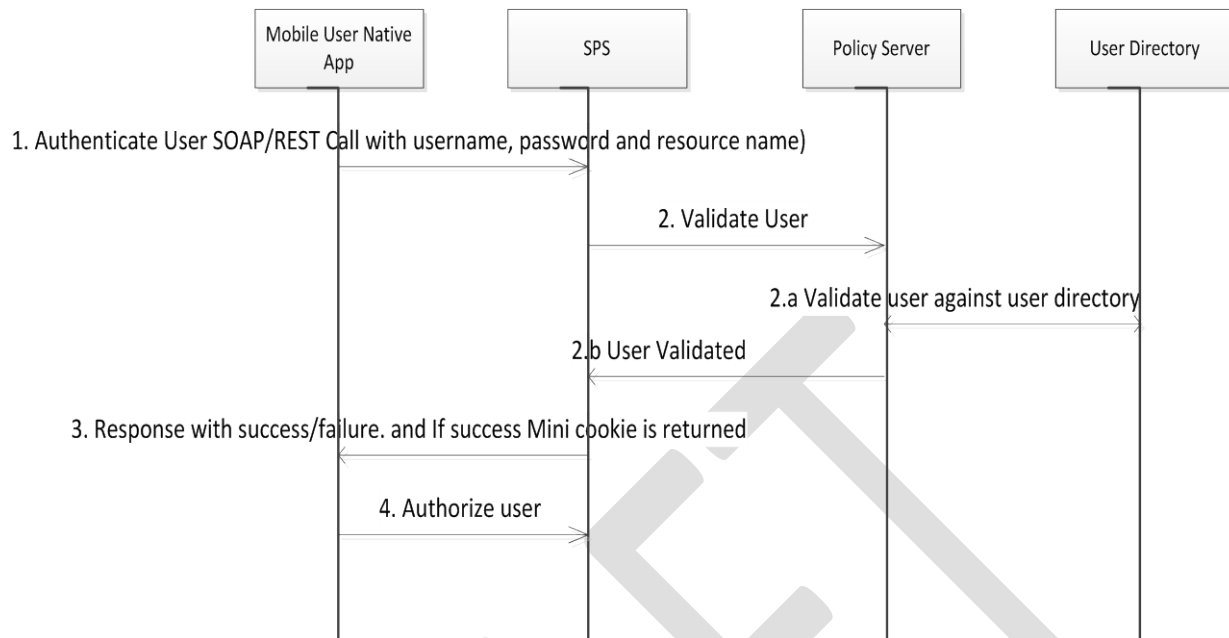
#### 6.2.1.2. SSOi Support for LOA 4 SAML Token/Holder of Key (HOK)

TBD. This section will be updated during the designated sprint cycle.

#### 6.2.1.3. SSOi Mobility Support



**Figure 24: SSOi Mobility Support Sequence Diagram**



**Figure 25: Access Mobile Application Using Native Apps Sequence Diagram**

**Table 24: Access Mobile Application Using Native Apps**

Field	Description
Use Case Name	SSOi Mobility Support
Description	This use case describes the process by which SSOi user performs authentication through mobile devices. Mini and SM Session cookies are supported.
Actors	1. Mobile User 2. SSOi Integrated Application(s) 3. User Directory
Pre-Conditions	The user has a mobile device with access to VA applications.
Trigger	Mobile User initiates authentication to application via a mobile device.
Actions	<b>Http Session Cookie is Valid</b> <ol style="list-style-type: none"> <li>1. Mobile User accesses the application URL through a mobile device</li> <li>2. CA Secure Proxy Server (SPS) intercepts the requests and check for the mini cookie availability</li> <li>3. If <b>http Session cookie</b> is valid then SPS will validate and update the session cookie with updated time stamp and pass the control back to the application</li> <li>4. After user entering the credentials, SPS validates the user by making a call to the Policy Server</li> <li>5. Policy Server validates the credentials by verifying it against user directory</li> <li>6. SiteMinder Policy Server validates the user</li> </ol>

Field	Description
	<p>7. SiteMinder Policy Server redirects to Mobile application</p> <p>8. Browser presents application to the Mobile user</p> <p><b>Http Session cookie does not exist or is not valid</b></p> <ol style="list-style-type: none"> <li>1. If <b>http Session Cookie</b> is not valid or does not exist it will redirect to the logon server</li> <li>2. Prompt for the Centralized Mobile authentication Page with UserID/Password option and PIV/PIN option (both authentication mechanisms follow the same process that is described in section <b>Error! Reference source not found.</b>)</li> <li>3. User enters credentials based on the option selected</li> <li>4. Policy Server validates the credentials</li> <li>5. Verify the credentials against user directory</li> <li>6. Receive valid user response</li> <li>7. After user validation completed, redirect to the application</li> <li>8. SPS creates and sets the http Session cookie</li> <li>9. Pass the control to the application.</li> <li>10. Present application to the Mobile User</li> </ol> <p><b>Access to mobile application using native apps</b></p> <ol style="list-style-type: none"> <li>1. A native app calls the authentication SOAP/REST based web service exposed by Secure Proxy server Authentication web service with respective input parameters such as username, password, and resource Uri. (Note - Currently SPS Webservice Solution does not support X509 tags due to limitation of the product)</li> <li>2. Authentication Web service validates the credentials</li> <li>3. Validate against policy server/user store.</li> <li>4. Receive validated user notification</li> <li>5. If successful then mini session cookie is returned as part of response code</li> <li>6. Application call Authorization service to get permit /deny response from Authorization web service else fail result code is returned as response</li> </ol>
Main Success Scenarios	User is authenticated successfully and application is presented to the user
Main Failure Scenarios	<ul style="list-style-type: none"> <li>• Default failed UserId/Password authentication will redirect the user to the centralized log Username/Password page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed UserID/Password authorization will redirect the user to centralized login page but can be customized to redirect to any application page based on the policies.</li> <li>• Default Session Timeout redirects the users back to the centralize Mobile logon page</li> <li>• Default Authorization Failure to the application will redirect the user to the</li> </ul>

Field	Description
	<p>centralize Mobile failedlogin page</p> <ul style="list-style-type: none"> <li>Default Application Logout Page will redirect the user back to the centralize Mobile logon page</li> <li>For Native apps, since it utilizes the SSOi web service, the error codes are mentioned in section <b>Error! Reference source not found.</b></li> </ul>

#### 6.2.1.4. Federation IdP and SP for Internal Users

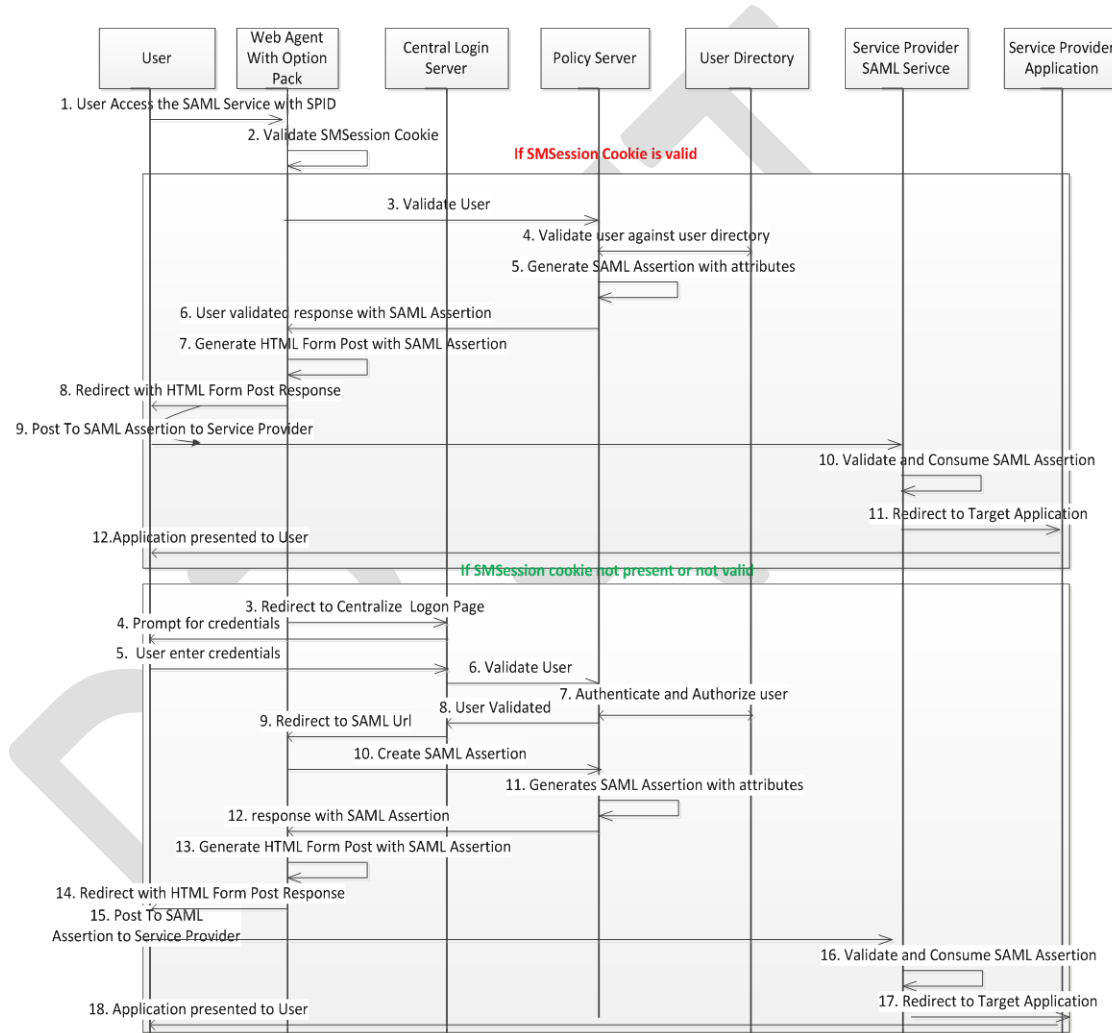
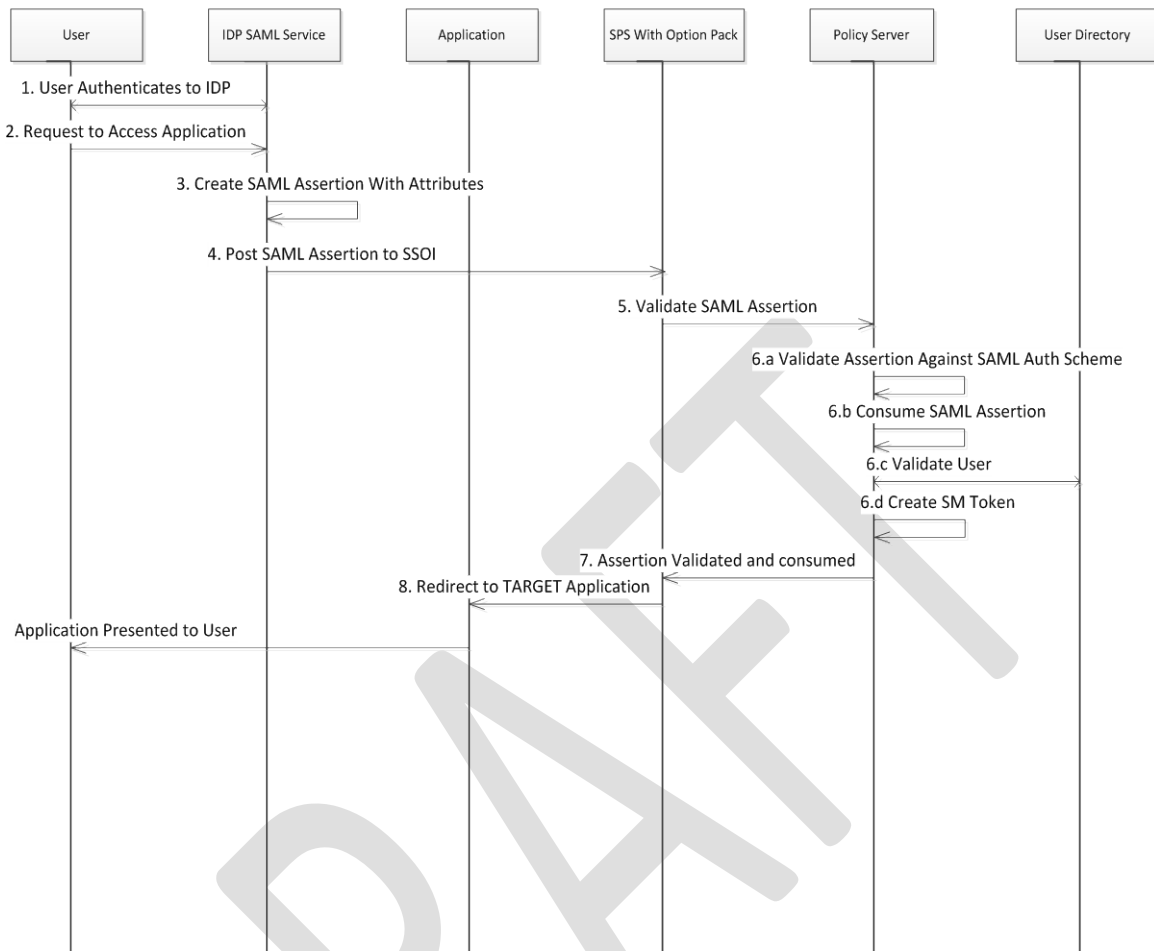


Figure 26: Federation IdP and SP for Internal Users





**Figure 27: Application Protected by Separate IdP (Other than SSOi) Sequence Diagram**

**Table 25: Application Protected by Separate IdP (Other than SSOi)**

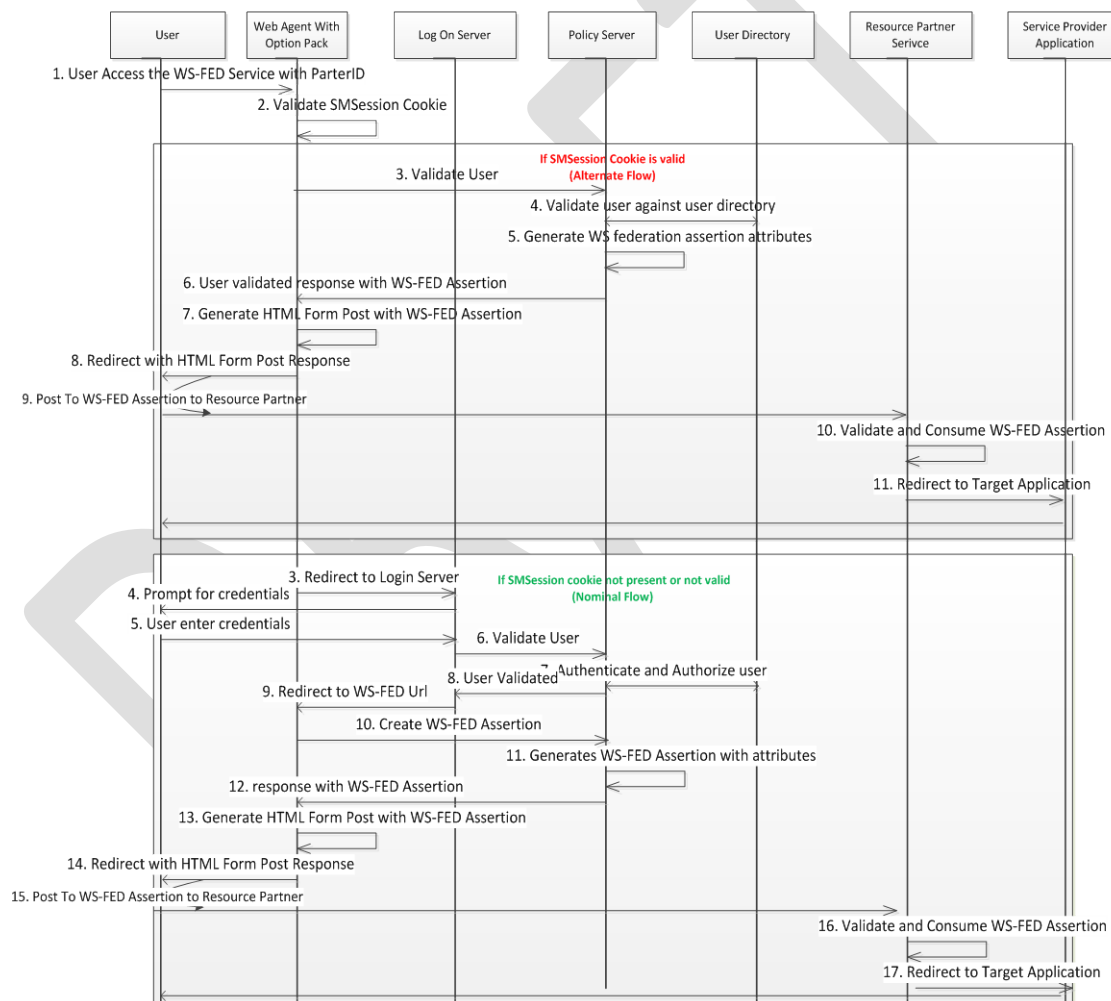
Field	Description
Use Case Name	Federation Identity Provider (IdP) and Service Provider (SP) services
Description	This use case describes the process by which a federated user is authenticated to the SSOi activity.
Actors	1. Internal Users 2. SSOi 3. SSOi Integrated Applications
Pre-Conditions	A valid integration/trust between Identity Provider (IdP) and Service Provider (SP)
Trigger	User Access application protected with SAML federation authentication mechanism
Actions	<b>Identity Provider (IdP) – Without a Valid Session Cookie</b>

Field	Description
	<ol style="list-style-type: none"> <li>1. An internal user accesses an application (IdP-protected URL) which is at service provider without a SiteMinder session cookie.</li> <li>2. Web server redirects user to centralize log on page and prompted for authentication credential</li> <li>3. User enters the credentials</li> <li>4. SiteMinder Policy server validates against User store.</li> <li>5. SiteMinder Policy server authenticates and authorizes the user</li> <li>6. SiteMinder Policy Server creates valid user token</li> <li>7. SiteMinder Policy server redirects to SAML URL</li> <li>8. SiteMinder Policy server generates the SAML Assertion by:</li> <li>9. SiteMinder Policy Server adds the required attributes such as user Principal Name (UPN), email, firstname and lastname</li> <li>10. IdP posts the SAML assertion to the Service Provider SAML Assertion Consumer service:</li> <li>11. SiteMinder Policy Server (IdP) generates HTML Form Post with SAML assertion</li> <li>12. SiteMinder Policy Server (IdP) redirects with HTML form post response</li> <li>13. SiteMinder Policy Server (IdP) posts SAML assertion to Service Provider</li> <li>14. Service provide Consumes the SAML assertion generated by SSOi and grants the access to the user</li> <li>15. Service provider redirect to protected SSOi application with valid SiteMinder session</li> <li>16. Service provider present application to the end user</li> </ol> <p><b>Identity Provider (IdP) – With a Valid Session Cookie</b></p> <ol style="list-style-type: none"> <li>1. An internal user accesses an application (IdP protected URL) which is at service provider with a SiteMinder Session cookie.</li> <li>2. The user is validated by SiteMinder Policy Server.</li> <li>3. SiteMinder Policy Server generates the SAML assertion by adding all the required attributes such as user Principal Name (UPN), email, firstname and lastname</li> <li>4. IdP posts the SAML assertion to the Service Provider SAML Assertion Consumer service:</li> <li>5. IdP generates HTML form post with SAML Assertion</li> <li>6. IdP redirects with HTML Form Post Response</li> <li>7. IdP posts to SAML Assertion to Service Provider</li> <li>8. Service provider consumes the SAML assertion generated by SSOi and grants the access to the user</li> <li>9. Service provider redirects to protected SSOi application with valid SiteMinder session</li> <li>10. Service provider presents application to the end user</li> </ol>

Field	Description
	<b>Service Provider (SP)</b> <ol style="list-style-type: none"> <li>1. An internal user accesses an application that is protected by a separate IdP other than SSOi.</li> <li>2. User enters the credentials and validated with IdP</li> <li>3. The SAML assertion is generated by adding the required attributes such as user Principal Name (UPN), email, firstname, and lastname by IdP</li> <li>4. IdP posts the SAML assertion to the Service Provider which is configured at SiteMinder</li> <li>5. SPS / Webagent option pack validates the SAML Assertion with the Policy Server</li> <li>6. Service provider consumes the SAML assertion generated by IdP</li> <li>7. Service provider validates the user attributes.</li> <li>8. Service provider creates SiteMinder Token</li> <li>9. SAML Assertion is consumed by Service provider</li> <li>10. User is redirected protected SSOi application with valid SiteMinder session by Service provider</li> </ol>
Main Success Scenarios	User is authenticated and Application is presented to the user.
Main Failure Scenarios	<p>VA as IdP:</p> <ul style="list-style-type: none"> <li>• Default failed Kerberos authentication will redirect the user to the centralized log Username/Password page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed Kerberos authorization will redirect the user to centralized login page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed UserID/Password authentication will redirect the user to the centralized log Username/Password page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed UserID/Password authorization will redirect the user to centralized login page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed PIV authentication will redirect the user to the centralized log Username/Password page but can be customized to redirect to any application page based on the policies.</li> <li>• Default failed PIV authorization will redirect the user to centralized login page but can be customized to redirect to any application page based on the policies.</li> <li>• Default Session Timeout redirects the users back to the centralize logon page</li> <li>• Default Authorization Failure to the application will redirect the user to the centralize failedlogin page</li> <li>• Default Logout will redirect the user back to the centralize logon page</li> </ul>

Field	Description
	<p>VA as SP:</p> <ul style="list-style-type: none"> <li>SSOi consumes the assertion and generates the SMsession to provide the access to the application. The application policy will drive the failure conditions.</li> <li>Default Session Timeout redirects the users back to the centralize logon page</li> <li>Default Authorization Failure to the application will redirect the user to the centralize failed login page</li> <li>Default Logout will redirect the user back to the centralize logon page</li> </ul>

### 6.2.1.5. WS Federation for Internal Users



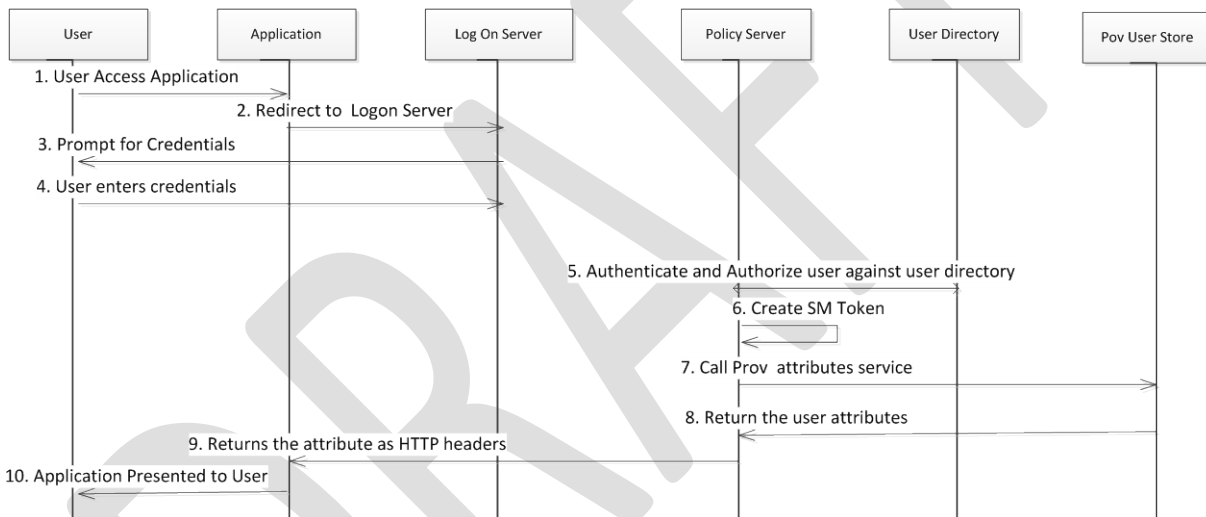
**Figure 28: WS Federation for Internal Users Sequence Diagram**

**Table 26: WS Federation for Internal Users**

Field	Description
Use Case Name	WS Federation for Internal Users
Description	This use case describes the process by which a user gets seamless access to the relying partner application using WS trust.
Actors	<ol style="list-style-type: none"> <li>1. Internal Users</li> <li>2. SSOi</li> <li>3. SSOi Integrated Application(s)</li> <li>4. User Directory</li> </ol>
Pre-Conditions	A valid WS integration/ trust between Identity provider and relying partner
Trigger	User access application protected with WS federation authentication
Actions	<ol style="list-style-type: none"> <li>1. User accesses the application without a valid SiteMinder session</li> <li>2. Web server redirects to Login Server</li> <li>3. User prompted for credentials</li> <li>4. User enters the credentials</li> <li>5. Validated against SiteMinder Policy server</li> <li>6. Authenticate and authorize user against User store</li> <li>7. Notify Log On Server of validated user</li> <li>8. Generate the WS Federation Assertion token:</li> <li>9. Redirect to WS Federation URL</li> <li>10. Create WS Federation Assertion</li> <li>11. Generate WS Federation assertion with attributes such as User Principal Name (UPN), email, firstname, and lastname</li> <li>12. Notify Web Agent of transaction</li> <li>13. SiteMinder posts the WS Federation assertion to the Relying party configured</li> <li>14. Redirect with HTML Form Post</li> <li>15. Relying party validates the WS federation token generated</li> <li>16. Relying party consumes WS- Federation</li> <li>17. Redirect to Target Application</li> </ol> <p><b>Alternate Flow</b></p> <ol style="list-style-type: none"> <li>1. User accesses the application with a valid SiteMinder session</li> <li>2. Validate user credentials:</li> <li>3. Validated against SiteMinder Policy server User store.</li> <li>4. Authenticate and authorize user against User store</li> <li>5. SiteMinder Policy server generates the WS Federation Assertion token by adding all the required attributes such as user Principal Name (UPN), email, firstname, lastname.</li> <li>6. SiteMinder posts the WS Federation assertion to the Relying party</li> </ol>

Field	Description
	<p>configured.</p> <p>7. Redirect with HTML Form Post</p> <p>8. Relying party validates the WS federation token generated</p> <p>9. Relying party consumes WS-Federation</p> <p>10. Redirect to Target Application</p>
Main Success Scenarios	User is authenticated and Application is presented to the user.
Main Failure Scenarios	Assertion failure errors generated by Relying party unable to consume WS-Federation assertions.

#### 6.2.1.6. SSOi Support for Attribute Service



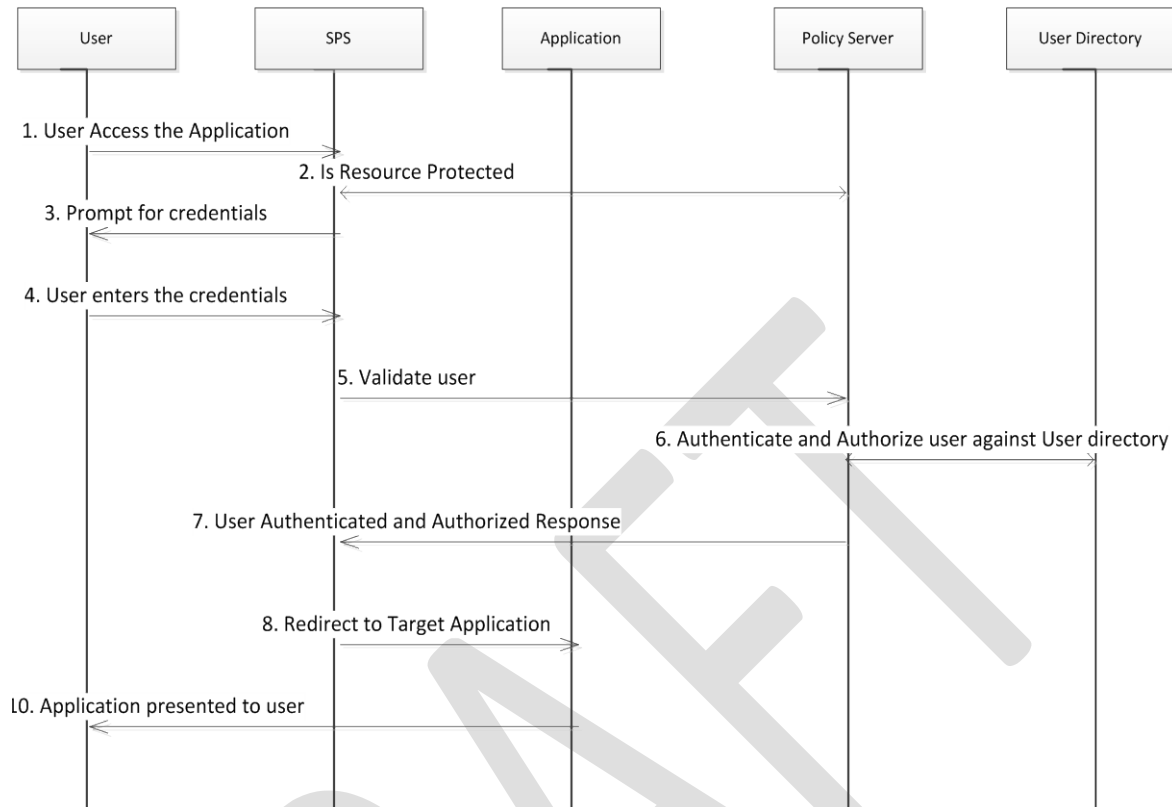
**Figure 29: SSOi Support for Attribute Service Sequence Diagram**

**Table 27: SSOi Support for Attribute Service**

Field	Description
Use Case Name	SSOi Support for Attribute Service
Description	This use case describes the process by which SiteMinder calls the attribute service from VDS during authorize policy evaluation.
Actors	<p>1. Users</p> <p>2. SSOi</p> <p>3. SSOi Integrated Application(s)</p> <p>4. User Directory</p> <p>5. Prov User Store</p>
Pre-	A valid WS integration/ trust between Identity and Relying partner

Field	Description
Conditions	
Constraints	SSOi will be depend the capability of VDS attribute service capability to get appropriate attributes
Trigger	User authenticated with SSOi and needs specific attributes from VDS
Actions	<ol style="list-style-type: none"> <li>1. User authenticates in to SSOi</li> <li>2. SSOi redirects to Logon Server</li> <li>3. Logon server prompts for Credentials</li> <li>4. User enters credentials</li> <li>5. SSOi authenticates and authorize user against user store</li> <li>6. SiteMinder session token is generated by SiteMinder Policy Server</li> <li>7. During evaluation of authorization policies SiteMinder policy server call the Prov user store with input as userid</li> <li>8. Attribute service returns the attribute set to SiteMinder policy server at the run time.</li> <li>9. SiteMinder set them on http headers as response and provide it back to the application.</li> </ol>
Main Success Scenarios	User is authenticated and Application is presented to the user.
Main Failure Scenarios	Failure to receive attribute will result in blank response, which will be handled by application to display application specific error codes.

### 6.2.1.7. SSOi Proxy Authentication Request



**Figure 30: SSOi Proxy Authentication Request Sequence Diagram**

**Table 28: SSOi Proxy Authentication Request**

Field	Description
Use Case Name	SSOi Proxy Authentication Request
Description	This use case describes the process by exchanges that SiteMinder offers proxy capability for the authentication request. The centralized login page will be integrated with SPS for implementing SSOI using multiple authentication methods (LOA2, LOA3,).
Actors	1. Users 2. SSOi 3. SSOi Integrated Application(s)
Pre-Conditions	All application access requests go through SPS
Trigger	User accesses application protected and proxy through SPS
Actions	1. The user access to the application which proxy through Secure proxy server 2. The Secure proxy Server verifies the policy server to check the resource is protected 3. If the resource is protected SPS, it prompts the user for credentials 4. User submits credentials



Field	Description
	5. Secure proxy server validates the credentials with policy server 6. SiteMinder Policy Server authenticates and authorizes user against User Store 7. SiteMinder Policy server sets the cookie and passes control back to SPS 8. Secure Proxy Server invokes proxy engine and passes control to the application
Main Success Scenarios	User is authenticated and Application is presented to the user
Main Failure Scenarios	<ul style="list-style-type: none"> <li>• Default Session Timeout redirects the users back to the logon handler</li> <li>• Default Authorization Failure to the application will redirect the user to the failedlogon handler</li> <li>• Default Application Logout Page will redirect the user back to the logon handler</li> </ul>

#### 6.2.1.8. Session Management

TBD. This section will be updated during the designated sprint cycle.

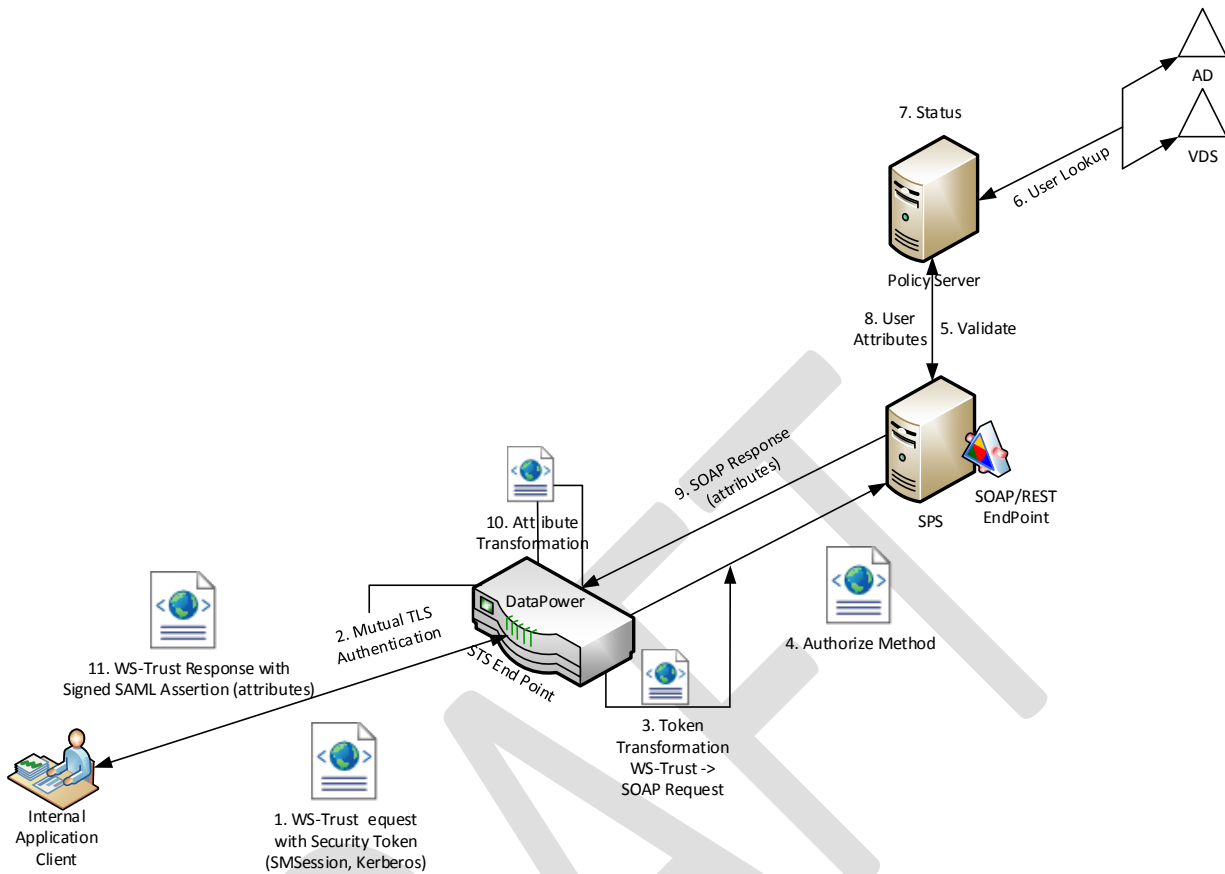
#### 6.2.1.9. SSOi STS Architecture Flow

The IAM STS WS is the service that builds, signs, and issues security tokens according to the Web Service (WS-\*) collection of standards, to include WS-Trust, WS-Policy, and WS-Security.

The IAM STS WS acts as an authorization broker between consumer (client or client application) and a producer. In lieu of generic or service account credentials, the consuming application utilizes security tokens or active SSO sessions, to request additional security tokens (or claims) from the IAM STS WS. The consuming application may then use the security tokens to authorize with producing applications. The producing application uses these tokens to make authorization decisions, and return appropriate business data to the consuming application.

The service will be multi-protocol and multi-token, allowing the greatest amount of flexibility in the requesting and issuance of tokens.

The SSOi STS architecture is depicted in the following diagram. Details are pending design completion in Sprint 5.

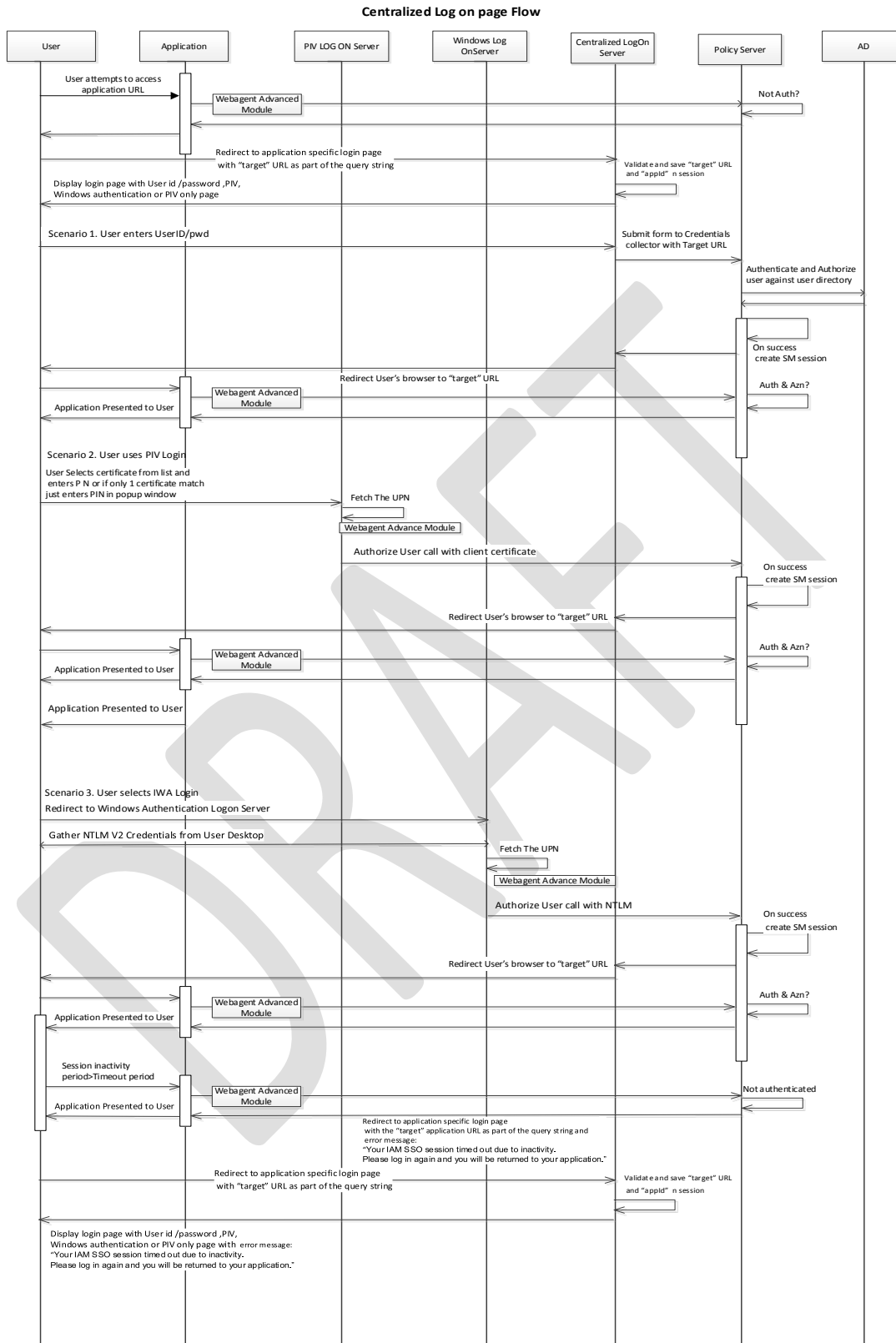


**Figure 31: SSOi STS Architecture Diagram**

Figure 31 is the high-level STS relationship diagram depicting a consumer and producer using an STS token service – To be updated during Sprint.

#### 6.2.1.10. Centralized Login Page

To support accessing VA applications with multiple authentication mechanisms at one place, the SSOi activity provides a static centralized logon page to support userID / Password, PIV, or Microsoft Windows authentication. This page is modifiable for each application to reflect only the authentication mechanisms selected by the integrating VA application. Also, to support VA applications with PIV compliance, the SSOi activity provides a static PIV only centralized logon page to support only using PIV card. A pre-condition to this is client has certificate to support mutual TLS authentication. (Refer to Section 3.2.3.1 for screen mock-ups.)

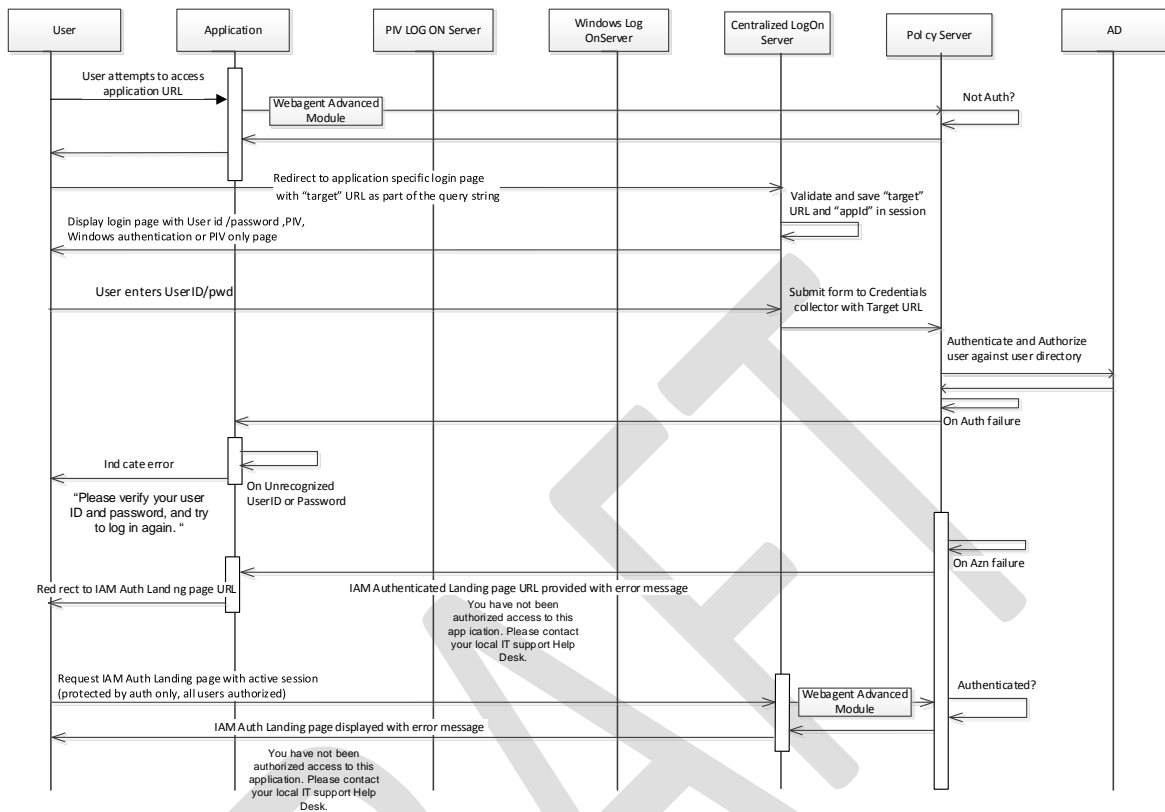


**Figure 32: Centralized Logon Page Flow**

The centralized logon page flow includes the following steps:

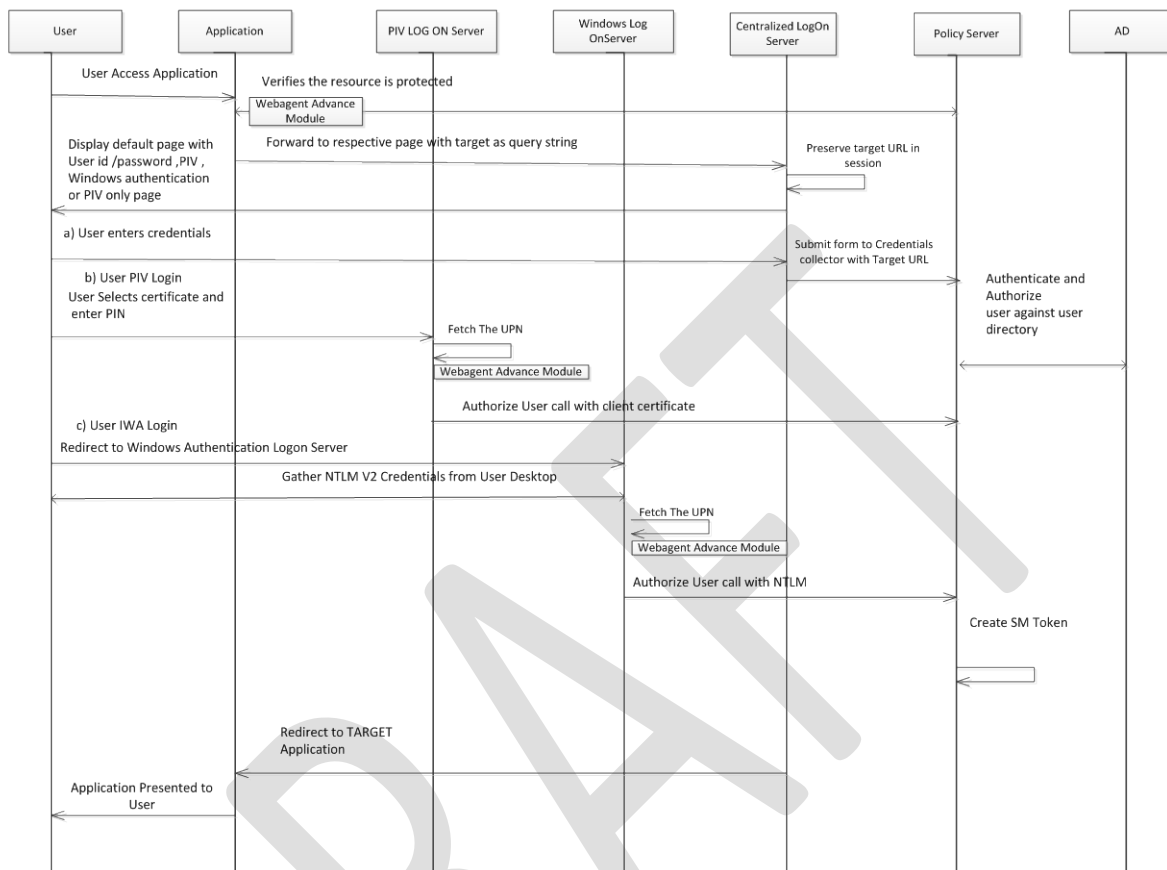
1. User attempts to access VA application protected by SiteMinder multiple authentication
2. Web Agent Intercepts the request to access integrated application and verifies it with policy server
3. SiteMinder redirects the request to respective (PIV or Default) static centralized log on page with application name and target URL of the application as query string.
4. Central login page handler preserves the target and displays static central login page to user.
5. For multiple authentication supported applications, central login page handler provides user with an option to choose either user ID, password, PIV card, or windows authentication method to log into application.
6. For PIV only and PIV compliance supported applications, PIV only central login page handler displays PIV only login page.
7. If user submits user ID and password, the request is sent by the browser to central login handler which submits the credentials to login FCC SiteMinder credential collector.
8. If user login with Windows authentication, the request is sent by the browser to windows NTLM logon sever which checks with policy server for authentication.
9. If user login with PIV card and for PIV only login, the request is sent by the browser to PIV logon sever which checks with policy server for authentication
10. Policy server authenticates the user against the Active Directory.
11. If the user is authorized by Policy Server to access the resource then a token is generated
12. SSOi creates SiteMinder Token and redirects the user to application

The following figure represents the error handling capabilities behind the authentication and authorization processes, implemented within the Centralized Login page.

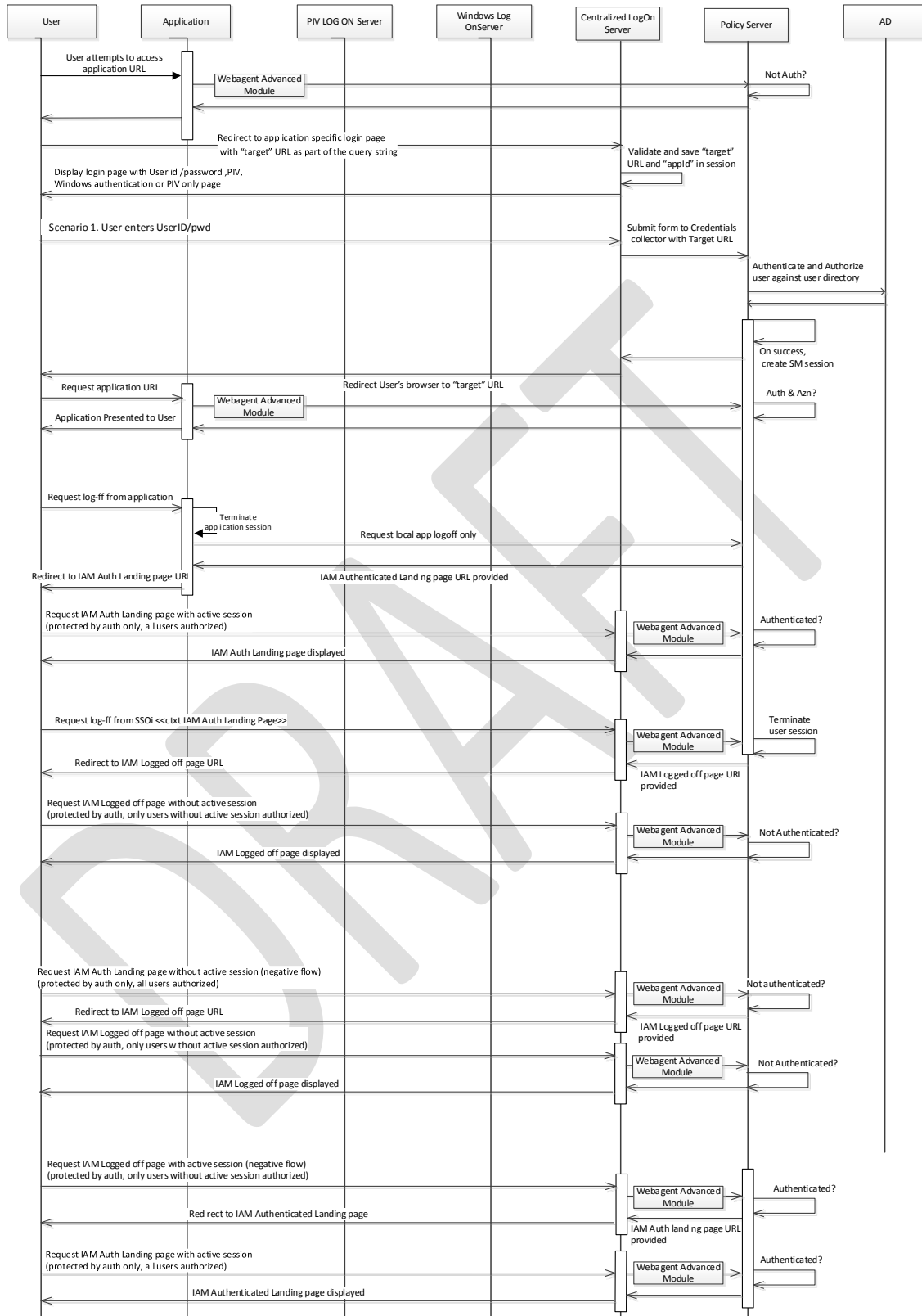


**Figure 33: Centralized Logon Page Error Handling Flow**

The following figure represents the supported partial and complete logoff capabilities implemented within the Centralized Login page.

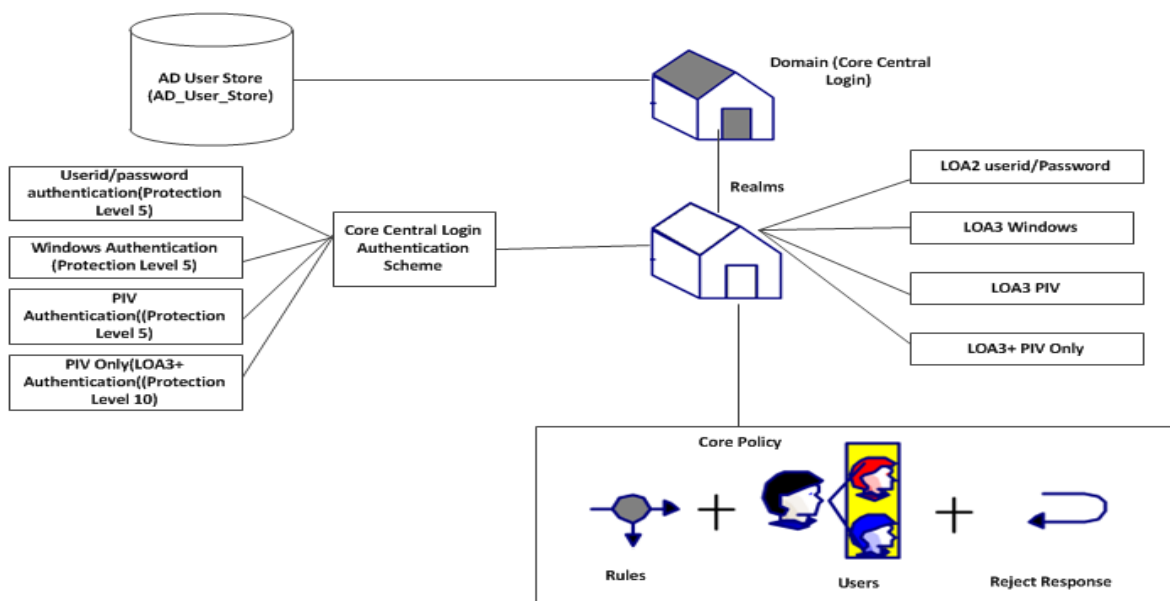


**Figure 34: Centralized Login Page Supported Partial and Complete Logoff Capabilities**



**Figure 35: Centralized Logon Page – Logoff Flows**

The following diagram depicts the SiteMinder policy architecture for core centralized authentication flows that was described above



**Figure 36: SiteMinder Policy Architecture for Core Centralized Authentication Flows**

#### 6.2.1.11. OAuth

OAuth 2.0 (OAuth) is an authorization framework that provides a mechanism for clients to access resources on behalf of a resource owner. It also enables a resource owner to, explicitly or implicitly, provide consent (permit/deny) for accessing resources, also known as a three-legged flow. By introducing an authorization layer, OAuth alleviates the need for resource owners to share their credentials with the clients. OAuth defines four roles as part of the protocol flow:

- Resource Owner: Entity capable of providing access to a protected resource. The resource owner can be system or a user.
- Resource Server: Entity responsible for protecting resources and authorizing access to them. Resource Servers play the role of the Policy Enforcement Point.
- Client: An application accessing resources on behalf of the resource owner. The OAuth 2.0 specification defines two types of clients:
  - Client: Clients incapable of maintaining the confidentiality of their credentials (e.g., clients running on devices); and
  - Confidential Client: Clients capable of maintaining the confidentiality of their credentials (e.g., server side web application). These clients are suitable for the Authorization Code grant.
- Authorization Server: Entity responsible for issuing authorization codes and access tokens.

OAuth defines a number of methodologies for clients to access resources on behalf of the resource owners.

Refer to the SSOe SDD for additional information.



### 6.2.1.12. Product Perspective

Refer to section 3.1.3 for information on COTS products for the SSOi.

#### 6.2.1.12.1. User Interfaces

Refer to section 3.2.3 for information on user interfaces.

#### 6.2.1.12.2. Hardware Interfaces

Refer to section 6.1 for information on hardware configurations and interfaces.

#### 6.2.1.12.3. Software Interfaces

Refer to section 4.2 for software architecture design for the SSOi.

#### 6.2.1.12.4. Communications Interfaces

The following table displays the necessary port communications and protocols used for each component-based server. The ports described must be open for both inbound and outbound communications. The ports mentioned below indicate inbound ports and are opened to AcS components for communication.

**Table 29: Port Communications and Protocols**

Application	Network	Port(s)	Reason	Protocol(s)
Oracle Database	Internal		Oracle SQL Net Listener	TCP (JDBC)
Oracle Database	Internal		DataGuard	TCP
Oracle Database	Internal		Connection Manager	TCP
Oracle Database	Internal		Oracle Management Agent	TCP
Oracle Database	Internal		Oracle Enterprise Database Console (HTTP Port)	HTTP
Oracle Database	Internal		Oracle Enterprise Database Console (RMI Port)	TCP
Oracle Database	Internal		Oracle Enterprise Database Console (JMS Port)	TCP
Oracle Database	Internal		Agent command and control listening port	TCP
Oracle Database	Internal		CA UARM collection server	TCP
Oracle Database	Internal		SAILPT – Role manager internal database	TCP
CA Directory (CSP,IP, Provisioning)	Internal		Provisioning router dsa	TCP
CA Directory (CSP,IP, Provisioning)	Internal		Provisioning main dsa	TCP

Application	Network	Port(s)	Reason	Protocol(s)
CA Directory (CSP,IP, Provisioning)	Internal		Provisioning common objects dsa	TCP
CA Directory (CSP,IP, Provisioning)	Internal		Provisioning inclusions dsa	TCP
CA Directory (CSP,IP, Provisioning)	Internal		Provisioning notify dsa	TCP
CA Directory (CSP,IP, Provisioning)	Internal		DXWebserver Listener (SSL)	HTTPS
CA Directory (CSP,IP, Provisioning)	Internal		DXWebserver Listener for shutdown command	TCP
CA Directory (CSP,IP, Provisioning)	Internal		Dxmanager-DXadmin communication	TCP
CA Directory (CSP,IP, Provisioning)	Internal		CSP/PROV/SMPS Router DSA	LDAPS
CA Directory (CSP,IP, Provisioning)	Internal		CSP Data DSA	LDAPS
CA Directory (CSP,IP, Provisioning)	Internal		SMPS Data DSA	LDAPS
CA Directory (CSP,IP, Provisioning)	Internal		PROV Data DSA	LDAPS
CA Directory (CSP,IP, Provisioning)	Internal		DXadmin Secure LDAP	TCP
CA Directory (CSP,IP, Provisioning)	Internal		Agent command and control listening port	TCP
CA Directory (CSP,IP, Provisioning)	Internal		CA UARM Collection Server	TCP

Application	Network	Port(s)	Reason	Protocol(s)
Web Tier IIS Servers (CSP WebUI, IP WebUI, Provisioning WebUI)	DMZ		Accounting port	TCP
Web Tier IIS Servers (CSP WebUI, IP WebUI, Provisioning WebUI)	DMZ		Authentication port	TCP
Web Tier IIS Servers (CSP WebUI, IP WebUI, Provisioning WebUI)	DMZ		Authorization port	TCP
Web Tier IIS Servers (CSP WebUI, IP WebUI, Provisioning WebUI)	DMZ		Auditing Port	TCP
Web Tier IIS Servers (CSP WebUI, IP WebUI, Provisioning WebUI)	DMZ		SSL port for reverse proxy	HTTPS
Web Tier IIS Servers (CSP WebUI, IP WebUI, Provisioning WebUI)	DMZ		Agent command and control listening port	TCP
Web Tier IIS Servers (CSP WebUI, IP WebUI, Provisioning WebUI)	DMZ		CA UARM Collection Server	TCP
CA Report Server	Internal		WebLogic Port for Report Server	HTTPS

Application	Network	Port(s)	Reason	Protocol(s)
CA Report Server	Internal		Central Management Console Server Port	TCP
CA Report Server	Internal		Agent command and control listening port	TCP
CA Report Server	Internal		CA UARM Collection Server	TCP
Federation Option Pack	DMZ		ServletExec port for listening incoming requests from IIS	TCP
Federation Option Pack	DMZ		Agent command and control listening port	TCP
Federation Option Pack	DMZ		CA UARM Collection Server	TCP
CA Identity Manager (CSP,IP, Provisioning)	Internal		Administration Port	HTTPS
CA Identity Manager (CSP,IP, Provisioning)	Internal		Manage Server Port	TCP
CA Identity Manager (CSP,IP, Provisioning)	Internal		Node Manager	TCP
CA Identity Manager (CSP,IP, Provisioning)	Internal		Agent command and control listening port	TCP
CA Identity Manager (CSP,IP, Provisioning)	Internal		CA UARM Collection Server	TCP
Provisioning Server	Internal		Provisioning Server	TCP
Provisioning Server	Internal		Agent command and control listening port	TCP
Provisioning Server	Internal		CA UARM Collection Server	TCP
CA SiteMinder	Internal		Accounting port	TCP
CA SiteMinder	Internal		Authentication port	TCP

Application	Network	Port(s)	Reason	Protocol(s)
CA SiteMinder	Internal		Authorization port	TCP
CA SiteMinder	Internal		Auditing Port	TCP
CA SiteMinder	Internal		SSL port for reverse proxy	HTTPS
CA SiteMinder	Internal		WebLogic port for SiteMinder Admin UI	TCP
CA SiteMinder	Internal		Agent command and control listening port	TCP
CA SiteMinder	Internal		CA UARM Collection Server	TCP
CA SiteMinder SPS	DMZ/Internal		Apache HTTP Port	HTTP
CA SiteMinder SPS	DMZ/Internal		Apache SSL port	HTTPS
CA SiteMinder SPS	DMZ/Internal		Tomcat/ SPS HTTP Port	HTTP
CA SiteMinder SPS	DMZ/Internal		Tomcat/SPS SSL Port	HTTPS
CA UARM	Internal		Administration Port for CA UARM	TCP
CA UARM	Internal		SSL Port (reverse proxy to administration port 5250) for CA UARM	HTTPS
CA UARM	Internal		Syslog port (UDP) for CA UARM server	TCP
CA UARM	Internal		Syslog TCP listening port for CA UARM	TCP
CA UARM	Internal		Agent command and control listening port	TCP
CA UARM	Internal		Communication port for ODBC /JDBC driver	TCP
CA UARM	Internal		Audit client communication with port-mapper	TCP
CA UARM	Internal		Dispatcher SME listener	TCP
CA UARM	Internal		CA Directory LDAP DXadmin port (CA Directory bundled with CA UARM)	TCP
CA UARM	Internal		Dispatcher Service in SSL mode for events from Client Connector	TCP

Application	Network	Port(s)	Reason	Protocol(s)
CA SSO Server	Internal		Port for ticket granting agent (Windows Authentication Agent)	TCP
CA SSO Server	Internal		Access Control port bundled with CA SSO	TCP
CA SSO Server	Internal		LDAP communication port for CA Directory bundled with CA SSO for user directory	LDAPS
CA SSO Server	Internal		LDAP communication port for CA Directory bundled with CA SSO for token directory	LDAPS
CA SSO Server	Internal		TCP SSL port where the SSO Server will listen.	TCP
CA SSO Server	Internal		Agent command and control listening port	TCP
CA SSO Server	Internal		CA UARM Collection Server	TCP
DataPower XI52	Internal		Administration port	TCP
DataPower XI52	Internal		Web services	HTTPS
ARX CoSign	Internal		API Calls	HTTPS
eSig WebLogic	Internal		Administration Port	HTTPS
eSig WebLogic	Internal		Manage Server Port	TCP
eSig WebLogic	Internal		Node Manager	TCP
Radiant Logic VDS	Internal		LDAP SSL port	LDAPS
Radiant Logic VDS	Internal		Application server Admin Port	HTTP
Radiant Logic VDS	Internal		Application server HTTP Port	HTTP
Radiant Logic VDS	Internal		Application server HTTPS port	HTTPS
Radiant Logic VDS	Internal		Application server JMX port	TCP
Radiant Logic VDS	Internal		Control Panel Web server port	HTTP
Radiant Logic VDS	Internal		Control Panel Web server Port	HTTPS
Radiant Logic VDS	Internal		Web Services Port	HTTPS

Application	Network	Port(s)	Reason	Protocol(s)
Axiomatics	Internal	████	HTTP Connector Port	HTTP
Axiomatics	Internal	████	AJP Connector Port	TCP
Axiomatics	Internal	████	Server Shutdown Port	TCP
Axiomatics	Internal	████	HTTPS Connector Port	HTTPS
SailPoint	Internal	████████	HTTPS Connector Port	HTTPS

#### 6.2.1.12.5. Memory Constraints

This section is not applicable to the SSOi.

#### 6.2.1.12.6. Special Operations

This section is not applicable to the SSOi.

#### 6.2.1.13. Product Features

The SSOi is based on the foundation of CA COTS products. The table below describes the SSOi products. The software applicable to SSOi are items 1 and 2.

**Table 30: SSOi Products**

#	Software	Description
1	CA SiteMinder Web Access Manager	SiteMinder Web Access Manager is a web access management system that enables user authentication and secure Internet SSO (single sign-on), policy-driven authorization, federation of identities, and auditing of access to the web applications it protects.
2	CA Directory	<p>CA Directory provides directory services and security for online applications for organizations. For example, it enables customers to access their electronic accounts; employees can access critical business data.</p> <p>This product is generally considered a highly scalable and distributable implementation of directory services, including security services (e.g., authentication).</p> <p>CA Directory is supported on a variety of Windows and UNIX platforms, as well as 64-bit operating systems such as Linux 64, Solaris 10/Intel 64, UltraSparc 64, IBM Power5 64 and HPUX Itanium 64.</p> <p>CA Directory supports open standards including: LDAP (and related RFCs), X.500 (DAP, DSP, DISP), Security (SSL, TLS, password hashes), Management (SNMP and related RFCs), Network (IPv6, RFC1006), and US Federal Government standards (FIPS 140-2, Common Criteria EAL3, and Section 508).</p>

#	Software	Description
3	WebLogic	BEA WebLogic Portal is now known as WebLogic Portal. WebLogic Portal is a well-known, widely used, Java-based portal product and a portal framework. The WebLogic Portal product is out-of-the-box software that aggregates information, content, applications, business processes, and knowledge assets into a personalized display. The WebLogic Portal framework is the portal product in kit form, providing a set of tools to extensively build and customize a portal with specialized functionality. The WebLogic Portal framework comes packaged with an Eclipse-based integrated development environment (IDE) to assemble and extend the capabilities of the portal using the provided API and tools. The paired IDE is known as Oracle Workshop for WebLogic (formerly Workspace Studio). WebLogic Portal offers support for industry standards, enterprise-class portal federation, publication, and syndication capabilities including bidirectional integration with other portals and Web applications. My HealthVet (MHV) and the Clinical Information Support System (CISS) are deployed with WebLogic Portal.
4	Oracle Database	The Oracle relational database management system. There are several Oracle editions (Express, Personal, Standard, Enterprise, and Real Application Cluster). This assessment is concerned with the Standard and Enterprise editions of Oracle.
5	CA Single Sign-On	CA Single Sign-On improves security and simplifies user access by automating login to applications through a single authentication. This enables implementation of stronger security practices without burdening users with remembering multiple username and password combinations.
6	Radiant Logic	Radiant Logic acts as a virtual user store from multiple endpoints. It has evolved into an easy-to-use, enterprise-grade solution for stronger authentication and richer authorization.

#### 6.2.1.14. User Characteristics

Refer to section 1.5 and section 3.1.3 for user-related information.

#### 6.2.1.15. Dependencies and Constraints

Refer to section 2.4 for SSOi constraints and dependencies.

### 6.2.2. Specific Requirements

This SDD provides the foundational detailed design for AcS activities under VA Development Support program. VA AcS components leverage the installation and configuration of COTS products to meet the technical requirements that sufficiently meet the detailed functional requirements. The design applies specific configurations and customizations made to the base infrastructure to create the technical solution necessary to meet the business requirements provided in requirements documents listed in section 1.6.



#### 6.2.2.1. Database Repository

TBD.

#### 6.2.2.2. System Features

TBD

#### 6.2.2.3. Design Element Tables

N/A.

##### 6.2.2.3.1. Routines (Entry Points)

N/A

##### 6.2.2.3.2. Templates

N/A

##### 6.2.2.3.3. Bulletins

N/A

##### 6.2.2.3.4. Data Entries Affected by the Design

N/A

##### 6.2.2.3.5. Unique Record(s)

N/A

##### 6.2.2.3.6. File or Global Size Changes

N/A

##### 6.2.2.3.7. Mail Groups

N/A

##### 6.2.2.3.8. Security Keys

Table 31: Pre-Production PKI Certificate List

Computer Name (Hostname)	Common Name (CN)	FQDN	Cert Type	Issuer	Cert Function	Description	Comments

Computer Name (Hostname)	Common Name (CN)	FQDN	Cert Type	Issuer	Cert Function	Description	Comments
-----------------------------	---------------------	------	--------------	--------	------------------	-------------	----------

--	--	--	--	--	--	--	--

Computer Name (Hostname)	Common Name (CN)	FQDN	Cert Type	Issuer	Cert Function	Description	Comments
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]

Table 32: Production Server PKI Certificate List

Computer Name (Hostname)	Common Name (CN)	FQDN	Cert Type	Issuer	Cert Function	Comments
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]



Computer Name (Hostname)	Common Name (CN)	FQDN	Cert Type	Issuer	Cert Function	Comments
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]

#### 6.2.2.3.9. Options

N/A

#### 6.2.2.3.10. Protocols

N/A

#### 6.2.2.3.11. Remote Procedure Call (RPC)

N/A

#### 6.2.2.3.12. Constants Defined in Interface

N/A

#### 6.2.2.3.13. Variables Defined in Interface

N/A

#### 6.2.2.3.14. Types Defined in Interface

N/A

<b>6.2.2.3.15.</b>	<b>GUI</b>
N/A	
<b>6.2.2.3.16.</b>	<b>GUI Classes</b>
N/A	
<b>6.2.2.3.17.</b>	<b>Current Form</b>
N/A	
<b>6.2.2.3.18.</b>	<b>Modified Form</b>
N/A	
<b>6.2.2.3.19.</b>	<b>Components on Form</b>
N/A	
<b>6.2.2.3.20.</b>	<b>Events</b>
N/A	
<b>6.2.2.3.21.</b>	<b>Methods</b>
N/A	
<b>6.2.2.3.22.</b>	<b>Special References</b>
N/A	
<b>6.2.2.3.23.</b>	<b>Class Events</b>
N/A	
<b>6.2.2.3.24.</b>	<b>Class Methods</b>
N/A	
<b>6.2.2.3.25.</b>	<b>Class Properties</b>
N/A	
<b>6.2.2.3.26.</b>	<b>Uses Clause</b>
N/A	
<b>6.2.2.3.27.</b>	<b>Forms</b>
N/A	
<b>6.2.2.3.28.</b>	<b>Functions</b>
N/A	
<b>6.2.2.3.29.</b>	<b>Dialog</b>
N/A	
<b>6.2.2.3.30.</b>	<b>Help Frame</b>
N/A	

#### 6.2.2.3.31. HL7 Application Parameter

N/A

#### 6.2.2.3.32. HL7 Logical Link

N/A

#### 6.2.2.3.33. COTS Interface

N/A

### 6.3. Network Detailed Design

Refer to section 4.3 for detailed network design for the SSOi solution.

### 6.4. Service Oriented Architecture / ESS Detailed Design

Details will be provided at the end of Sprint 3.

Refer to SSOe SDD

Secure Token Service:

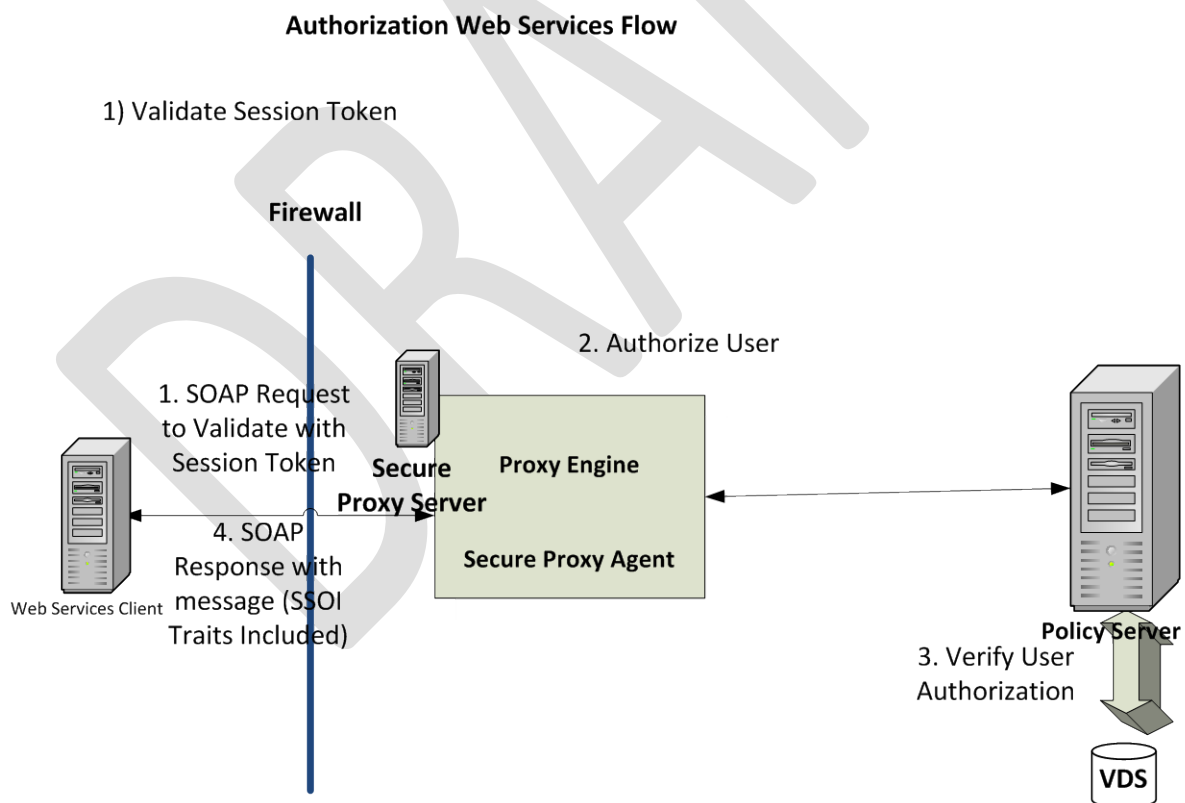


Figure 37: Secure Token Service

**Table 33: SSOi Support for Secure Token Service**

Field	Description
Use Case Name	SSOi Support for Secure Token Service
Description	This use case describes the process by which Webservices client calls the Secure Proxy server to validate the Session token
Actors	6. Users 7. SSOi 8. SSOi Integrated Application(s) 9. User Directory 10. VDS User Store
Pre-Conditions	A valid WS integration/ trust between Web service Client and SPS
Constraints	SSOi Service will be depend the capability of VDS attribute service to get appropriate Traits
Trigger	Webservices client calls the SPS Authorization service with valid Session token
Actions	10. SPS validate the session token using siteminder policy server 11. During evaluation of authorization policies SiteMinder policy server call the VDS user store with input as userid 12. VDS returns the attribute set (Traits) to SiteMinder policy server at the run time. 13. SiteMinder set them on http headers as response and provide it back to the SPS. 14. SPS returns all the SSOi Traits to the Webservice client in SOAP Response
Main Success Scenarios	User's session is validated and SSOi Traits returned to user.
Main Failure Scenarios	Failure to receive attribute will result in blank response, which will be handled by Web services client to display error.

#### **6.4.1. Service Description for <Consumed Service Name>**

#### **6.4.2. Service Design for <Provided Service Name>**

TBD

##### **6.4.2.1. Introduction**

##### **6.4.2.1.1. Purpose and Scope of Service**

TBD

##### **6.4.2.1.2. Links to Other Documents**

N/A



#### **6.4.2.2. Service Details**

N/A

##### **6.4.2.2.1. Service Identification**

N/A

##### **6.4.2.2.2. Service Versions**

N/A

##### **6.4.2.2.3. Summary of Design and Platform Details**

N/A

###### **6.4.2.2.3.1. SOA Pattern(s) Implemented**

TBD

###### **6.4.2.2.3.2. COTS Platform vendor names and versions for hosting platform**

N/A

#### **6.4.2.3. Dependencies**

N/A

#### **6.4.2.4. Service Design Details**

TBD

##### **6.4.2.4.1. Interface Technical Specs**

TBD

###### **6.4.2.4.1.1. Service Invocation Type**

TBD

###### **6.4.2.4.1.2. Service Interface Type**

TBD

###### **6.4.2.4.1.3. Service Name**

TBD

###### **6.4.2.4.1.4. Interface**

TBD

###### **6.4.2.4.1.5. End Points**

TBD

###### **6.4.2.4.1.6. Operations or Methods**

TBD

Operation Name	Inputs	Outputs	Transactional Qualities if relevant (Updating?, Atomic?, Can participate in transaction?)	Pre and Post Conditions	Exception (s)
TBD	TBD	TBD	TBD	TBD	TBD
TBD	TBD	TBD	TBD	TBD	TBD

#### 6.4.2.4.1.7. Message Schemas

TBD

#### 6.4.2.4.2. Information Model

TBD

#### 6.4.2.4.2.1. Class Diagram and Description of Entities Involved

TBD

#### 6.4.2.4.2.2. Mappings from ELDM to Standards Based Schemas

TBD

#### 6.4.2.4.3. Behavior Model (AKA Use Case Realization)

TBD

#### 6.4.2.4.3.1. Use Cases (Use Case Model)

TBD

#### 6.4.2.4.3.2. Interaction Diagrams

TBD

#### 6.4.2.5. Gap Analysis

TBD

Design Elements→ Policies / SLD elements etc.↓	Design Element A	Design Element B	Design Element C	Comment for non-conformance
TBD	TBD	TBD	TBD	TBD
TBD	TBD	TBD	TBD	TBD
TBD	TBD	TBD	TBD	TBD
TBD	TBD	TBD	TBD	TBD

#### 6.4.2.5.1. Variances from Enterprise Target Architecture

TBD

#### 6.4.2.5.2. Variances from SLDs

TBD

#### 6.4.2.5.3. Variances from Standards and Policies

TBD

#### 6.4.2.5.4. Justification for Exceptions and Mitigation

TBD

## 7. External System Interface Design

Refer to the [master Interface Control Documents \(ICDs\)](#) and [integration ICDs](#), which are available on the VA SharePoint site.

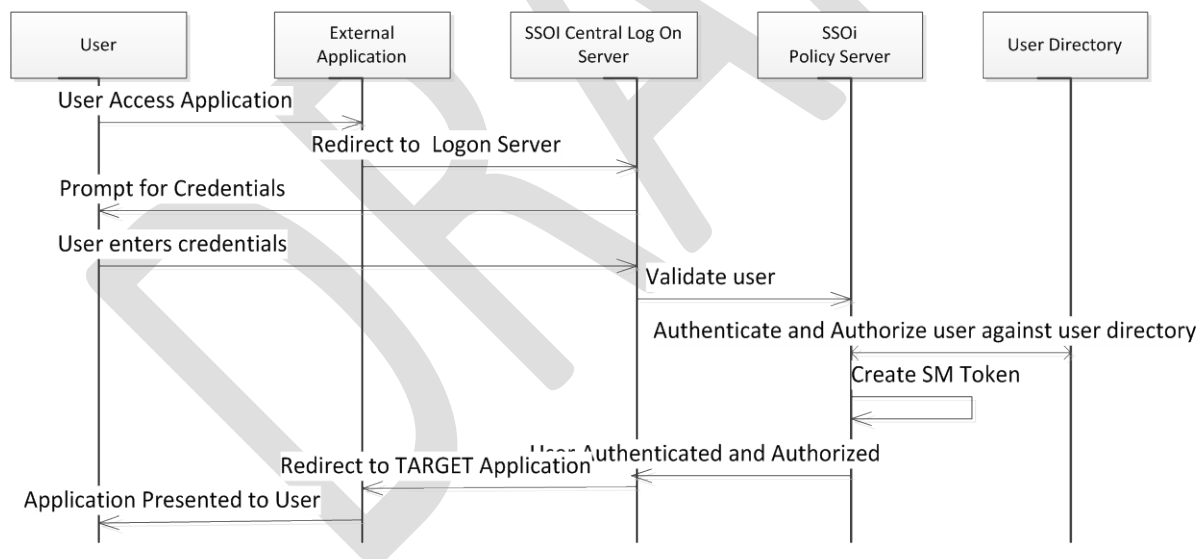
### 7.1. Interface Architecture

TBD

### 7.2. Interface Detailed Design

#### 7.2.1. WebAgent Pattern

External applications can integrate with SSOi using the webagent pattern. In this model, Webagent is installed on external application webserver and communicated to policy server



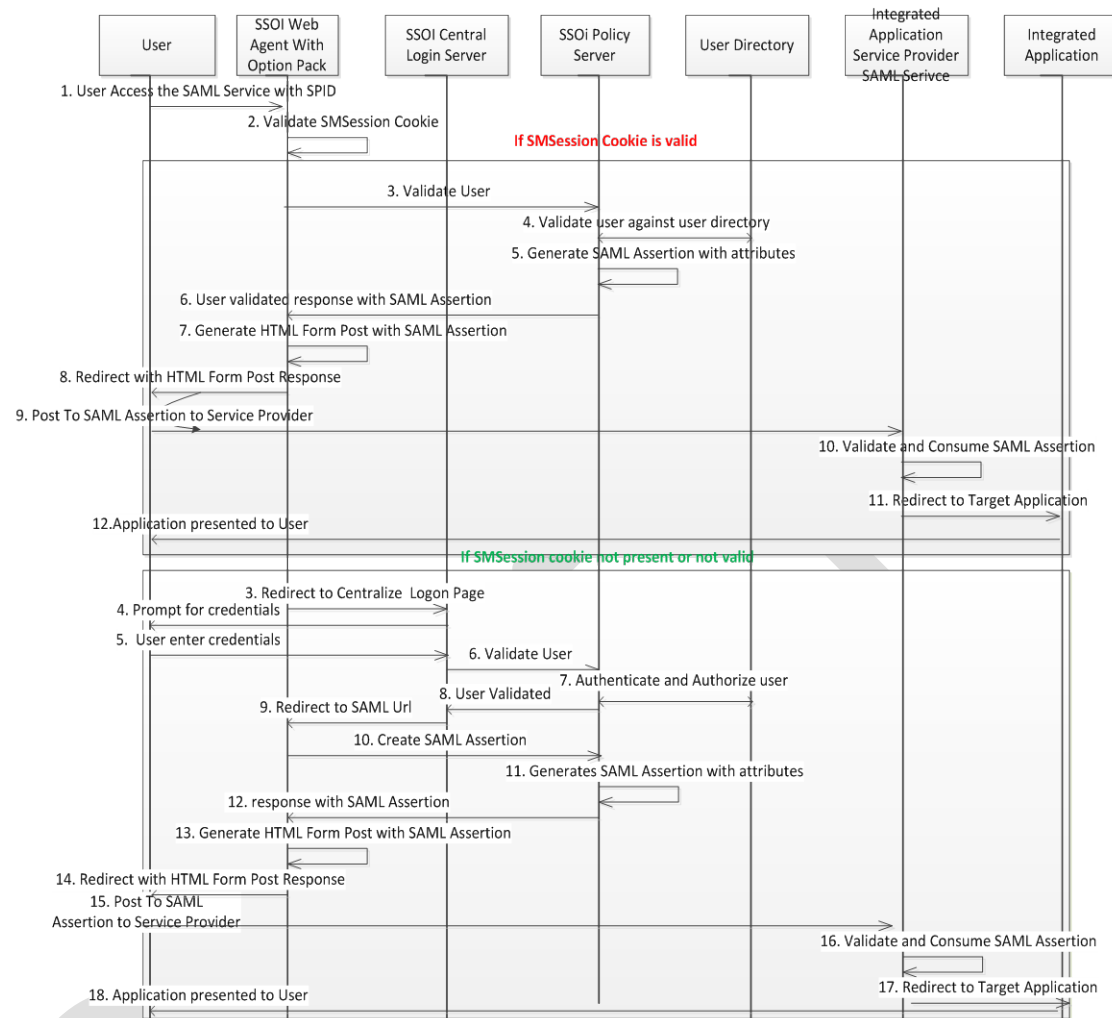
**Figure 38: WebAgent Integration Pattern**

1. User access the application URL in browser.
2. Integrated application webagent contacts SSOi policy server to identify the resource is protected.
3. If the resource on Integrated application webserver is protected, then policy server notifies the application webserver to redirect the user to SSOi Central Login page for Authentication
4. SSOi Central Logon Server prompts for credentials

5. The user enters the credentials
6. The SSOi Central Logon Server passes the control to SiteMinder Policy server to authorize the user
7. SSOi SiteMinder Policy Server authenticates and authorizes user against Active Directory
8. SSOi SiteMinder Policy Server create SiteMinder Token
9. User is authenticated and authorized to access the resource by SSOi SiteMinder Policy Server
10. The SSOi Central Logon server redirects the user to Integrated application

### **7.2.2. Federation Pattern**

External applications can integrate with SSOi using the Federation pattern. In this model, Identity Provider creates a SAML assertion and posts into Service Provider. SSOi can act as both Identity Provider and Service Provider, depending upon the integration application requirement.



**Figure 39: SSOI as Identity provider**

1. An internal user accesses an application (IdP-protected URL) that is at service provider without a SiteMinder session cookie.
2. Web server redirects user to SSOi centralize log on page and prompts for authentication credential.
3. User enters the credentials.
4. SSOi SiteMinder Policy server validates against User store.
5. SSOi SiteMinder Policy server authenticates and authorizes the user.
6. SSOi SiteMinder Policy Server creates valid user token.
7. SSOi SiteMinder Policy server redirects to SAML URL.
8. SSOi SiteMinder Policy server generates the SAML Assertion.
9. SSOi SiteMinder Policy Server adds the required attributes such as user Principal Name (UPN), email, firstname and lastname.

10. SSOi IdP posts the SAML assertion to the Integrated Application Service Provider SAML Assertion Consumer service.
11. Integrated application Service provides Consumes the SAML assertion generated by SSOi and grants the access to the user.
12. Integrated application Service provider redirect to Integrated application.
13. Integrated application presented to the end user.

#### **7.2.2.1. Identity Provider (IdP) – With a Valid Session Cookie**

1. An internal user accesses an application (IdP protected URL) which is at service provider with a SiteMinder Session cookie.
2. The user is validated by SSOi SiteMinder Policy Server.
3. SSOi SiteMinder Policy Server generates the SAML assertion by adding all the required attributes such as user Principal Name (UPN), email, firstname and lastname.
4. SSOi IdP posts the SAML assertion to the Service Provider SAML Assertion Consumer service:
5. Integrated application Service provides. Consumes the SAML assertion generated by SSOi and grants the access to the user.
6. Integrated application Service provider redirects to Integrated application.
7. Integrated application presented to the end user.

#### **7.2.2.2. Service Provider (SP)**

1. An internal user accesses an application that is protected by an Integrated IdP.
2. User enters the credentials and validated with IdP.
3. The SAML assertion is generated by adding the required attributes such as user Principal Name (UPN), email, firstname, and lastname by IdP.
4. Integrated application IdP posts the SAML assertion to the Service Provider that is configured at SiteMinder.
5. SSOi Service provider consumes the SAML assertion generated by IdP.
6. SSOi Service provider validates the user attributes.
7. SSOi Service provider creates SiteMinder Token.
8. SSOi SAML Assertion is consumed by Service provider.
9. User is redirected protected SSOi application with valid SiteMinder session by SSOi Service provider.

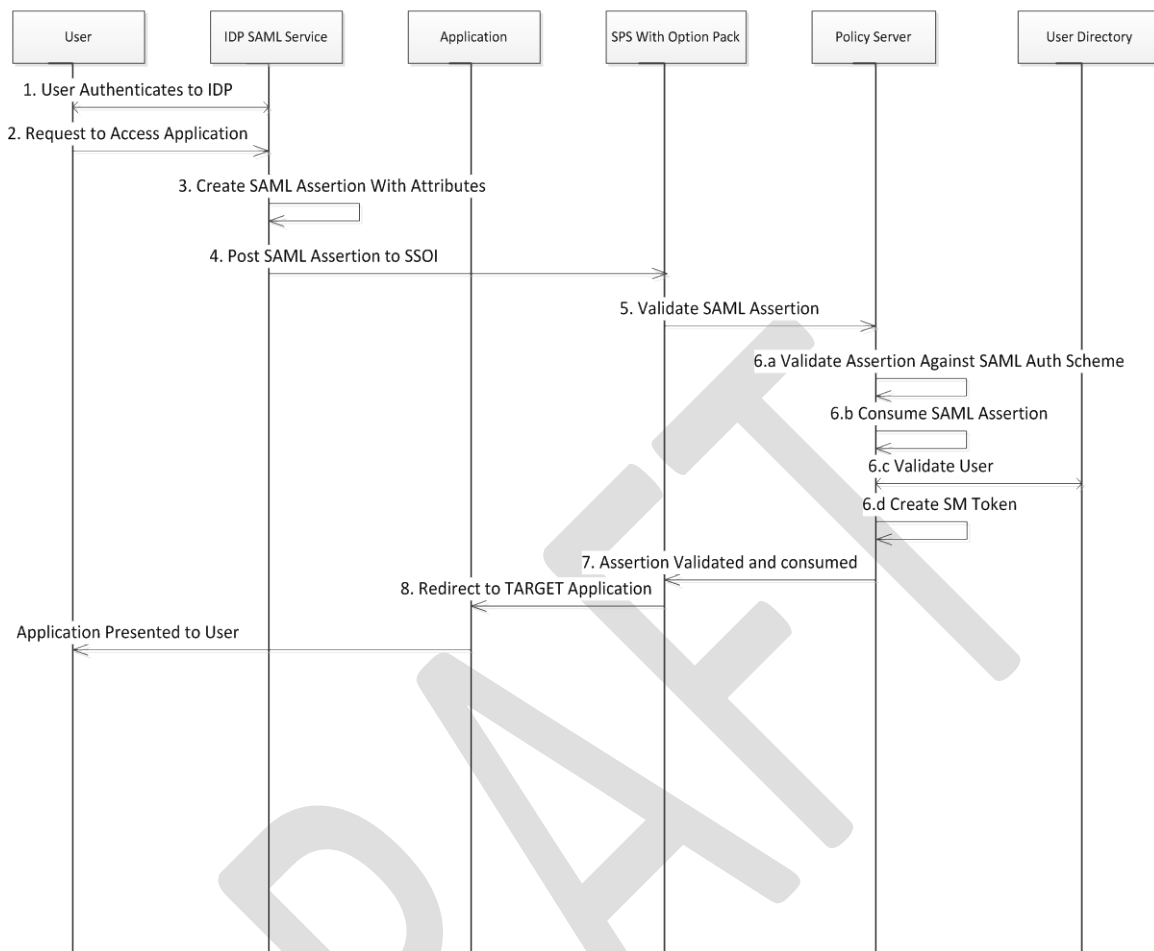


Figure 40: SSOi as Service Provider

## 8. Human-Machine Interface

For user interface information related to COTS administrator functions, refer to the product documentation available at the following websites:

- CA support site: <https://support.ca.com>Active Directory
- Oracle support site: <https://support.oracle.com>
- IBM support site: <https://www.ibm.com/support>
- Radiant Logic site: <http://www.radiantlogic.com>

Refer to section 3.2.3, which provides the interfaces that AcS activities use as appropriate for the end users.

### 8.1. Interface Design Rules

The following design rules apply to the user interfaces for the SSOi activities:

- The user and administrator interfaces comply with VA's branding specifications. Consistent sequences of actions should be required in similar situations. Identical

terminology should be used in prompts, menus, and help screens, and consistent color, layout, capitalization, fonts, and so on should be employed throughout.

- Exceptions, such as required confirmation of the delete command or no echoing of passwords, should be comprehensible and limited in number.
- The interface is easy to navigate with self-explanatory instructions / fields.
- Recognize the needs of diverse users and design for plasticity, facilitating transformation of content. Novice to expert differences, age ranges, disabilities, and technological diversity each enrich the spectrum of requirements that guides design. Adding features for novices, such as explanations, and features for experts, such as shortcuts and faster pacing, can enrich the interface design and improve perceived system quality.
- The interface provides user-friendly messages / information on error. Every user action should have system feedback. For frequent and minor actions, the response can be modest, whereas for infrequent and major actions, the response should be more substantial. Visual presentation of the objects of interest provides a convenient environment for showing changes explicitly.
- The interface supports web browsers using Internet Explorer 7 (IE7), for Windows XP, IE9 for Windows7, and Mozilla Firefox3.6.23.
- The interface is Section 508 compliant (for nonadministrator, end-user facing interfaces); the exception is CAR.
- The web interface provides necessary validation checks such as blanks for mandatory fields, special characters, and invalid email id format before form submission.
- SSOi error codes include:
  - Regular SiteMinder Integration /Proxy Based Integration
    - OnAuthAttempt (User not found) – Redirect to failedlogin.aspx
    - OnAuthReject (User enters invalid credentials) – Redirect to failedlogin.aspx
    - OnAccessReject (User not Authorized to access resource) – Redirect to failedlogin.aspx
    - Server Error (500,401,403) – Graceful handling not implemented currently
    - IdleTimeout – Forward to Login Page
  - Federation: As Service Provider
    - User Not Found – Redirect to failedlogin.aspx
    - Invalid SSO Message – Redirect to failedlogin.aspx
    - Unaccepted User Credential (SSO Message) – Redirect to failedlogin.aspx
    - Server Error – Graceful handling not implemented currently
    - Invalid Request – Graceful handling not implemented currently
    - Unauthorized Access – Redirect to failedlogin.aspx
  - Federation: As Identity Provider
    - Server Error – Graceful handling not implemented currently



- Invalid Request – Redirect to login page
- Unauthorized Access – Redirect to failedlogin.aspx

## **8.2. Inputs**

This section is not applicable.

## **8.3. Outputs**

In addition to web-based output and the ability to save web pages using native browser options, the following report media are generated by SSOi:

- PDF
- Comma Separated File (CSF)
- Excel

## **8.4. Navigation Hierarchy**

Central Login is the .net application hosted on Central Login servers. This application gives the functionality for login, login error handling, session time out, and logout flow pages. The below is hierarchy of the pages in Central login application.

inetpub ▾ wwwroot ▾ CentralLogin ▾




















with ▾ New folder

Name ▴	Date modified	Type	Size
bin	10/7/2014 12:22 PM	File folder	
core	6/12/2014 8:04 PM	File folder	
CustomError	12/16/2014 12:21 PM	File folder	
obj	6/12/2014 8:04 PM	File folder	
Properties	6/12/2014 8:04 PM	File folder	
resources	6/12/2014 8:04 PM	File folder	
azfailed	12/15/2014 1:15 PM	ASPX File	1 KB
centrallanding	12/12/2014 3:27 PM	ASPX File	2 KB
concurloginfailed	2/25/2014 3:40 PM	ASPX File	1 KB
contactus	3/10/2014 10:50 AM	ASPX File	2 KB
contactus.aspx	11/27/2013 9:39 AM	CS File	1 KB
contactus.aspx.designer	11/27/2013 9:39 AM	CS File	1 KB
contactUsMobile	5/14/2014 7:50 PM	ASPX File	3 KB
contactUsMobile.aspx	4/9/2014 3:11 PM	CS File	1 KB
contactUsMobile.aspx.designer	4/9/2014 4:36 PM	CS File	1 KB
cspluginfailed	12/15/2014 1:14 PM	ASPX File	1 KB
cspstyle	10/3/2014 2:18 PM	Cascading Style Sh...	11 KB
csserror	9/11/2014 1:27 PM	ASPX File	1 KB
Default	3/11/2015 2:12 PM	ASPX File	11 KB

**Figure 41: Central Login Application Page Hierarchy (1 of 7)**

inetpub ▾ wwwroot ▾ CentralLogin ▾				
with ▾ New folder				
Name ^	Date modified	Type	Size	
Error	11/27/2013 9:39 AM	ASPX File	1 KB	
Error.aspx	11/27/2013 9:39 AM	CS File	1 KB	
Error.aspx.designer	11/27/2013 9:39 AM	CS File	1 KB	
exception	10/3/2014 12:42 PM	ASPX File	1 KB	
failedlogin	12/15/2014 1:07 PM	ASPX File	1 KB	
failedlogin.aspx.bak	5/14/2014 4:38 PM	BAK File	1 KB	
Header.Master	1/2/2015 10:09 AM	MASTER File	6 KB	
Header.Master	11/27/2013 9:39 AM	CS File	1 KB	
Header.Master.designer	11/27/2013 9:39 AM	CS File	2 KB	
help	11/27/2013 9:39 AM	ASPX File	2 KB	
help.aspx	11/27/2013 9:39 AM	CS File	1 KB	
help.aspx.designer	11/27/2013 9:39 AM	CS File	1 KB	
hrisloginfailed	5/19/2014 8:40 PM	ASPX File	1 KB	
HSPDhelp	11/27/2013 9:39 AM	ASPX File	1 KB	
HSPDhelp.aspx	11/27/2013 9:39 AM	CS File	1 KB	
HSPDhelp.aspx.designer	11/27/2013 9:39 AM	CS File	1 KB	
idletimeout	12/19/2014 1:44 PM	ASPX File	1 KB	
ipad	5/14/2014 7:38 PM	ASPX File	14 KB	
ipad.aspx	5/13/2014 8:47 PM	CS File	3 KB	
ipad.aspx.designer	5/13/2014 7:18 PM	CS File	2 KB	

Figure 42: Central Login Application Page Hierarchy (2 of 7)

inetpub ▾ wwwroot ▾ CentralLogin ▾					Search
with ▾ New folder					
Name ▴	Date modified	Type	Size		
 ipad.aspx	5/13/2014 8:47 PM	CS File	3 KB		
 ipad.aspx.designer	5/13/2014 7:18 PM	CS File	2 KB		
 iphone	5/14/2014 7:50 PM	ASPX File	22 KB		
 iphone.aspx	5/14/2014 11:47 AM	CS File	3 KB		
 iphone.aspx.designer	5/14/2014 11:47 AM	CS File	2 KB		
 iploginfailed	12/15/2014 1:14 PM	ASPX File	1 KB		
 IWAhelp	11/27/2013 9:39 AM	ASPX File	1 KB		
 IWAhelp.aspx	11/27/2013 9:39 AM	CS File	1 KB		
 IWAhelp.aspx.designer	11/27/2013 9:39 AM	CS File	1 KB		
 loggedout	12/12/2014 4:25 PM	ASPX File	2 KB		
 mvloginfailed	12/15/2014 1:15 PM	ASPX File	1 KB		
 PIVContactUS	4/14/2014 4:14 PM	ASPX File	2 KB		
 PIVContactUS.aspx	11/27/2013 9:39 AM	CS File	1 KB		
 PIVContactUS.aspx.designer	11/27/2013 9:39 AM	CS File	1 KB		
 PIVdefault	7/3/2014 2:45 PM	ASPX File	8 KB		
 PIVdefault.aspx	2/19/2014 2:28 PM	CS File	1 KB		
 PIVdefault.aspx.designer	11/27/2013 9:39 AM	CS File	2 KB		
 pivHead.Master	3/3/2014 2:13 PM	MASTER File	4 KB		
 pivHead.Master	11/27/2013 9:39 AM	CS File	1 KB		

**Figure 43: Central Login Application Page Hierarchy (3 of 7)**

netpub ▾ wwwroot ▾ CentralLogin ▾			
with ▾ New Folder			
Name ^	Date modified	Type	Size
PIVContactUS.aspx	11/27/2013 9:39 AM	CS File	1 KB
PIVContactUS.aspx.designer	11/27/2013 9:39 AM	CS File	1 KB
PIVdefault	7/3/2014 2:45 PM	ASPX File	8 KB
PIVdefault.aspx	2/19/2014 2:28 PM	CS File	1 KB
PIVdefault.aspx.designer	11/27/2013 9:39 AM	CS File	2 KB
pivHead.Master	3/3/2014 2:13 PM	MASTER File	4 KB
pivHead.Master	11/27/2013 9:39 AM	CS File	1 KB
pivHead.Master.designer	11/27/2013 9:39 AM	CS File	2 KB
PIVLogin	11/27/2013 9:39 AM	ASPX File	5 KB
provloginfailed	12/15/2014 1:14 PM	ASPX File	1 KB
teamsiteloginfailed	12/15/2014 1:16 PM	ASPX File	1 KB
UIDhelp	11/27/2013 9:39 AM	ASPX File	1 KB
UIDhelp.aspx	11/27/2013 9:39 AM	CS File	1 KB
UIDhelp.aspx.designer	11/27/2013 9:39 AM	CS File	1 KB
vicloginfailed	12/15/2014 1:14 PM	ASPX File	1 KB
Web	4/16/2015 2:00 PM	CONFIG File	3 KB

**Figure 44: Central Login Application Page Hierarchy (4 of 7)**

inetpub ▾ wwwroot ▾ CentralLogin ▾ core ▾			
with ▾ New Folder			
Name ^	Date modified	Type	Size
IWA	6/12/2014 8:04 PM	File folder	
PIV	6/12/2014 8:04 PM	File folder	
redirect	3/10/2014 1:24 PM	ASPX File	1 KB
redirect.aspx	2/19/2014 2:03 PM	CS File	1 KB
redirect.aspx.designer	2/19/2014 2:07 PM	CS File	1 KB

**Figure 45: Central Login Application Page Hierarchy (5 of 7)**

Name	Date modified	Type	Size
redirect	2/19/2014 3:38 PM	ASPX File	1 KB

**Figure 46: Central Login Application Page Hierarchy (6 of 7)**

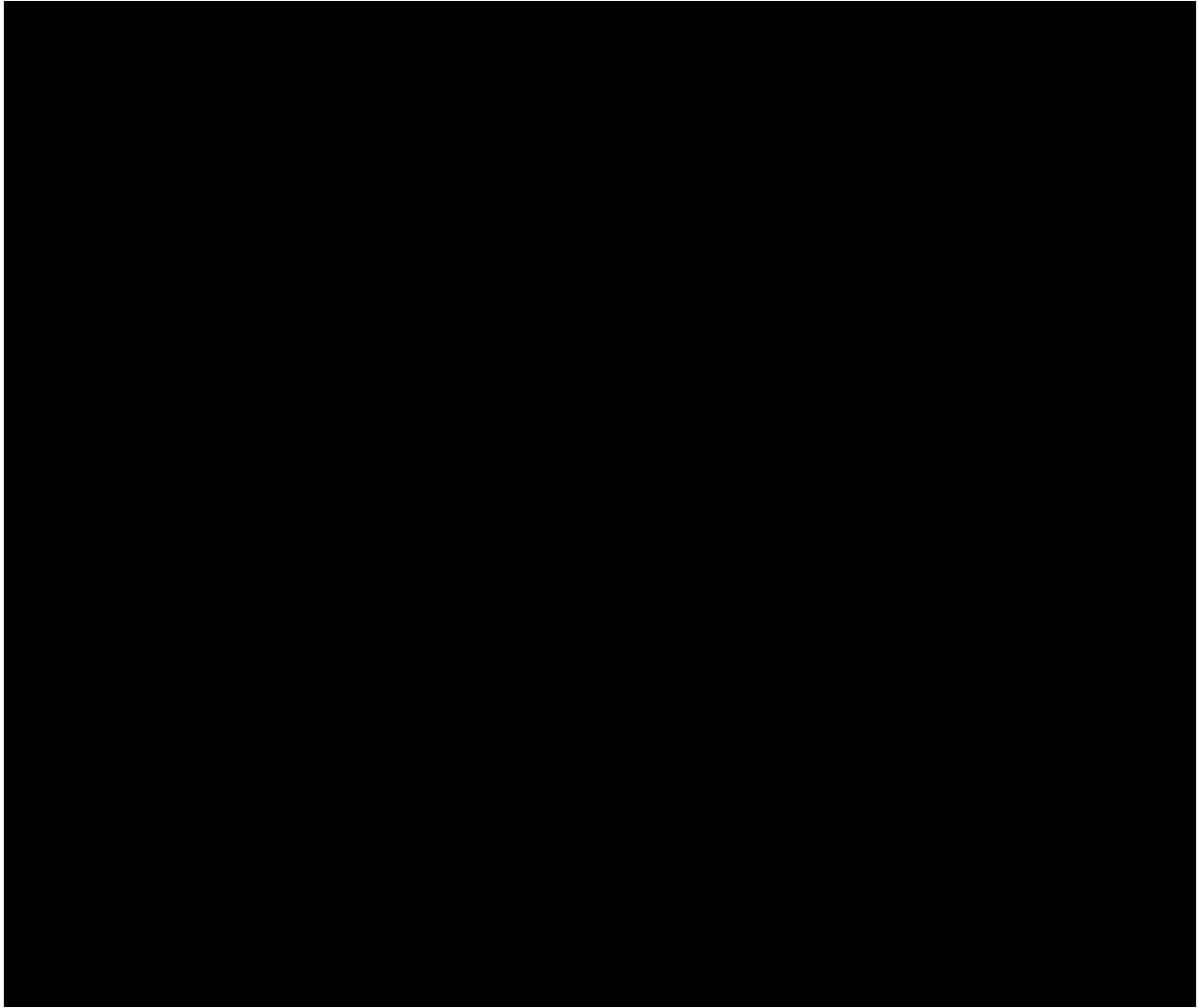
Name	Date modified	Type	Size
piv	2/19/2014 3:38 PM	ASPX File	1 KB

**Figure 47: Central Login Application Page Hierarchy (7 of 7)**

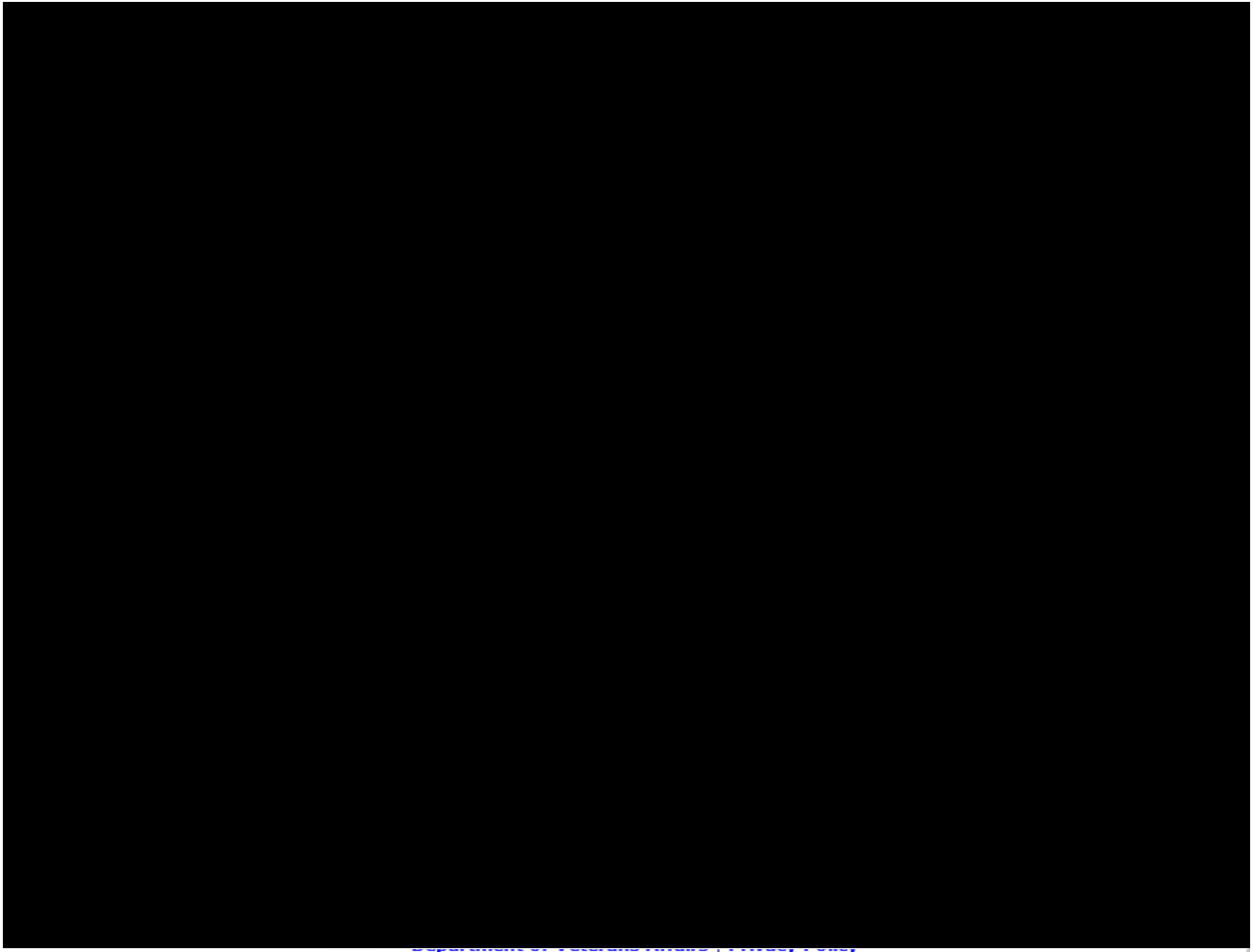
## 8.5. Maintenance Pages

During system maintenance or outages, customer/user facing Graphical User Interfaces (GUI) have to be updated to inform the specific SSOi audience of the state of the system, what the expected return to service timeframe is, and any additional references (e.g., help desk numbers, ANR information) that system users may need to allow them to either use an out-of-band process or be kept informed of any progress, as needed.

The following figure reflects the general form of the Maintenance page to be displayed for any AcS component with customer/user facing GUI during a period of either scheduled or unscheduled maintenance.



**Figure 48: Maintenance Page for AcS Component**



**Figure 49: SSOi Maintenance Page**

Each of the following AcS sub-systems (IP, CSP, Prov, SSOi Central Login, CAR, and SailPoint) will be updated to include a customized maintenance page with appropriate system reference on each page.

The AcS e-Sig, SAC and VDS services do not provide a GUI for users or customers. Any maintenance being performed on those services will have to be communicated to the customers/partners consuming those services so they can display the appropriate notification for partial or full loss of functionality/service on their respective GUIs.

**Note:** Administrative GUIs for the AcS COTS products will not have a maintenance page, as they are not exposed to external audience and are necessary for the actual maintenance process.

In order to allow for a centralized control of maintenance page requirements, maintenance page enforcement policies in Siteminder Policy server shall control the redirects to the appropriate maintenance pages for IP, CSP, Provisioning, SSOi Central Login, and SailPoint. Individual policies per AcS component will ensure each component can be enforced separately or in a group. Policies will be enabled and disabled as dictated by the deployment plans for each production deployment for the above listed components.



CAR maintenance pages will be handled through rules on the Apache servers supporting each of the CAR UI components. Those rules will be enabled and disabled as dictated by the deployment plans for each production deployment of CAR.

Maintenance pages will have two principal locations, dictated by the various circumstances in which maintenance can occur.

Maintenance pages for all AcS system components, except CAR will be hosted on the Centralized login page servers as a primary location.

Maintenance that requires downtime of all Siteminder Policy servers, all Centralized login page servers, or any of the Policy server supporting user stores in effect disallows policy evaluation and enforcement for Maintenance Page policies. When this occurs, a secondary hosting location will be used on each of the forward facing web servers for each AcS component (CAR falls under this category because it does not integrate with SSOi).

Local WebServer rules will be configured to enforce the complete bypass of any requests for application resources and redirect to a web server available to host the maintenance pages, with the same business rules as the Siteminder Policies. These web server maintenance page rules will follow the same enforcement by the deployment plan for each production deployment.

Since the individual WebServer rule usage will require multiple server modifications, care needs to be exercised not to omit any of the applicable web servers and not enable rules on a component that should not be under maintenance.

Content of the Maintenance pages will be updated before maintenance period's start in both primary and secondary hosting locations as well as during the maintenance to ensure consistency.

For future implementations, a CMS like TeamSite or similar may be used to provide ease of rolling out maintenance pages.

As an alternative, Maintenance pages can be hosted at a LoadBalancer appliance, but this approach will make the control of the maintenance pages outside of the AcS control and requires AIDE involvement at each maintenance period.

## **9. Security and Privacy**

### **9.1. Security**

Data security is critical for VA to safeguard user information and ensure that data in motion as well as rest is secured properly. For the SSOi, the following security measures and integrity controls are in place.

#### **9.1.1. Data in Motion**

“Data in Motion” is secured using the combination of FIPS encryption and VA issued certificates. Internal communications between CA components are encrypted using the cryptographic libraries that meet FIPS requirement. CA IdentityMinder uses the Advanced Encryption Standard (AES) adapted by the US Government. CA IdentityMinder incorporates the RSA Crypto-J v3.5 and Crypt-C ME v2.0 cryptographic libraries, which have been validated as

meeting the FIPS 140-2 Security Requirements for Cryptographic Modules. CA SiteMinder Policy Server uses certified FIPS140-2 (AES) compliant cryptographic libraries.

CA UARM uses its own trusted root certificate, which is incorporated across agent and component communications. For AcS system internal communications, there is no compelling need these certificates to be replaced with VA Internal Certificate Authority (CA) or commercially trusted CA issued ones.

For communications outside of the AcS environment, certificates issued by VA Internal CA will be used for securing communications between the AcS and VA internal systems/applications and commercially trusted certificates will be used when the communication is exposed to external to VA clients and/or third parties.

### 9.1.2. Data at Rest

The following table explains the “data at rest” points.

**Table 34: Data Points and Security**

<b>Data Points</b>	<b>Data Type</b>	<b>Explanation</b>
Oracle	Sensitive	<ul style="list-style-type: none"><li>• Stores the audit log for SiteMinder and needs to be secured, but not encrypted, as there is no PII.</li><li>• Stores the audit log for CA IDM and must be encrypted and secured for PII.</li><li>• See vendor documentation for additional information regarding actual encryption algorithms used.</li></ul>
Directory	Sensitive	<ul style="list-style-type: none"><li>• Stores encrypted SiteMinder policy data.</li><li>• Stores SiteMinder user data. Only sensitive user attributes will be encrypted.</li><li>• See vendor documentation for additional information regarding actual encryption algorithms used.</li></ul>
File Store	Non-Sensitive/ Sensitive	<ul style="list-style-type: none"><li>• IM is stored in a JMS data in file system and contains transactional data. It does not contain any sensitive information.</li><li>• A FIPS encryption key file is stored in the file system. Access to the file should be restricted and enforced by setting the directory/file access permissions for specific groups and/or users.</li></ul>

The security controls for the data at reset are managed through the encryption of sensitive attributes at the directory level for the SSOi. The FIPS 140-2 encryption is applied on the identified PII and sensitive attributes stored in the SSOi directory attributes.

## 9.2. Privacy

The SSOi service only allows access to authenticated users. SSOi configures user authentication according to federal and VA security policies. The SSOi service integrates with the CAR

framework for auditing and reporting. The system stores authentication information only, no additional sensitive and PII is stored. SSOi implements proper access control to secure the user information.

### **9.2.1. Confidentiality of Sensitive Information**

The SSOi Service CA SSO toolset stores user profile and authentication information required for authentication only, and does not store any additional sensitive PII in CA Directory. The user password is stored in an Advanced Encryption Standard (AES) 256 encrypted/hashed format in CA Directory. The transmission of information occurs only over an SSL channel. The user information is secured using proper access control implementation. CA SiteMinder does not store user information; it connects to the appropriate user store to fetch the information.

### **9.2.2. Privacy of Personal Information**

The SSOi service does not store any Personally Identifiable Information (PII) of the user.

### **9.2.3. Process Integrity**

The SSOi service is designed to provide authentication services. The user authentication credentials are collected and validated. The user is only granted access to data and functionality that they are authorized to access.

### **9.2.4. System Availability**

The SSOi service implementation is highly available and provides controls to minimize system failures, and access control to minimize man-made failures.

## Attachment A – Approval Signatures

This section is used to document the approval of the System Design Document. The review should be conducted face to face where signatures can be obtained ‘live’ during the review. If unable to conduct a face-to-face meeting then it should be held via LiveMeeting and concurrence captured during the meeting. The Scribe should add /es/name by each position cited. Example provided below.

The Chair of the governing Integrated Project Team (IPT), Business Sponsor, IT Program Manager, and Project Manager are required to sign.

Signed: \_\_\_\_\_

Integrated Project Team Chair and Business Sponsor Date

Signed: \_\_\_\_\_

OIS Business Sponsor Date

Signed: \_\_\_\_\_

IAM Program Manager Date

Signed: \_\_\_\_\_

AcS Program Manager Date

Signed: \_\_\_\_\_

Chief Architect Date

Signed: \_\_\_\_\_

, Enterprise Architecture Date

Signed: \_\_\_\_\_

SDE

Signed: \_\_\_\_\_ Date: \_\_\_\_\_

## **A. Additional Information**

### **A.1. RTM**

The Requirements Traceability Matrix (RTM) is provided as a separate document.

### **A.2. Packaging and Installation**

The deployment package for Infrastructure will provide details for special considerations if any for each of the components. The CA SSO client is deployed as a package to the desktop by Enterprise System Engineering (ESE) team. Using the CA SSO client installation and configuration documentation and response files provided in the deployment package, the ESE package builds and automates the process of CA SSO client to users system

### **A.3. Design Metrics**

The design for IAM services is calculated based on requirements from PWS, BRD and CSP population estimates provided by VA. The CSP population estimate spreadsheet is attached below.



VA CSP User  
Population Estimates.

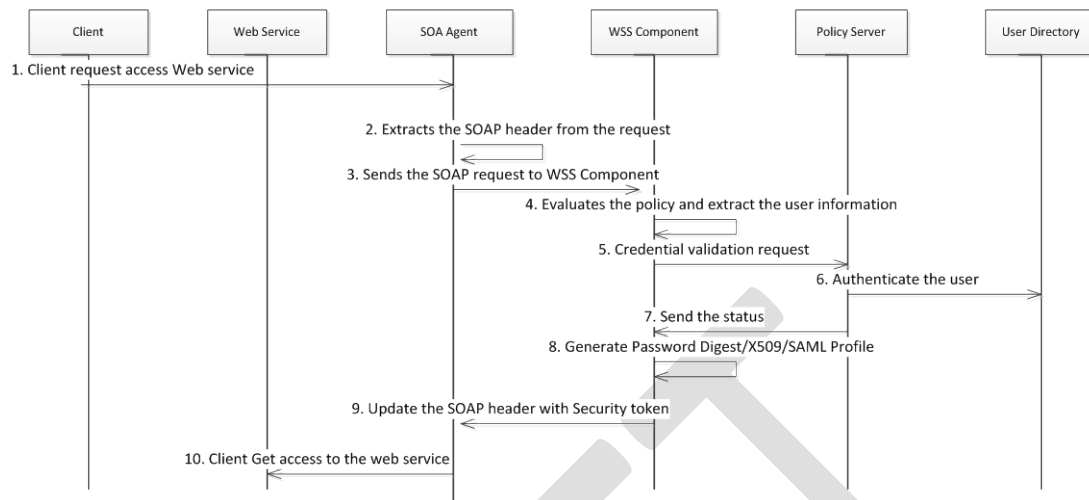
### **A.4. Acronym List and Glossary**

The acronyms and terms used in this SDD are defined in the [Identity and Access Services Master Glossary](#).

### **A.5. Required Technical Documents**

Refer to the CA vendor support/web site for detailed product documentation.

## A.6. Responses to Produce WS Security Headers



**Figure 50: Responses to Produce WS Security Headers Sequence Diagram**

**Table 35: Responses to Produce WS Security Headers**

Field	Description
Use Case Name	Responses to Produce WS Security Headers
Description	This use case describes the process by exchanges which SiteMinder generates and manages WS security headers.
Actors	<ol style="list-style-type: none"> <li>1. Users</li> <li>2. Client</li> <li>3. Web Service</li> </ol>
Pre-Conditions	A valid attribute service end point from VDS which provides a response with set of attributes for a request sent by SiteMinder
Trigger	Client access web service endpoint protected by SOA agent
Actions	<ol style="list-style-type: none"> <li>1. Client sends WS SOAP request to web service end point</li> <li>2. SOA agent intercepts WS SOAP request check for user credentials</li> <li>3. Extracts the SOAP header</li> <li>4. Sends the SOAP request to WSS Component</li> <li>5. WSS Component evaluates the policy and extracts the user information</li> <li>6. Sends Policy Server validation request</li> <li>7. Policy server validates the credentials from the input message and add the session token in to WS-header</li> <li>8. Sends the validation status</li> <li>9. Password Digest/X509/SAML Profile generated</li> <li>10. Update the SOAP header with Security token</li> <li>11. Client gets access to the web service.</li> </ol> <p><b>Alternate Flow</b></p>

Field	Description
	<ol style="list-style-type: none"> <li>1. Client sends WS SOAP request to web service end point</li> <li>2. SOA agent intercepts WS SOAP request check for session token</li> <li>3. Policy server validates the token and update WS-Security header</li> <li>4. Client gets access to the web service.</li> </ol>
Main Success Scenarios	User is authenticated and Application is presented to the user
Main Failure Scenarios	SOAP fault with authentication failure message returned to client in case of validation of user credential fail

## A.7. Responses to XML Encryptions, Decryptions, and Digital Signature

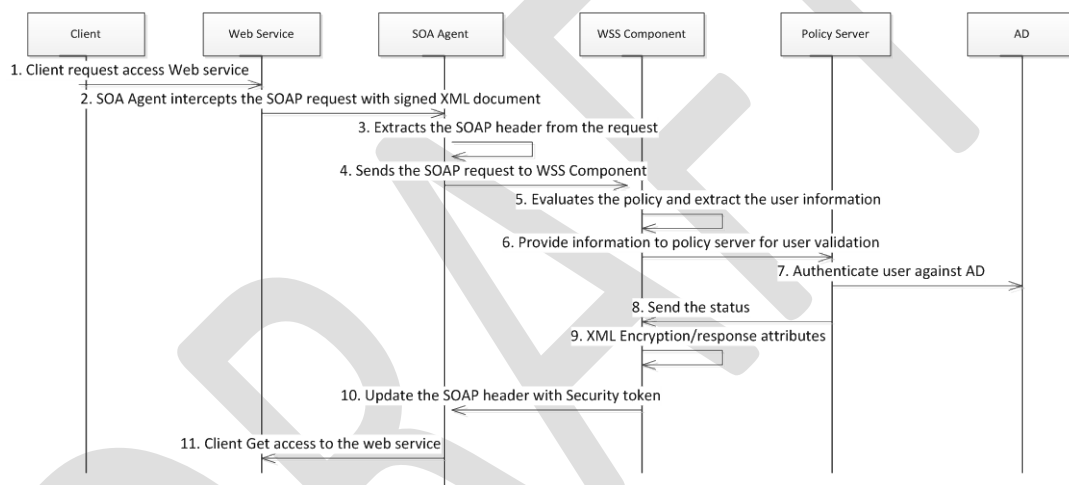


Figure 51: Responses to XML Encryptions, Decryptions, and Digital Signature Sequence Diagram

Table 36: Responses to XML Encryptions, Decryptions, and Digital Signature

Field	Description
Use Case Name	Responses to XML Encryptions, Decryptions and Digital Signature
Description	This use case describes the process by exchanges which SiteMinder generates and manages WS security headers.
Actors	<ol style="list-style-type: none"> <li>1. Users</li> <li>2. Client</li> <li>3. Web Service</li> </ol>
Pre-Conditions	A X509 certificate signer should be available to digitally sign a complete XML document
Trigger	Client access web service endpoint protected by SOA agent
Actions	<ol style="list-style-type: none"> <li>1. A web service consumer application places it in XML format</li> <li>2. Wraps it with SOAP headers, placing destination's X.509 certificate in a WS-</li> </ol>

Field	Description
	<p>Security header</p> <ol style="list-style-type: none"> <li>3. Sends the SOAP request to the WSS component</li> <li>4. The web service is protected by the SSOi WS-Security authentication scheme and an authorization policy configured to do the following:</li> <li>5. Obtain the intended recipient's public key certificate from the message headers</li> <li>6. Authenticate the user</li> <li>7. Receive the Status of the Authentication</li> <li>8. Encrypt the required header and message elements.</li> <li>9. SOA agent then forwards the encrypted message to a destination web service.</li> </ol> <p><b>SSOi Responses to XML Digital Signatures</b></p> <ol style="list-style-type: none"> <li>1. A web service consumer application places a digitally signed XML document using its PIV certificate containing (Signature, KeyInfo, KeyName)</li> <li>2. SOA agent intercepts Web service authentication requests and validates the certificate and compare a certificate UPN with AD</li> <li>3. SOA agent forwards message to a destination protected web service</li> </ol>
Main Success Scenarios	User is authenticated, and Application is presented to the user.
Main Failure Scenarios	SOAP fault with authentication failure message returned to client in case of validation of user credential fail



## Template Revision History

Date	Version	Description	Author
January 2015	2.8	Updated to latest Section 508 guidelines and remediated with Common Look Office Tool	Process Management
September 2014	2.7	Adds Enterprise Shared Services terms and requires AERB Compliance Certificate attachment.	Process Management
August 2014	2.6	Signature block update authorized by AERB CR_018934	Process Management
March 2014	2.5	Section 508 repairs to new version approved by AERB Chair approved	Process Management
August 2013	2.3	Replaced the Service Architecture sub-section with new sub-sections for consumed and provided services. Also applied miscellaneous feedback from VA team.	ASD Enterprise Shared Services (ESS) Work Group
June 2013	1.3	Upgraded to MS Office 2007-2010 format	Process Management
June 2013	1.2	Address inconsistencies in Section 3, Conceptual Design, Correct headings	Process Management
March 2013	1.1	Formatted to documentation standards and edited for Section 508 conformance	Process Management
January 2013	1.0	Initial Document	PMAS Business Office

---

See TOGAF® 9.1, Part III: ADM Guidelines & Techniques, Gap Analysis on TOGAF website at <http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap27.html>