Department of Veterans Affairs

Identity and Access Management (IAM)

**Access Services Solution 2.0 Increment 7 (i7) -**

Compliance Audit and Reporting (CAR)

System Design Document (SDD)



**June 2016**

Version 2.7

Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| 6/10/2016 | 2.7 | Updated documentation to reflect removal of IP and AMS from CAR reporting requirements. Modified “VA CSP” and “CSP” system names to “VA Logon”. Updated Sections 3.3.3, 4.1, and 4.2 to reflect architecture changes. | By Light |
| 2/24/2016 | 2.6.0 | Final AcS 2.0 i6 version | AcS Tech Lead |
| 12/07/2015 | 2.5.3 | Updated section 6.2.2.2 per anomaly log | Insignia |
| 11/23/2015 | 2.5.2 | Updated per anomaly log | Insignia |
| 10/28/2015 | 2.5.1 | Updated for Increment 6 and new ProPath template | Insignia |
| 6/28/2015 | 0.5 | Completed a quality review | xxxxx Engility |
| 5/11/2015 | 0.4 | Finalized | Insignia Technology Services |
| 5/01/2015 | 0.3 | Updated per re-opened anomalies | Insignia Technology Services |
| 4/17/2015 | 0.2 | Updated per anomalies | Insignia Technology Services |
| 3/25/2015 | 0.1 | Updated to new ProPath template and applied CAR specific details | xxxx  xxxx  Insignia Technology Services |

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.

When to Complete Each Section of the SDD

| Section | Completed On or Before PMAS Phase | Rationale |
| --- | --- | --- |
| 1 – Introduction | MS 0 Review; updated thereafter | Conceptual design should inform evaluation of investments |
| 2 - Background | MS 0 Review; updated thereafter | Conceptual design should inform evaluation of investments |
| 3 – Conceptual Design | MS 0 Review; updated thereafter | Conceptual design should inform evaluation of investments |
| 4 – System Architecture | MS 0 Review; updated thereafter | Conceptual design should inform evaluation of investments |
| 5 – Data Design | MS 1 Review; updated thereafter | Design details should be elaborated upon during PMAS Planning phase and prior to development |
| 6 – Detailed Design | MS 1 Review; updated thereafter | Design details should be elaborated upon during PMAS Planning phase and prior to development |
| 7 – External System Interface Design | MS 1 Review; updated thereafter | Design details should be elaborated upon during PMAS Planning phase and prior to development |
| 8 – Human Machine Interfaces | MS 1 Review; updated thereafter | Design details should be elaborated upon during PMAS Planning phase and prior to development |
| Attachments | MS 1 Review; updated thereafter | Design details should be elaborated upon during PMAS Planning phase and prior to development |

A product’s system design should be defined conceptually prior to the allocation of personnel and resources that occur at project initiation. This gives the enterprise an opportunity to evaluate IT investments before project teams are stood up and funding is allocated. Sections 1- 4 which discuss the high level design should be completed prior to MS 0. All sections should be completed and updated before MS 1. Projects will need to address all SDD approval constraints prior to the MS 2 review. In addition, the SDD should reflect the as-built product going into the MS 2 review.

Table of Contents

[1. Introduction 1](#_Toc453320877)

[1.1. Scope 1](#_Toc453320878)

[1.2. User Profiles 1](#_Toc453320879)

[2. Background 1](#_Toc453320880)

[2.1. Overview of the System 1](#_Toc453320881)

[2.2. Overview of the Business Process 2](#_Toc453320882)

[2.3. Overview of the Significant Requirements 3](#_Toc453320883)

[3. Conceptual Design 3](#_Toc453320884)

[3.1. Conceptual Application Design 3](#_Toc453320885)

[3.1.1. Application Context 3](#_Toc453320886)

[3.1.2. High-Level Application Design 5](#_Toc453320887)

[3.1.3. Application Locations 8](#_Toc453320888)

[3.2. Conceptual Data Design 8](#_Toc453320889)

[3.2.1. Project Conceptual Data Model 9](#_Toc453320890)

[3.2.2. Database Information 9](#_Toc453320891)

[3.2.3. User Interface Data Mapping 10](#_Toc453320892)

[3.2.3.1. CAR Screen Interface 10](#_Toc453320893)

[3.2.3.2. Report Interface 11](#_Toc453320894)

[3.2.3.3. Unmapped Data Element 11](#_Toc453320895)

[3.3. Conceptual Infrastructure Design 11](#_Toc453320896)

[3.3.1. System Criticality and High Availability 13](#_Toc453320897)

[3.3.2. Special Technology 13](#_Toc453320898)

[3.3.3. Technology Locations 13](#_Toc453320899)

[3.3.4. Conceptual Infrastructure Diagram 15](#_Toc453320900)

[3.3.4.1. Location of Environments and External Interfaces 16](#_Toc453320901)

[3.3.4.2. Conceptual Production String Diagram 16](#_Toc453320902)

[4. System Architecture 17](#_Toc453320903)

[4.1. Hardware Architecture 18](#_Toc453320904)

[4.2. Software Architecture 20](#_Toc453320905)

[4.3. Network Architecture 22](#_Toc453320906)

[4.4. Service Oriented Architecture/ESS 22](#_Toc453320907)

[4.5. Enterprise Architecture 22](#_Toc453320908)

[5. Data Design 22](#_Toc453320909)

[5.1. DBMS Files 23](#_Toc453320910)

[5.2. Non-DBMS Files 23](#_Toc453320911)

[5.3. Data View 23](#_Toc453320912)

[6. Detailed Design 23](#_Toc453320913)

[6.1. Hardware Detailed Design 24](#_Toc453320914)

[6.2. Software Detailed Design 24](#_Toc453320915)

[6.2.1. Conceptual Design 24](#_Toc453320916)

[6.2.1.1. Product Perspective 24](#_Toc453320917)

[6.2.1.1.1. User Interfaces 24](#_Toc453320918)

[6.2.1.1.2. Hardware Interfaces 24](#_Toc453320919)

[6.2.1.1.3. Software Interfaces 25](#_Toc453320920)

[6.2.1.1.4. Communications Interfaces 25](#_Toc453320921)

[6.2.1.1.5. Memory Constraints 27](#_Toc453320922)

[6.2.1.1.6. Special Operations 27](#_Toc453320923)

[6.2.1.2. Product Features 27](#_Toc453320924)

[6.2.1.3. User Characteristics 28](#_Toc453320925)

[6.2.1.4. Dependencies and Constraints 28](#_Toc453320926)

[6.2.1.4.1. Design Assumptions 28](#_Toc453320927)

[6.2.1.4.2. Design Constraints 28](#_Toc453320928)

[6.2.1.4.3. Design Trade-offs 29](#_Toc453320929)

[6.2.2. Specific Requirements 29](#_Toc453320930)

[6.2.2.1. Database Repository 29](#_Toc453320931)

[6.2.2.2. System Features 29](#_Toc453320932)

[6.2.2.2.1. Log Rotation 29](#_Toc453320933)

[6.2.2.2.2. Suppression Rules/Filters 30](#_Toc453320934)

[6.2.2.3. Design Element Tables 30](#_Toc453320935)

[6.2.2.3.1. Routines (Entry Points) 30](#_Toc453320936)

[6.2.2.3.2. Templates 31](#_Toc453320937)

[6.2.2.3.3. Bulletins 31](#_Toc453320938)

[6.2.2.3.4. Data Entries Affected by the Design 31](#_Toc453320939)

[6.2.2.3.5. Unique Record(s) 31](#_Toc453320940)

[6.2.2.3.6. File or Global Size Changes 31](#_Toc453320941)

[6.2.2.3.7. Mail Groups 31](#_Toc453320942)

[6.2.2.3.8. Security Keys 31](#_Toc453320943)

[6.2.2.3.9. Options 31](#_Toc453320944)

[6.2.2.3.10. Protocols 31](#_Toc453320945)

[6.2.2.3.11. Remote Procedure Call (RPC) 31](#_Toc453320946)

[6.2.2.3.12. Constants Defined in Interface 31](#_Toc453320947)

[6.2.2.3.13. Variables Defined in Interface 31](#_Toc453320948)

[6.2.2.3.14. Types Defined in Interface 31](#_Toc453320949)

[6.2.2.3.15. GUI 31](#_Toc453320950)

[6.2.2.3.16. GUI Classes 32](#_Toc453320951)

[6.2.2.3.17. Current Form 32](#_Toc453320952)

[6.2.2.3.18. Modified Form 32](#_Toc453320953)

[6.2.2.3.19. Components on Form 32](#_Toc453320954)

[6.2.2.3.20. Events 32](#_Toc453320955)

[6.2.2.3.21. Methods 32](#_Toc453320956)

[6.2.2.3.22. Special References 32](#_Toc453320957)

[6.2.2.3.23. Class Events 32](#_Toc453320958)

[6.2.2.3.24. Class Methods 32](#_Toc453320959)

[6.2.2.3.25. Class Properties 32](#_Toc453320960)

[6.2.2.3.26. Uses Clause 32](#_Toc453320961)

[6.2.2.3.27. Forms 32](#_Toc453320962)

[6.2.2.3.28. Functions 32](#_Toc453320963)

[6.2.2.3.29. Dialog 32](#_Toc453320964)

[6.2.2.3.30. Help Frame 33](#_Toc453320965)

[6.2.2.3.31. HL7 Application Parameter 33](#_Toc453320966)

[6.2.2.3.32. HL7 Logical Link 33](#_Toc453320967)

[6.2.2.3.33. COTS Interface 33](#_Toc453320968)

[6.3. Network Detailed Design 33](#_Toc453320969)

[6.4. Security and Privacy 33](#_Toc453320970)

[6.4.1. Security 33](#_Toc453320971)

[6.4.2. Privacy 34](#_Toc453320972)

[6.4.2.1. CAR 34](#_Toc453320973)

[6.4.2.2. Confidentiality of Sensitive Information 35](#_Toc453320974)

[6.4.2.3. Privacy of Personal Information 35](#_Toc453320975)

[6.4.2.4. Process Integrity 35](#_Toc453320976)

[6.5. Service Oriented Architecture/ESS Detailed Design 35](#_Toc453320977)

[6.5.1. Service Description for CAR 35](#_Toc453320978)

[6.5.2. Service Design for CAR 35](#_Toc453320979)

[6.5.2.1. Introduction 35](#_Toc453320980)

[6.5.2.1.1. Purpose and Scope of Service 35](#_Toc453320981)

[6.5.2.1.2. Links to Other Documents 35](#_Toc453320982)

[6.5.2.2. Service Details 35](#_Toc453320983)

[6.5.2.2.1. Service Identification 35](#_Toc453320984)

[6.5.2.2.2. Service Versions 36](#_Toc453320985)

[6.5.2.2.3. Summary of Design and Platform Details 36](#_Toc453320986)

[6.5.2.2.3.1. SOA Pattern(s) Implemented 36](#_Toc453320987)

[6.5.2.2.3.2. COTS Platform vendor names and versions for hosting platform 36](#_Toc453320988)

[6.5.2.3. Dependencies 36](#_Toc453320989)

[6.5.2.4. Service Design Details 36](#_Toc453320990)

[6.5.2.4.1. Interface Technical Specs 36](#_Toc453320991)

[6.5.2.4.1.1. Service Invocation Type 36](#_Toc453320992)

[6.5.2.4.1.2. Service Interface Type 36](#_Toc453320993)

[6.5.2.4.1.3. Service Name 36](#_Toc453320994)

[6.5.2.4.1.4. Interface 36](#_Toc453320995)

[6.5.2.4.1.5. End Points 36](#_Toc453320996)

[6.5.2.4.1.6. Operations or Methods 36](#_Toc453320997)

[6.5.2.4.1.7. Message Schemas 36](#_Toc453320998)

[6.5.2.4.2. Information Model 36](#_Toc453320999)

[6.5.2.4.2.1. Class Diagram and Description of Entities Involved 37](#_Toc453321000)

[6.5.2.4.2.2. Mappings from ELDM to Standards Based Schemas 37](#_Toc453321001)

[6.5.2.4.3. Behavior Model (AKA Use Case Realization) 37](#_Toc453321002)

[6.5.2.4.3.1. Use Cases (Use Case Model) 37](#_Toc453321003)

[6.5.2.4.3.2. Interaction Diagrams 37](#_Toc453321004)

[6.5.2.5. Gap Analysis 37](#_Toc453321005)

[6.5.2.5.1. Variances from Enterprise Target Architecture 37](#_Toc453321006)

[6.5.2.5.2. Variances from SLDs 37](#_Toc453321007)

[6.5.2.5.3. Variances from Standards and Policies 37](#_Toc453321008)

[6.5.2.5.4. Justification for Exceptions and Mitigation 37](#_Toc453321009)

[7. External System Interface Design 37](#_Toc453321010)

[7.1. Interface Architecture 37](#_Toc453321011)

[7.2. Interface Detailed Design 39](#_Toc453321012)

[8. Human-Machine Interface 39](#_Toc453321013)

[8.1. Interface Design Rules 39](#_Toc453321014)

[8.2. Inputs 40](#_Toc453321015)

[8.3. Outputs 40](#_Toc453321016)

[8.4. Navigation Hierarchy 40](#_Toc453321017)

[8.4.1. Screen Shots 40](#_Toc453321018)

[Attachment A – Approval Signatures 41](#_Toc453321019)

[A. SEDR Additional Information 43](#_Toc453321020)

[A.1. Identification of Technology and Standards 43](#_Toc453321021)

[A.2. Constraining Policies, Directives and Procedures 43](#_Toc453321022)

[A.3. RTM 45](#_Toc453321023)

[A.4. Packaging and Installation 45](#_Toc453321024)

[A.5. Design Metrics 45](#_Toc453321025)

[A.6. Acronym List and Glossary 45](#_Toc453321026)

[A.7. Required Technical Documents 45](#_Toc453321027)

[A.8. Attach Documents 45](#_Toc453321028)

List of Figures

[Figure 1: CAR Context Diagram 4](#_Toc453321029)

[Figure 2: CAR Application Design 6](#_Toc453321030)

[Figure 3: Conceptual Infrastructure Design 12](#_Toc453321031)

[Figure 4: Conceptual Networks and Environments 16](#_Toc453321032)

[Figure 5: Logical Network String Diagram 17](#_Toc453321033)

[Figure 6: CAR System Architecture 18](#_Toc453321034)

[Figure 7: Hardware Architecture 19](#_Toc453321035)

[Figure 8: Software Architecture 21](#_Toc453321036)

[Figure 9: Data Design 23](#_Toc453321037)

[Figure 10: Log File Mask 30](#_Toc453321038)

[Figure 11: Suppression Rules/Filters 30](#_Toc453321039)

List of Tables

[Table 1: Business Processes 2](#_Toc453323815)

[Table 2: Business Process 3](#_Toc453323816)

[Table 3: Significant Requirement Documents 3](#_Toc453323817)

[Table 4: (Grouping): Application Context Description Interfaces Internal to OIT 5](#_Toc453323818)

[Table 5: Activities in the High-Level Application Design 8](#_Toc453323819)

[Table 6: Application Locations 9](#_Toc453323820)

[Table 7: Application Users 9](#_Toc453323821)

[Table 8: User Activity Reporting Module Attributes 10](#_Toc453323822)

[Table 9: CAR Components 19](#_Toc453323823)

[Table 10: VMs and Applian ces 20](#_Toc453323824)

[Table 11: Compliance Audit and Reporting (CAR) 23](#_Toc453323825)

[Table 12: CAR Activity Architecture 26](#_Toc453323826)

[Table 13: Port Communications and Protocols 27](#_Toc453323827)

[Table 14: CAR Products 28](#_Toc453323828)

[Table 15: Assumptions 29](#_Toc453323829)

[Table 16: Constraints 30](#_Toc453323830)

[Table 17: Data Points and Security 35](#_Toc453323831)

[Table 18: CAR Server 39](#_Toc453323832)

[Table 19: System Identification 44](#_Toc453323833)

[Table 20: Policies, Directives, and Procedures 44](#_Toc453323834)

# Introduction

Compliance Audit and Reporting (CAR) is a commercial off-the-shelf (COTS) product providing centralized monitoring, alerting, and auditing, as well as compliance reports in association with Access Services (AcS) 2.0. CAR establishes a common compliance auditing framework, providing protections and security for audit data, as required.

## Scope

This SDD focuses on the technical system design to provide the foundation for the CAR Service. It provides an overview of the core capabilities, architecture, and design. It does not include default COTS product design nor does it include out of the box (OOTB) data definitions, tables, or models; except where the design alters such elements and components. Refer to the [Identity and Access Management (IAM) Access Services (AcS) Business Requirements Document (BRD) in the Technical Service Project Repository (TSPR)](http://xxxxxxxx/warboard/anotebk.asp?proj=1803&Type=Active#Documentation).

**NOTE:** There are no Enterprise Requirements for CAR in Increment 7 (i7).

## User Profiles

The user community for the CAR activities consists of internal users including VA employees, contractors and affiliates. All users identified above are internal users. At this point none of the CAR reports and functionalities are external or Veteran facing.

# Background

IAM AcS 2.0 offers several services that 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 identity federation capabilities for external users. It also offers authorization capabilities, both coarse and fine-grained, for application access. AcS Provisioning enables capture of user access criteria for access to VA applications via self-service or sponsor initiated workflows, and is the single point of control for user life cycle management. AcS offers Auditing, Reporting and monitoring capabilities for compliance purposes. It also offers a Digital signature capability that enables external users to digitally sign forms that require a high level of verification that the user signing the document is a legitimate user.

## Overview of the System

The purpose of the AcS Development Support task is to design, develop, implement, integrate, operationalize, and sustain an enterprise-wide AcS 2.0 for Enterprise Shared Services (ESS). In order to coordinate AcS across several ESS 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.

The CAR system design is based on a Service Oriented Architecture (SOA) approach. The service architecture uses accepted COTS products for each 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.

The CAR Service provides the ability to proactively monitor and mitigate various potential compliance infractions and incidents. It provides a common compliance auditing and reporting framework to be leveraged throughout the VA. It sets the foundation to support adherence to applicable government policies and regulations and provides the capability to monitor IAM services to produce reports and generate alerts triggered by events or breach of predetermined thresholds. The focus of this IAM activity is to consolidate and to lessen the occurrence of redundancies, where possible, while reducing the administrative burden of compliance reporting for the VA.

## Overview of the Business Process

CAR, based on the CA User Activity Reporting Module (UARM) product, provides centralized monitoring, alerting, and auditing as well as compliance reporting in association with the AcS 2.0. It establishes a compliance auditing framework that will provide the protections and security for the audit data as required. The CAR Service is integrated with Specialized Access Control (SAC), Credential Service Provider (CSP) (VA Logon), Provisioning (Prov), e-Signature (e-Sig) and Single Sign-On external (SSOe) activities as well as Enrollment Services (ES). The CAR service provides the unique ability to proactively monitor, mitigate, and recover from potential compliance infractions and incidents, a common compliance auditing framework to be leveraged throughout the VA to provide the foundation for adherence within applicable government policy and regulation and the capability to monitor AcS services to produce reports and generate alerts triggered by events or breach of predetermined thresholds.

**NOTE:** CAR provides reports on SSO audit data; however, CAR cannot be integrated with SSO for login purposes.

Table : Business Processes

| Business Process ID | Business Process Name | Owner | Description |
| --- | --- | --- | --- |
| 1 | Manage Auditable Events | VA IAM AuthR/CAR PM | Provides ability for Privileged Users to define and configure auditable events. |
| 2 | Generate Standard Reports | VA IAM AuthR/CAR PM | Provides ability for Privileged Users to generate standard reports. |
| 3 | Define and Generate Ad-Hoc Reports | VA IAM AuthR/CAR PM | Provides ability for Privileged Users to define and generate ad-hoc reports using data elements available to CAR. |
| 4 | Define and Generate Alerts | VA IAM AuthR/CAR PM | Provides ability for Privileged Users to define parameters which will initiate alerts. |
| 5 | Manage User Access | VA IAM AuthR/CAR PM | Provides ability for CAR privileged user to manage user access to reporting features. |

Refer to the [VA AcS 2.0 Requirements Specification Document (RSD)](http://xxxxxx/warboard/anotebk.asp?proj=1803&Type=Active#Documentation), use case, and Requirements Traceability Matrix (RTM) documents for the business process flows.

Refer to the [Use Case Model](http://xxxxx/warboard/anotebk.asp?proj=1653&Type=Active) for CAR for applicable diagrams to support this section and the following Use Cases:

Table : Business Process

| Business Process ID | Business Process Name | Type | Owner | Description |
| --- | --- | --- | --- | --- |
| 1 | [VA 2.0 Increment 2 Use Case Model Document](http://xxxxxxx/warboard/ProjectDocs/Access_Services/VA_AcS_2.0_i2_Solution_UC_Model_SAC.pdf) | Use Cases | PD OIT | Use Case Model Document |
| 2 | [VA i4 Use Case Model](http://txxxxxx/warboard/ProjectDocs/Access_Services/AcS_i4_Use_Case_Model.pdf) | Use Cases | PD OIT | i4 Use cases |
| 3 | [CAR Use Case Model](http://xxxxxxxx/warboard/ProjectDocs/Access_Services/VA_IAM_CAR_Use_Case_Model.pdf) | Use Cases | PD OIT | CAR Use cases |

## Overview of the Significant Requirements

There are no implementation requirements for CAR in i7.

Table : Significant Requirement Documents

| Document | Content Description | Location |
| --- | --- | --- |
| AcS Business Requirements Document (BRD) | Identifies the business needs of the customer/business owner; stakeholders; and an overview of the request requirements, constraints, and Information Technology (IT) options considered | [Technical Service Project Repository (TSPR)](ttp://xxxxxxx/warboard/anotebk.asp?proj=1803&Type=Active) |
| i7AcS Requirements Specification Document (RSD) | Summarizes the business and functional requirements that are required for the development and implementation of AcS 2.0 Increment 7. | [Technical Service Project Repository (TSPR)](http://xxxxxxx/warboard/ProjectDocs/Access_Services_Phase_2/AcS_2.0_i7_RSD.pdf) |
| CAR Functional Requirements | Overview of significant functional requirements | Overarching AcS Requirements |
| CAR Workload/Performance Requirements | Overview of the functional workload/performance requirements | Overarching AcS Requirements |
| CAR Operational Requirements | Overview of operational requirements | Overarching AcS Requirements |
| CAR pivotal technical requirements | Overview of pivotal technical requirements | Overarching AcS Requirements |
| CAR Security and Privacy Requirements | Overview of the security and privacy requirements | Overarching AcS Requirements |
| CAR Availability Requirements | Overview of system criticality and high availability requirements | Overarching AcS Requirements |
| CAR Single Sign On Requirements | Overview of SSO requirements | Overarching AcS Requirements |
| CAR Enterprise Portals Use | Overview of use of enterprise portals | Overarching AcS Requirements |
| CAR Special Device Requirements | Overview of special device requirements | Overarching AcS Requirements |

# Conceptual Design

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

* Conceptual Application Design
* Conceptual Data Design
* Conceptual Infrastructure Design

## Conceptual Application Design

### Application Context

Compliance Audit and Reporting (CAR) provides the capability to monitor AcS activities to produce reports and generate alerts triggered by events or breach of predetermined event thresholds. Enabling an enterprise CAR service provides VA a common compliance auditing framework enabling the foundation for adherence within applicable government policy and regulation. The CAR service provides Compliance Reporting and Policy Violation Alerting.

The primary actors interacting with the CAR Service are the following:

* Administrator (Privileged User): Responsible for control and maintenance of CAR and to generate reports
* Report User: Responsible for generating reports
* Data Supplier: Responsible for providing the endpoint data needed for reporting

The CAR Service is based on the User Activity Reporting Module (UARM) COTS provided by CA Technologies (CA). CAR integrates with the following services:

* SAC
* CSP (VA Logon)
* SSOi
* SSOe/VA Authentication Federation Infrastructure (VAAFI)
* eSig
* Prov (including Virtual Directory Service [VDS] and SailPoint)
* Enrollment Services (ES)

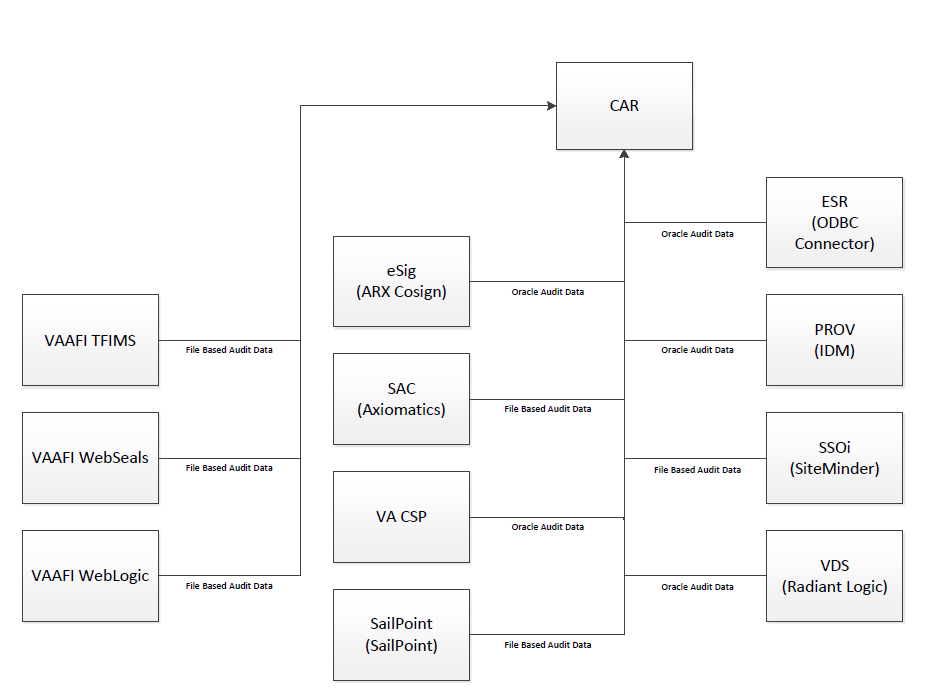


Figure : CAR Context Diagram

Table : (Grouping): Application Context Description Interfaces Internal to OIT

| ID | Name | Related Object | Input Messages | Output Messages | External Party |
| --- | --- | --- | --- | --- | --- |
| 1 | eSig | Digital Signatures | SQL queries | ODBC Response | ODBC interface is queried by CAR agent connector to collect the audit logs from the eSig Audit Source |
| 2 | VDS | LDAP Queries, DSML queries | SQL queries | ODBC Response | ODBC interface is queried by CAR agent connector to collect to the VDS audit source |
| 3 | CSP (VA Logon) | SSOi Service | SQL queries | ODBC Response | ODBC interface is queried by CAR agent connector to collect audit information from Oracle DB. |
| 4 | Provisioning | SSOi Service | SQL queries | ODBC Response | ODBC interface is queried by CAR agent connector to collect the CA IDM audit information |
| 5 | SAC | SSOi Service | File Reader Queries | File Reader Response | File base Reader is used by CAR agent to collect the SAC text based audit logs |
| 6 | SSOi | SSOi Service | File Reader Queries | File Reader Response | File base Reader is used by CAR agent to collect the SSOi text based audit logs |
| 7 | SailPoint | SSOi Service | File Reader Queries | File Reader Response | File base Reader is used by CAR agent to collect the SailPoint text based audit logs |
| 8 | SSOe | SSOe Service | File Reader Queries | File Reader Response | File base Reader is used by CAR agent to collect the SSOe text based audit logs |
| 9 | VAAFI | VAAFI (TFIM, WebSeal and Weblogic) | File Reader Queries | File Reader Response | File base Reader is used by CAR agent to collect the VAAFI text based audit logs |
| 10 | ES | ES Service | SQL queries | ODBC Response | ODBC interface is queried by CAR agent connector to collect the CA IDM audit information |

### High-Level Application Design

The CAR Service requires agents to be deployed on integrated application servers to consume audit data that is then transmitted to the CAR servers for collection and reporting. A high-level application design for the CAR, and the major AcS activities and/or relationships with VA applications, are shown in Figure 2.

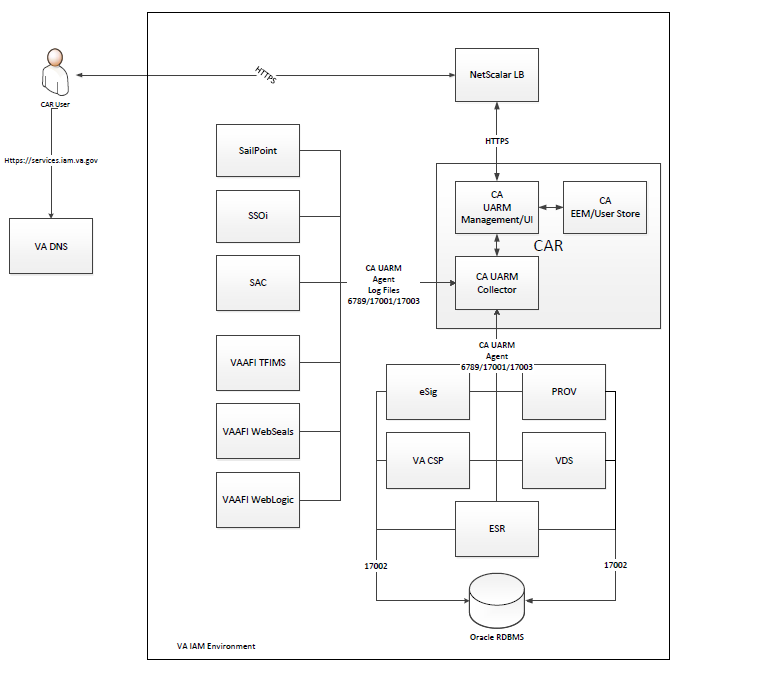


Figure : CAR Application Design

All the internal interactions for CAR are handled by the UARM COTS product. As such, they are obscured within the application.

The following table provides 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 : Activities in the High-Level Application Design

| ID | Name | Description | Service or Legacy Code | External Interface Name | Internal Interface Name |
| --- | --- | --- | --- | --- | --- |
| 1 | CSP (VA Logon) | CSP (VA Logon) provides external user’s credentials to VA applications that are not eligible for another VA approved credential. | Service | Self Service and Registration | VAAFI, CAR |
| 2 | eSig | eSig provides the ability to sign documents electronically. | Service | N/A | CAR |
| 3 | 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 |
| 4 | Prov | Prov 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 |
| 5 | 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, CSP (VA Logon), Provisioning, SAC |
| 6 | CAR | CAR provides the ability to proactively monitor, mitigate, and recover from potential compliance infractions and incidents. | Service | N/A | SSOi, Provisioning, CSP (VA Logon), eSig, SAC |

### Application Locations

Table : Application Locations

| Application Component | Description | Prod / PreProd Hosting Facility | Type | SQA |
| --- | --- | --- | --- | --- |
| CA UARM | User Audit and Reporting Module. | Terremark Culpeper, VA (Primary)  Terremark Miami, FL (Disaster Recovery) | N/A | AITC |
| Management/Reporting Server | Managing the UARM interface to the user | Terremark Culpeper, VA (Primary)  Terremark Miami, FL (Disaster Recovery) | Presentation Service | AITC |
| Collection Server | Repository for normalized UARM logs | Terremark Culpeper, VA (Primary)  Terremark Miami, FL (Disaster Recovery) | Data Layer | AITC |
| Oracle | Repository for CSP (VA Logon) audit events | Terremark Culpeper, VA (Primary)  Terremark Miami, FL (Disaster Recovery) | Data Layer | AITC |
| DataPower Audit Logs | The output consisting of the events for DataPower exposed services | Terremark Culpeper, VA (Primary)  Terremark Miami, FL (Disaster Recovery) | Data Layer | AITC |

Table : Application Users

| Application Component | Location | User |
| --- | --- | --- |
| CAR | Performs administrative functions including management of UARM reports dashboard, generation of reports, and creating other users in UARM | CAR Administrator |
| CAR | Read Only Users | Auditor |
| CAR | Runs reports and tracks audit records to verify continual system conformance with security and policy with some edit capabilities | Analyst |

## Conceptual Data Design

The CAR data model aligns the necessary data attributes to meet the functional requirements, as well as standard, FIPS compliant technologies to secure sensitive data types. The CAR data model design leverages the default schema used by consumers such as CSP (VA Logon), as well as flat file structure such as SAC log files, and is imported into the default schema for UARM product using the default normalization. The default schema is internal to the COTS and is not visible to the end user.

### Project Conceptual Data Model

CAR is based on a closed system (CA UARM) that does not interact with any separate repositories for its functions and therefore follows data model which is provided out of the box with the product. CAR is COTS-based; please refer to the CA UARM documentation for the applicable data model.

### Database Information

The CAR Service stores the event data in the UARM logs. The UARM schema designed to meet the needs of the CAR Service, and associated functional and technical requirements, is shown in the table below.

**NOTE:** The end user may choose the attributes selected for display within the report. Selecting a particular event will display the following attributes:

Table : User Activity Reporting Module Attributes

| Index | Attribute | Attribute Description | Data Format |
| --- | --- | --- | --- |
| 1 | source\_address | The source address for the event | Any Text |
| 2 | source\_hostdomainname | The host domain name for the source | Any Text |
| 3 | source\_hostname | The hostname for the source | Any Text |
| 4 | source\_objectname | The object name for the event | Any Text |
| 5 | source\_processname | The process name | Any Text |
| 6 | dest\_hostname | The destination hostname | Any Text |
| 7 | event\_action | Action | Any Text |
| 8 | event\_category | Category of the event | Any Text |
| 9 | event\_class | Class of the event | Any Text |
| 10 | event\_count | Count for the event | Any Text |
| 11 | event\_datetime | Date and time for the event | Any Text |
| 12 | event\_day\_datetime | The day | Any Text |
| 13 | event\_hour\_datetime | Hour | Any Text |
| 14 | event\_logname | The logname | Any Text |
| 15 | event\_minute\_datetime | Minute | Any Text |
| 1 | event\_month\_datetime | Month | Any Text |
| 17 | event\_quarterhour\_datetime | Quarterhour | Any Text |
| 18 | event\_sequence | Sequence | Any Text |
| 19 | event\_summarized | Summary | Any Text |
| 20 | event\_time\_gmt | Time | Any Text |
| 21 | event\_time\_hour | Hour | Any Text |
| 22 | event\_time\_minute | Minute | Any Text |
| 23 | event\_time\_month | Month | Any Text |
| 24 | event\_time\_monthday | Day of the Month | Any Text |
| 25 | event\_time\_weekday | Weekday info | Any Text |
| 26 | event\_time\_year | Year of the event | Any Text |
| 27 | event\_year\_datetime | Datetime of the event | Any Text |
| 28 | ideal\_model | The model | Any Text |
| 29 | event\_result | The event result string | Any Text |
| 30 | result\_string | The detailed result string | Any Text |
| 31 | event\_source\_address | Event source address | Any Text |
| 32 | event\_source\_hostdomainname | The host domain name for the source | Any Text |
| 33 | event\_source\_hostname | The hostname for the source | Any Text |
| 34 | agent\_address | The address of the agent | Any Text |
| 35 | agent\_connector\_name | The connector name for the agent | Any Text |
| 37 | agent\_group | The group of the agent | Any Text |
| 38 | agent\_hostdomainname | The host domain name for the agent | Any Text |
| 39 | agent\_hostname | The hostname of the agent | Any Text |
| 40 | agent\_name | The name of the agent | Any Text |
| 41 | agent\_version | The version number for the agent | Any Text |
| 42 | raw\_event | The raw event string | Any Text |
| 43 | receiver\_hostaddress | The hostaddress for the receiver | Any Text |
| 44 | receiver\_hostname | The hostname of the receiver | Any Text |
| 45 | receiver\_name | The name of the receiver | Any Text |
| 46 | receiver\_port | The port for the receiver | Any Text |
| 47 | receiver\_time\_gmt | The GMT time for the receiver | Any Text |
| 48 | receiver\_timezone | The timezone for the receiver | Any Text |
| 49 | receiver\_version | The version number for the receiver | Any Text |

**NOTE:** Any text is nomenclature specific to UARM. Any text supports dynamic length alphanumeric characters.

### User Interface Data Mapping

This section describes and defines the data that will be available for users of the CAR Service. Out-of-the-box screens are not shown.

#### CAR Screen Interface

N/A

#### Report Interface

N/A

#### Unmapped Data Element

N/A

## Conceptual Infrastructure Design

CAR is the central reporting authority for the IAM service for the VA. The following list describes the event sources for various IAM services.

* Provisioning provides the information through the CA Identity Manager and SiteMinder connections. These events are captured in the Oracle instance and, in turn, that information is transferred into UARM logs.
* CSP (VA Logon) provides the information in the Oracle instance, which then transfers events to the UARM logs.
* The IBM DataPower is the underlying backbone for numerous IAM services (Ex: SAC), and the CAR system pulls in the data events in form of log files from DataPower and parses them appropriately
* SSOe, SSOi, and SailPoint provide audit data from their component servers and the CAR system pulls in the data events in form of log files and parses them appropriately
* VDS, ESR, and eSig store audit data within the Oracle instance which is then collected in the UARM logs

Figure 3 presents the conceptual representation from CAR.



Figure : Conceptual Infrastructure Design

The Internal Administrator will access the load balancer admin URL to view the reports. The load balancer will route the request to the report servers where the user may view different compliance reports. The report servers exist as three separate virtual machines that each run the UARM product, which is a virtual appliance based on a proprietary version of CENTOS Linux. The three servers are linked in a federated mesh that keeps them all synched with one another.

In conjunction with the integration contained herein, the IAM components (Identity Manager, SiteMinder, CA SSO, SailPoint, SSOe, and SAC) will have UARM agent installed. For components that utilize the Oracle database, the agent will access the related audit store on the Oracle instance. The agent will transfer the logs into the UARM Collector server. The Report server will collect the data from the collector server and display it to the end user.

### System Criticality and High Availability

The VA AcS infrastructure supports critical business systems. The current availability requirement for mission critical systems is 99.99%. The current data center Service Level Agreement with Terremark supports 99.9% availability. Terremark hosts the Production, Pre-Production, and Disaster Recovery (DR) Data Centers in Culpeper, VA, and Miami, FL. Terremark currently does not support an active/active geographic failover and load balancing; thus, failover to the DR site could take between one and eight hours.

**NOTE**: The 99.9% availability does not include scheduled maintenance for patches, software updates, and related activities.

Terremark does not currently support an active/active geographic failover and load balancing thus failover to the DR site could take between one and eight 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 a contingency site that will resume data center operations in the event of a site failure. Load balancing, fault tolerance, backups and archiving, are functions of the hosting facility, Terremark, and the data center operations team. Backups are described more fully in the [Production Operations Manual (POM)](http://xxxxxxxxxx/warboard/anotebk.asp?proj=xxx&Type=Active), 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

The CAR architecture maintains two Collector Servers and one Reporting Server. The Collector Servers are the main actors that collect the data events and are designed to have an instant failover. The agents for the collectors would failover to the appropriate collector, which will reduce the likelihood of data loss in transit.

The reporting servers are designed with a hot and a cold instance. Since the reporting server is not responsible for any data collection, the hot and cold instances addresses HA requirements in that the collector server will be switched to the cold instance in case of a failure.

The VA AcS infrastructure supports critical business systems. The current business requirements for disaster recovery are for the system to ensure 24/7/365 availability such as load balancing, redundancy, backups, continuity, and disaster recovery plans. The Production, Preproduction, and Disaster Recovery (DR) Data Center is hosted by Terremark in Culpeper, Virginia and Miami, Florida. Terremark provides an RPO of 15 minutes and RTO of 12 hours.

### Special Technology

N/A

### Technology Locations

Development Environment (INT) 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

Partner Integration Environment (PINT) AITC – Austin, TX

* This environment is utilized by the Development team for integration testing with IAM partners for service enhancements, integrations with consuming applications, defect resolution, and quality testing
* This is a tightly controlled environment for the AcS developers and IAM partners 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 one second

### Conceptual Infrastructure Diagram

This section depicts the CAR infrastructure. Each component 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.

#### Location of Environments and External Interfaces

Image captures conceptual networks and environments.


Figure : Conceptual Networks and Environments

#### Conceptual Production String Diagram

The following diagram provides a logical view of CAR.



Figure : Logical Network String Diagram

# System Architecture

The following section describes the overall system architecture along with hardware and software architecture view of the CAR Service. This section provides an overview of the various components involved in the solution and their interaction both internally and externally. An abstract operational view of the service is depicted in the conceptual design above. The [Hardware Architecture](#_Hardware_Architecture) and [Software Architecture](#_Software_Architecture) sections elaborate the CAR Service system architecture.

**NOTE:** CA UARM end of life is December 2017. At this time the replacement tool has not been selected. The [CAR Analysis of Alternatives (AoA)](http://xxxxxxxxx/sites/vrm/IAM/IAM%20Documents/Forms/AllItems.aspx?RootFolder=%2Fsites%2Fvrm%2FIAM%2FIAM%20Documents%2FIAM%20Development%20Documentation%2FInsignia%2DAcS%20Development%20Integration%20i5%2F5%2E2%2E3%2E1%20Solution%20Design%2FCAR%20Analysis%20of%20Alternatives%20Doc&InitialTabId=Ribbon%2EDocument&VisibilityContext=WSSTabPersistence) has been provided to the VA to determine the potential options to replace UARM. Current estimates place a decision point for end of FY2016. An open market purchase requisition is in progress to select a suitable replacement product. Replacement product installation and migration of audit reports/logs is planned for increment 8.

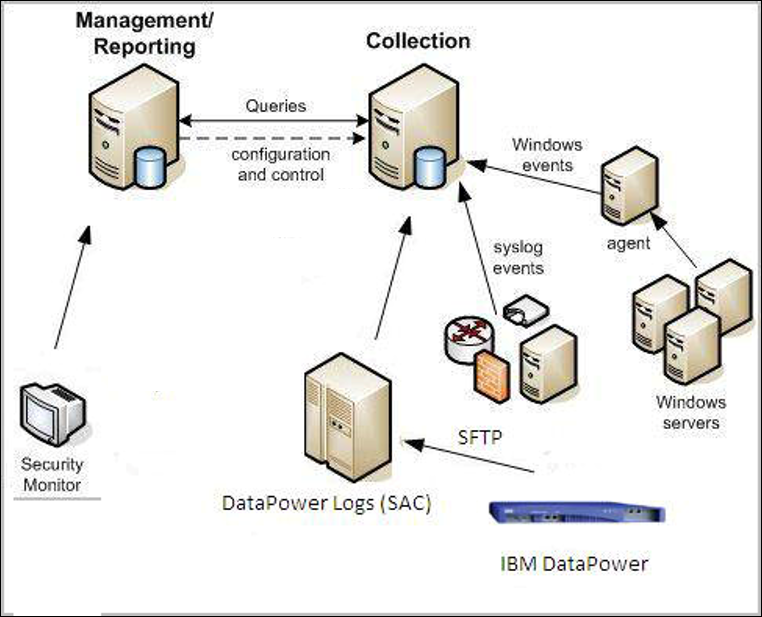


Figure : CAR System Architecture

Each component of the CAR Architecture is described in the table below.

Table : CAR Components

| Component | Definition / Description |
| --- | --- |
| Management Server | The Management server presents the GUI to the end user and aggregates reports from the Collection server. |
| Collection Server | The Collection server is the main engine of the CAR system. It aggregates event information from AcS Services. The Collection Server also contains the UARM logs that store the event information for CAR. |
| Agents | Various agents are deployed across IAM activities and Windows servers to poll the events that need to be recorded. Agent details are documented in Section 3.2.2 |

## Hardware Architecture

The CAR Service architecture is composed of the following components:

* UARM Report and Management Server
* UARM Collector
* UARM Agent and Connector

The following diagram shows the AcS 2.0 hardware architecture.



Figure : Hardware Architecture

The uniform resource locators (URLs) for CAR for production, pre-production and SQA are provided in the table below. The following table provides details on the CAR server such as ports, URLs, protocols hostnames for each application in every environment.

Table : VMs and Applian ces

| Environment | Application | # of VMs | # of Physical Servers | Hostname |
| --- | --- | --- | --- | --- |
| SQA (AITC) | CA UARM (Tomcat) | 4 | N/A | xxxxxxxxxxxxxxx  xxxxxxxxxxxxxxx  xxxxxxxxxxxxx  xxxxxxxxxxxxxx |
| INT (AITC) | CA UARM | 4 | N/A | xxxxxxxxxxx  xxxxxxxxxxx  xxxxxxxxxxx  xxxxxxxxxxx |
| PINT (AITC) | CA UARM | 4 | N/A | xxxxxxxxxxx  xxxxxxxxxxx  xxxxxxxxxxx  xxxxxxxxxxx |
| Pre-Production (Terremark Culpeper, VA) | CA UARM | 3 | N/A | xxxxxxxxxxx  xxxxxxxxxxx  xxxxxxxxxxx |
| Production (Terremark Culpeper, VA | CA UARM | 3 | N/A | xxxxxxxxxxx  xxxxxxxxxxx  . xxxxxxxxxxx  xxxxxxxxxxx |
| DR (Terremark Miami, FL) | CA UARM | 3 | N/A | xxxxxxxxxxx  xxxxxxxxxxx  xxxxxxxxxxx  . xxxxxxxxxxx |

## Software Architecture

The Management and Collection server form the backbone for the CAR system. The following diagram describes the software architecture for CAR users.



Figure : Software Architecture

The end of life of the CAR Service is December 31, 2017. The only changes that can be made are configuration changes on the service. Any requests for enhancements made to the underlying COTS product will be rejected by CA.

The Management server provides the GUI to the user and sends the query to the Collector Server and presents it to the user.

* The CAR Service deploys agents for Provisioning, SailPoint, VDS, SSOi, SSOe, and SAC Services
* The agents are deployed using the UARM administrator console
* There are two agents that are deployed for CSP (VA Logon) on the CSP (VA Logon) server, which also contains the legacy CSP. These agents poll the audit event changes to the Oracle DB. As mentioned later, the agents forward the events to the Collector server. The Collector server are configured for failover, so no data is lost in transition.
* There is one agent that is deployed for Provisioning on the Provisioning server. This agent polls the audit event changes to Identity Manager. As mentioned later, the agents forward the events to the Collector server. The Collector server are configured for failover, so no data is lost in transition.
* There are two agents that are deployed for SSOi on the SSOi servers. These agents consume audit data from the CA SSO log files on those servers. As mentioned later, the agents forward the events to the Collector server. The Collector server are configured for failover, so no data is lost in transition.
* There are two agents that are deployed for SailPoint on the SailPoint servers. These agents consume audit data exported from the SailPoint application running on those servers. As mentioned later, the agents forward the events to the Collector server. The Collector server are configured for failover, so no data is lost in transition.
* There is no agent deployed for VDS as the events are captured in the Oracle instance and are collected by CAR through a connector on the Identity Manager agent
* There are two agents deployed for SAC as the events are captured in log files which is are available on the ASM and PDP servers
* There are agents deployed on each of the VAAFI WebSeal, WebLogic, and TFIM servers to collect audit events from the log files there pertaining to SSOe
* In addition to the aforementioned agents, there is a self-monitoring agent which comes pre-bundled and preconfigured with UARM
* The connectors capture events in the CSP (VA Logon) and store them in the Oracle database
* The CAR Service utilizes a built-in Oracle connector to gather audit data, normalize the audit events, and store them in the UARM collector server
* The DataPower logs are transferred using the SFTP protocol to the UARM server. The Custom file log connector parses the events and stores them in the UARM collector store.
* The Management and Reporting server uses the internal UARM logs to provide the ad-hoc and standard reports/alerts

## Network Architecture

See Figure 2 in Section [3.1.2](#_High-Level_Application_Design) of this document for a depiction of the communication channels between the different AcS components and protocols used.

## Service Oriented Architecture/ESS

CAR is a self-contained system but is not an SOA service because it does not offer a Service interface and thus this section is N/A.

## Enterprise Architecture

The end of life of the CAR Service is December 31, 2017. The only changes that can be made are configuration changes on the service. Any requests for enhancements made to the underlying COTS product will be rejected.

Table : Compliance Audit and Reporting (CAR)

| Products | Abbreviation | Product Version / Release |
| --- | --- | --- |
| CA User Activity Reporting Module | CA UARM | 12.5 SP7 (12.5) |
| CA User Activity Reporting Module Agent | CA UARM Agent | 12.5 SP7 (12.5) |

# Data Design

CA UARM is a COTS solution; refer to the COTS documentation.



Figure : Data Design

## DBMS Files

In order for CAR to collect audit data, an agent binary is installed on the servers of integrated applications with access to the relevant audit data for reporting. A connector is configured on the agent for each event source. The connector configuration specifies the data source and the relevant data from that data source that CAR will collect.

Collected events specified by connectors are sent by the agent to a CA User Activity Reporting Module server for processing and initial storage. The events are stored within a SQL based log store that is part of the CA UARM application. These log stores are part of the COTS product and can only be modified by the application. They are then synched up between all three of the UARM Log Manager servers.

The event log store uses a federated system, with each host server maintaining its own local event log store and the ability to contact other stores in the CAR environment. When you query a server for event information, it can search its own local event log store as well as all others connected through the federation. This arrangement allows for data integrity to be maintained even if an issue develops with one of the servers.

## Non-DBMS Files

Refer to the appropriate ICD for integration related information.

## Data View

N/A

# Detailed Design

This section describes the design for the CAR Service and its activities in detail.

## Hardware Detailed Design

The following table provides the hardware information for CAR. The sizing, network, operating system, and number of virtual machines and detailed in the Server Planning Sheets at the following link:

<http:///sites/vrm/IAM/IAM%20Documents/Forms/AllItems.aspx?RootFolder=%2Fsites%2Fvrm%2FIAM%2FIAM%20Documents%2FIAM%20Development%20Documentation%2FByLight%20%2D%20AcS%20Development%20Integration%20i7%2F5%2E4%2E3%2E1%20Solution%20Design%2FUpdated%20Server%20Planning%20Sheets>

## Software Detailed Design

The CAR Service consolidates monitoring and audit reporting to a single solution for multiple AcS services. The CAR Service is based on the UARM COTS and integrates with the following AcS activities:

* CSP (VA Logon)
* Prov
* SAC
* SSOi
* SSOe
* eSig
* VDS
* SailPoint

### Conceptual Design

#### Product Perspective

CAR is based on a COTS product, and is independent and self-contained.

##### User Interfaces

Refer to COTS documentation for information on user interfaces.

The CA UARM is incompatible with SSOi and instead must rely on its own internal user store. User accounts are configured manually by an administrator and are solely used for accessing CAR. The three predefined roles included within the UARM COTS product are Administrator, Analyst, and Auditor. Although custom roles are created based on the requirements of individual integrations they all conform to the same basic privileges. Administrators have full access to everything in CAR including configuration changes. Analysts only have access to the reports, including the ability to create and modify reports, but they cannot access system configurations. Auditors are users that only have read access on reports.

##### Hardware Interfaces

N/A

##### Software Interfaces

N/A

##### Communications Interfaces

The CAR activity architecture contains agents and server communications where agents would be deployed on the destination systems that invoke a connector that is designed to recognize a specific log pattern and normalize into a common event grammar format that is stored on the UARM collector server.

* **Agent and Server Communications:** Agent collects the normalized events into its queue. The queue manager then sends the normalized events to the UARM collector server using dispatcher service.
* **Connectors:** Current implementation of UARM uses three out-of-box connectors (i.e., CA IdentityMinder, CA SiteMinder, CA SSO and custom connectors for Axiomatics, ARX CoSign, and ESR, VDS, SSOe, and SailPoint)
* **Connector Data Mapping File:** Data mapping file defines global definition and provides the output as the normalized events
* **Connector Parser File:** Parser disassembles the raw events and normalizes this information to common event grammar
* **Oracle Connectors:** The connectors for Oracle audit source use ODBC connections to connect and fetch audit events

Table : CAR Activity Architecture

| Field | Description |
| --- | --- |
| Use Case Name | Process Activity Logs |
| Description | This Use Case describes the process through which CAR will consume and process the data from the audit logs to generate reports. |
| Actors | Agent  Connector  UARM Connector  Oracle RAC/File Log |
| Pre-Conditions | The AcS activities have captured the audit logs and CAR is setup to connect with audit log store. |
| Trigger | The audit logs connector is invoked by the AcS activity agent. |
| Actions | The agent invokes the connector.  The agent makes an ODBC connection to the Oracle RAC to obtain the audit file.  The Oracle RAC returns the raw audit file data.  The connector normalizes the data.  The connector submits the data to the collection queue.  The agent executes the Dispatcher Service and sends the data to the UARM Collector for generation of reports. |
| Sequence Diagram |  |
| Main Success Scenarios | The Management and Reporting server uses the internal UARM logs to provide the ad-hoc and standard reports/alerts. |
| Main Failure Scenarios | No Audit Logs are retrieved to generate reports. |

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 : Port Communications and Protocols

| Application | Network | Port(s) | Reason | Protocol(s) |
| --- | --- | --- | --- | --- |
| CA UARM | Internal | xxxxx | Administration Port for CA UARM | TCP |
| CA UARM | Internal | xxxxx | SSL Port (reverse proxy to administration port 5250) for CA UARM | HTTPS |
| CA UARM | Internal | xxxxx | Syslog port (UDP) for CA UARM server | TCP |
| CA UARM | Internal | xxxxx | Syslog TCP listening port for CA UARM | TCP |
| CA UARM | Internal | xxxxx | Agent command and control listening port | TCP |
| CA UARM | Internal | xxxxx | Communication port for ODBC /JDBC driver | TCP |
| CA UARM | Internal | xxxxx | Audit client communication with port-mapper | TCP |
| CA UARM | Internal | xxxxx | Dispatcher SME listener | TCP |
| CA UARM | Internal | xxxxx | CA Directory LDAP DXadmin port (CA Directory bundled with CA UARM) | TCP |
| CA UARM | Internal | xxxxx | Dispatcher Service in SSL mode for events from Client Connector | TCP |

Refer to the POM for PreProd and Prod PKI Certificates.

##### Memory Constraints

The CA recommended system configuration requires at least 8 gigabytes of RAM per UARM server.

##### Special Operations

N/A

#### Product Features

AcS 2.0 is based on the foundation of CA COTS products. The table below describes the AcS 2.0 products for the CAR Service.

Table : CAR Products

| # | Software | Description |
| --- | --- | --- |
| 1 | CA User Activity Reporting Module (UARM) | CA User Activity Reporting Module is a high-performance log management solution. |

#### User Characteristics

Refer to Section 1.2 for user-related information.

CAR users are created manually by a CAR administrator through the CAR GUI. The permissions associated with a user can be customized based on assigned application group but they generally conform to either auditor, analyst, or administrator access levels. Auditors have read only access to reports. Analysts can view and edit reports. Administrators can also view and edit reports as well as manage users and apply system configuration changes.

#### Dependencies and Constraints

This section describes the assumptions and constraints that impact the design of the CAR Service. Refer to Section A.2 for additional information.

##### Design Assumptions

The end of life of the CAR Service is December 31, 2017. The only changes that can be made are configuration changes on the service. Any requests for enhancements made to the underlying COTS product will be rejected. This is due to the vendor only addressing bug fixes.

Table : Assumptions

| Assumption |
| --- |
| UARM is currently nearing its end of life. Any future enhancements of the product will be limited. |

##### Design Constraints

This document is developed under the schedule and cost defined in the contract for VA CAR 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.

AcS Service – CAR

* **CA User Activity Reporting Module:** Version 12.5.1 or greater must be used and be configured and operated in FIPS Mode. FIPS Mode is required to provide FIPS-certified security algorithms for event transport and other communications between the CA User Activity Reporting Module and the CA Embedded Entitlements Manager (EEM). Per CA, the product is slated for end of life by year 2014 but active support will continue until year 2017.
* **Operating Systems:** The CAR product only supports CentOS System, which is a closed vendor provided Virtual Appliance. All Subscription patches for the CentOS system are provided by the Vendor itself.
* **SSOi Integrations:** The CAR product (UARM) does not support integration with the current integration patterns offered by SSOi. Therefore, SSO with CAR at this point is not supported.

Table : Constraints

| Constraints |
| --- |
| * UARM does not store actual authoritative audit logs so it does not have the capability, nor is it intended, to protect the integrity of the authoritative audit data * UARM does not support PIV authentication. Since it is a flash-based application, it also cannot be integrated with CA SSO |

##### Design Trade-offs

The design trade-off for the CAR Service design is that since CAR (CA UARM) cannot integrate with SSOi, it will continue to use EEM for maintaining user information and authentication.

### Specific Requirements

This SDD provides the foundational detailed design for CAR activities under the VA Development Support program. CAR is a COTS product that sufficiently meets the detailed functional requirements.

#### Database Repository

CA UARM collects logs from a variety of applications and devices using agentless or agent-based methods. It then normalizes the log to CA Common Event Grammar (CEG) and reduces the volume of logs by filtering unwanted events based on pre-defined event filtering policies. Processed events are available for reporting, alerting, and multi-dimensional investigation. Based on log archival policy, CA UARM compresses logs and stores them on external storage systems for long-term storage. The CA UARM component is installed and configured in FIPS-only mode, per TRM.

#### System Features

##### Log Rotation

CAR supports Log file rotation based on File Age and Size, but is configured to rely on the individual Application or Servers to do its own log rotation. Most applications rename their most recent rotated log file with a ‘.1’ or a Timestamp appended to its current log file name.

For example, if a web server’s configured log file is ‘request.log’, when the file reaches the max size or max age, as per webserver’s log rotation policy, the log file is auto rotated in to request.log.1, request.log.2, and so on until the maximum number of files specified by the policy is reached.

By providing an application-specific Log file Mask, as highlighted below, CAR can aggregate logs from current and rotated log files.

Image captures the Log File Mask that can be aggregated by CAR.


Figure : Log File Mask

In cases where CAR must generate a custom report, a configuration change will be made to CAR. A file with a standard format will be provided for CAR to consume. It must be written with a delimiter recognizable by CAR.

##### Suppression Rules/Filters

CAR has a Suppression rules engine which supports built-in and custom rules to suppress noise or non-Audit events. In the example below, all events which have either an Action or Process Name or UId field missing are filtered out from the Reports.

Image captuers the suppression rules / filters.


Figure : Suppression Rules/Filters

#### Design Element Tables

N/A

##### Routines (Entry Points)

N/A

##### Templates

N/A

##### Bulletins

N/A

##### Data Entries Affected by the Design

N/A

##### Unique Record(s)

N/A

##### File or Global Size Changes

N/A

##### Mail Groups

N/A

##### Security Keys

N/A

##### Options

N/A

##### Protocols

N/A

##### Remote Procedure Call (RPC)

N/A

##### Constants Defined in Interface

N/A

##### Variables Defined in Interface

N/A

##### Types Defined in Interface

N/A

##### GUI

N/A

##### GUI Classes

N/A

##### Current Form

Refer to the [AcS Help Desk Training for CAR](http://xxxxxxxxxxxxxxx/sites/vrm/IAM/IAM%20Documents/Forms/AllItems.aspx?RootFolder=%2Fsites%2Fvrm%2FIAM%2FIAM%20Documents%2FIAM%20Development%20Documentation%2FInsignia%2D%20AcS%20Development%20Integration%20i6%2F5%2E2%2E3%2E5%20Solution%20Training%2FHelp%20Desk%20Training%20Package&InitialTabId=Ribbon%2EDocument&VisibilityContext=WSSTabPersistence).

##### Modified Form

N/A

##### Components on Form

N/A

##### Events

N/A

##### Methods

N/A

##### Special References

N/A

##### Class Events

N/A

##### Class Methods

N/A

##### Class Properties

N/A

##### Uses Clause

N/A

##### Forms

N/A

##### Functions

N/A

##### Dialog

N/A

##### Help Frame

N/A

##### HL7 Application Parameter

N/A

##### HL7 Logical Link

N/A

##### COTS Interface

N/A

## Network Detailed Design

Refer to Section 4.3 for detailed communication design for the CAR Service.

## Security and Privacy

Maintaining the Confidentiality, Integrity and Availability of user information is a critical VA requirement. Operational, Managerial and Technical safeguards to accomplish this objective are detailed in the Access Services System Security Plan (SSP) stored in the VA Risk Vision Governance, Risk and Compliance portal. The SSP details program level security and privacy safeguards in alignment with VA Directive 6500 and NIST Special Publication 800-53. The System Security Plan is a living document that is maintained and updated as new functionality is deployed or additional safeguards are implemented. For the AcS 2.0, the following security measures and integrity controls are in place.

### Security

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 requirements. 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 FIPS 140-2 (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 for these certificates to be replaced with VA Internal Certificate Authority (CA) or commercially trusted CA issued ones.

**Data at Rest**

The following table explains the data at rest points.

Table : Data Points and Security

| Data Points | Data Type | Explanation |
| --- | --- | --- |
| Oracle | Sensitive | * Stores the IdentityMinder objects- sensitive user attributes * Stores the audit log for SiteMinder and needs to be secured, but not encrypted, as there is no PII * Stores the audit log for CA IDM and must be encrypted and secured for PII * See vendor documentation for additional information regarding actual encryption algorithms used |
| Directory | Sensitive | * Stores encrypted SiteMinder policy data * Stores SiteMinder/IdentityMinder user data. Only sensitive user attributes will be encrypted * Provisioning server related objects and sensitive user attributes are encrypted * See vendor documentation for additional information regarding actual encryption algorithms used |
| File Store | Non-Sensitive/ Sensitive | * IM is stored in a JMS data in file system and contains transactional data. It does not contain any sensitive information. * 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. |

The security controls for the data at rest are managed through the encryption of sensitive attributes at the directory level for the AcS 2.0. The FIPS 140-2 encryption is applied on the identified PII and sensitive attributes stored in the AcS 2.0 directory attributes.

### Privacy

The requirements for Personally Identifiable Information (PII) are limited to data explicitly required in VA 6501 and NIST SP 800-63. However, the implementation adheres to the following integrity controls to ensure that acceptable security standards are met.

#### CAR

The CAR service does not have the permission to alter any information contained in other components of the IAM solution. Rather, it has a read only access and therefore the risk is very low. The CAR service will come pre-equipped with a car admin account already created. The credentials will be provided to VA staff acting as the CAR admin that will then create further users (privileged and regular) as necessary. The access by these users is monitored as well. Moreover, UARM self-monitors its own activity and logs are stored in secure and non-repudiated fashion.

#### Confidentiality of Sensitive Information

The CAR service is not exposed to any external network and the transmission of information occurs on SSL channel. The user information is secured using proper access control implementation.

#### Privacy of Personal Information

The system for the CAR Service does not intentionally store Personally Identifiable Information (PII). However, it could process PII data if it is contained in the collected logs/events. In this scenario, PII of the user is stored. Data in transit is FIPS mode encrypted. UARM admin users are stored in an internal directory and passwords for them are encrypted and maintained by COTS products.

#### Process Integrity

The system is designed to provide validation for input forms before submission and store the information for the user record. No information is entered by the end user other than the user credentials when the administrators are creating new accounts. The CAR service provides proper processing controls such as making sure that the same user ID is not issued to two users and maintaining the uniqueness of IDs. Additionally, with the full auditing of transactions, any misuse of authority is discernible and traceable in the audit logs/reports.

## Service Oriented Architecture/ESS Detailed Design

N/A

### Service Description for CAR

N/A

### Service Design for CAR

N/A

#### Introduction

N/A

##### Purpose and Scope of Service

N/A

##### Links to Other Documents

N/A

#### Service Details

##### Service Identification

N/A

##### Service Versions

N/A

##### Summary of Design and Platform Details

###### SOA Pattern(s) Implemented

N/A

###### COTS Platform vendor names and versions for hosting platform

N/A

#### Dependencies

N/A

#### Service Design Details

N/A

##### Interface Technical Specs

N/A

###### Service Invocation Type

N/A

###### Service Interface Type

N/A

###### Service Name

N/A

###### Interface

N/A

###### End Points

N/A

###### Operations or Methods

N/A

###### Message Schemas

N/A

##### Information Model

N/A

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

N/A

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

N/A

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

N/A

###### Use Cases (Use Case Model)

N/A

###### Interaction Diagrams

N/A

#### Gap Analysis

N/A

##### Variances from Enterprise Target Architecture

N/A

##### Variances from SLDs

N/A

##### Variances from Standards and Policies

N/A

##### Justification for Exceptions and Mitigation

N/A

# External System Interface Design

Interfacing mechanisms for CAR are encompassed within agent architecture.

Refer to the specific ICD for additional details.

## Interface Architecture

All server names where a CAR agent is installed, along with the names of the connectors configured on those agents and the applications that they connect to, are shown in the table below.

Table : CAR Server

| CAR Agent Server | Agent Connector Names | Integrated Application |
| --- | --- | --- |
| xxxxxxx | va\_cspip\_connector, and VDS\_log\_connector | CSP (VA Logon), VDS |
| xxxxxxx | VAAFI\_Trace\_Connector1 | SSOe |
| xxxxxxx | VAAFI\_Trace\_Connector2 | SSOe |
| xxxxxxx | VAAFI\_Trace\_Connector3 | SSOe |
| xxxxxxx | VAAFI\_Trace\_Connector4 | SSOe |
| xxxxxxx | VAAFI\_Weblogic\_Connector1 | SSOe |
| xxxxxxx | VAAFI\_Weblogic\_Connector2 | SSOe |
| xxxxxxx | VAAFI\_Weblogic\_Connector3 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector1  VAAFI\_3POBRequest\_Connector1 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector2  VAAFI\_3POBRequest\_Connector2 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector3  VAAFI\_3POBRequest\_Connector3 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector4  VAAFI\_3POBRequest\_Connector4 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector5  VAAFI\_3POBRequest\_Connector5 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector6  VAAFI\_3POBRequest\_Connector6 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector7  VAAFI\_3POBRequest\_Connector7 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector8  VAAFI\_3POBRequest\_Connector8 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector9  VAAFI\_3POBRequest\_Connector9 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector10  VAAFI\_3POBRequest\_Connector10 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector11  VAAFI\_3POBRequest\_Connector11 | SSOe |
| xxxxxxx | VAAFI\_Request\_Connector12  VAAFI\_3POBRequest\_Connector12 | SSOe |
| xxxxxxx | sailpoint\_cert\_connector1  sailpoint\_policy\_connector1  sailpoint\_risk\_connector1 | SSOe |
| xxxxxxx | sailpoint\_cert\_connector2  sailpoint\_policy\_connector2  sailpoint\_risk\_connector2 | SailPoint |
| xxxxxxx | va-sm-connector | SiteMinder |
| xxxxxxx | App\_ESR\_ODBC\_Roles\_Connector  App\_ESR\_Logins\_Connector  va-eSig-connector | ESig, ESR |
| xxxxxxx | va\_prov\_connector | Provisioning |
| xxxxxxx | va\_sso\_connector\_1 | SSOi |
| xxxxxxx | va\_sso\_connector\_2 | SSOi |
| xxxxxxx | sac\_pdp\_connector | SAC |
| xxxxxxx | sac\_asm\_connector | SAC |

## Interface Detailed Design

Refer to the ICDs for each integration for additional details.

# Human-Machine Interface

This section outlines the interfaces used to interact with the CAR solution. The interfaces may be categorized based on users as follows.

* **CAR Users:** The web interface used by the CAR Users to run standard reports
* **CAR Privileged:** The web interfaces used by the administrators to manage CAR Users and generate, view, and schedule ad hoc reports and manage alerts
* **CSP (VA Logon) Privileged:** The web interfaces used by the CSP (VA Logon) Privileged User to manage CSP (VA Logon) reports
* **SAC Privileged:** The web interfaces used by the SAC Privileged User to manage SAC reports
* **CAR ISSO:** The web interfaces used by the VA ISSO for auditing purposes
* **Super User:** CAR Super User to create and manage Privileged Users

## Interface Design Rules

The following design rules are applicable to the user interfaces for the CAR:

* The user and administrator interfaces comply with VA’s branding specifications
* The interface is easy to navigate with self-explanatory instructions/fields
* The interface provides user friendly messages/information on error
* 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 non-administrator, 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

## Inputs

The CAR service activities are web pages, accessible via VA standard web-browsers. Navigation and data entry require no special devices beside mouse and keyboard, while meeting Section 508 compliance where appropriate.

Refer to Section 8.4.1 for web interface screen information regarding inputs to the system.

## 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 CAR:

* Portable Document Format (PDF)
* Comma Separated File (CSF)
* Microsoft Excel

## Navigation Hierarchy

CAR is based on a COTS product and does not use custom interfaces.

### Screen Shots

Refer to the [AcS Help Desk Training for CAR](http://xxxxxxxx/sites/vrm/IAM/IAM%20Documents/Forms/AllItems.aspx?RootFolder=%2Fsites%2Fvrm%2FIAM%2FIAM%20Documents%2FIAM%20Development%20Documentation%2FInsignia%2D%20AcS%20Development%20Integration%20i6%2F5%2E2%2E3%2E5%20Solution%20Training%2FHelp%20Desk%20Training%20Package&InitialTabId=Ribbon%2EDocument&VisibilityContext=WSSTabPersistence)to review all navigational screen shots.

# 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 a 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: Date:

xxxxx

IAM Program Manager

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signed: Date:

xxxxx

AcS Program Manager

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signed: Date:

xxxxxx

IAM Integrated Project Team (IPT) Chair and Business Sponsor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signed: Date:

xxxxx

IAM BPMO Director

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signed: Date:

xxxxx

Chief Architect

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signed: Date:

xxxxx

SEDR

1. SEDR Additional Information
   1. Identification of Technology and Standards

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 2.0. This document applies to the following systems and software.

Table : System Identification

| Name | Description | Abbreviation | Version | Release |
| --- | --- | --- | --- | --- |
| Compliance Audit and Reporting | Provides audit and reporting capability based on data made available to the Access Services. | CAR | V 2.5.X | N/A |

* 1. 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 : Policies, Directives, and Procedures

| # | Issuing Agency | Policy, Directive, or Procedure | Purpose |
| --- | --- | --- | --- |
| 1 | VA | VA 6500 Handbook | * Directive Information Security Program * Defining overall Security Framework for VA |
| 2 | VA | VA 6501 Directive | * VA Identity Verification In-Person Proofing (IPP) Process * Defining overall Identity Proofing Methodology for VA IAM |
| 3 | VA | VA 6300 Directive | * Directive Records and Information Management * Defines information management framework for VA Access Services |
| 4 | NIST | SP 800-53-4 | * Special Publication – Recommended Security Controls for Federal Information Systems and Organizations * Defines the required security controls for IT systems under the Federal Information Security Management Act (FISMA) |
| 5 | NIST | SP 800-63-2 | * Special Publication – Electronic Authentication Guideline * Defines levels of assurance in user identities presented to IT systems over open networks * Defines the data and procedural requirements for VA Access Services |
| 6 | NIST | FIPS-201-2 | * Federal Information Processing Standards Publication – PIV of Federal Employees and Contractors * Provides Identity Proofing, credentialing and chain of trust requirements and processes * Defines the method for secure administrative interaction and control |
| 7 | NIST | FIPS-140-2 | * Federal Information Processing Standards Publication (FIPS) – Security Requirements for Cryptographic Modules * Defines the cryptographic standards and requirements |
| 8 | NIST | SP 800-122 | * Guide to Protecting the Confidentiality of Personally Identifiable Information (PII) * Provides technical procedures for protecting PII in information systems * Defines the information which can be used to distinguish or trace an individual's identity |
| 9 | US Congress | Section 508 Amendment to the Rehabilitation Act of 1973 | * Section 508 Electronic and information technology requirements for Federal departments and agencies * Accessibility, development, procurement maintenance, or use of electronic and information technology * Defines the “Human-Machine Interface” accessibility requirements |
| 10 | OMB | M-04-04 | * Memorandum to the Heads of All Department and Agencies – E-Authentication Guidance for Federal Agencies * Defines the E-Authentication requirement |
| 11 | OMB | M-11-11 | * Requirements for Accepting Externally-Issued Identity Credentials * FICAM architecture and procedures for federal agencies |
| 12 | GSA | FICAM | * Federal Identity, Credentialing and Access Management (FICAM) Roadmap and Implementation Guidance * Provides the common segment architecture and implementation guidance for federal ICAM programs |
| 13 | White House | NSTIC | National Strategy for Trusted Identities in Cyberspace (NSTIC) – Provides guidance for identity trust in cyberspace |
| 14 | US Congress | FISMA | * FISMA of 2002, Public Law 107-347 |
| 15 | US Congress | E-Government Act of 2002 | * Federal Management and Promotion of Electronic Government Services * Defines the requirements for electronic services |
| 16 | US Congress | The Privacy Act of 1974 | * § 552a. Records maintained on individuals * Defines VA Access Services Privacy assessment and control requirements |
| 17 | National Archives and Records Administration (NARA) | Federal Records Act | Establishes the framework for records management programs in Federal Agencies |
| 18 | VA | VA D 0735 | * Homeland Security Presidential Directive 12 (HSPD-12) Program * 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 |
| 19 | OMB | M-05-24 | Implementation of HSPD 12 – Policy for a Common Identification Standard for Federal Employees and Contractors |

* 1. RTM

Refer RSD and Rational Tool Composer for the AcS 2.0 RTM.

* 1. Packaging and Installation

N/A

* 1. Design Metrics

N/A

* 1. Acronym List and Glossary

The abbreviations and terms used in this SDD are defined in the [Identity and Access Services Master Glossary](http://xxxxxxxxxxxxxxxxx/warboard/ProjectDocs/Access_Services/Identity_and_Access_Services_Master_Glossary.pdf).

* 1. Required Technical Documents

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

* 1. Attach Documents

The AERB Design Compliance Decision Certificate is attached upon approval of the SDD.