

---

# AUTOMATION ANYWHERE ENTERPRISE 10 LTS

---

## SECURITY ARCHITECTURE

|                               |            |
|-------------------------------|------------|
| Document Version              | 1.0        |
| Date of Publication           | 03-14-2018 |
| Update(s) to Document Edition | -          |



## Table of Content

|  |    |
|--|----|
| Executive Summary.....   | 7  |
| 1. Automation Anywhere Enterprise Overview.....                          | 8  |
| 1.1 Introduction.....  | 8  |
| 1.2 Distributed Architecture .....                                       | 8  |
| 1.3 Solution Components .....  | 9  |
| 1.3.1 Control Room (CR) .....  | 9  |
| 1.3.2 Bot Creator: Development Client.....                               | 10 |
| 1.3.3 Bot Runners: Runtime Client .....                                  | 11 |
| 1.3.4 IQBots - Cognitive Bots.....                                       | 11 |
| 1.3.5 Bot Insight.....   | 11 |
| 1.3.6 BotFarm .....  | 12 |
| 1.4 Network Architecture and Boundary Controls.....                      | 13 |
| 1.5 Distributed architecture with HA/DR support.....                     | 14 |
| 1.5.1 Distributed Cache.....   | 15 |
| 1.5.2 HA/DR Deployment.....  | 15 |
| 2 Security Architecture.....   | 17 |
| 3 Access Control.....  | 18 |
| 3.1 Independent Control Planes: Bot Creators and Bot Runners.....        | 18 |
| 3.2 Role Based Access Control (RBAC).....                                | 18 |
| 3.2.1 RBAC on Bots .....   | 19 |
| 3.2.2 RBAC on Bot Runners .....  | 19 |
| 3.2.3 RBAC in Credential Vault: Credentials Management .....             | 20 |
| 3.2.4 Role-based Processing Domains .....                                | 20 |
| 3.2.5 RBAC on Audit Log .....  | 20 |
| 3.2.6 RBAC on Operations Room .....                                      | 21 |
| 3.2.7 RBAC on User Management .....                                      | 21 |
| 3.2.8 RBAC on Roles and Permissions Management.....                      | 21 |
| 3.2.9 RBAC on Bots Schedules .....                                       | 21 |
| 3.2.10 RBAC on License Management .....                                  | 21 |
| 3.3 Secure Application Partitioning.....                                 | 21 |
| 3.3.1 Secure Application Partitioning: Stealth Mode for automation ..... | 22 |
| 3.3.2 Secure Application Partitioning: Disable mouse and keyboard.....   | 22 |

|       |   |    |
|-------|---|----|
| 3.3.3 | Secure Application Partitioning: Configurable automation timeout .....                    | 22 |
| 3.3.4 | Secure Application Partitioning: Centralized control on automation running remotely ..... | 23 |
| 3.4   | Bot Execution Access: Dynamic Access Token .....  | 23 |
| 3.5   | Secure Credential Store: Credential Vault (CV) .....                                      | 23 |
| 3.5.1 | Configuration of CV: Encryption .....   | 23 |
| 3.5.2 | Credential storage in CV .....  | 24 |
| 3.5.3 | Credential Provisioning to bots .....   | 24 |
| 3.6   | Security at-rest .....  | 25 |
| 3.6.1 | Credentials storage in CV .....   | 25 |
| 3.6.2 | PGP Command .....   | 25 |
| 3.6.3 | Secure Recording .....  | 26 |
| 3.6.4 | Protection of software binaries .....   | 26 |
| 3.7   | Security in-transit: Support for secure protocols .....                                   | 26 |
| 3.7.1 | Authentication with Control Room .....  | 27 |
| 3.7.2 | Communication between CR and database .....   | 28 |
| 3.8   | Password Hashing .....  | 28 |
| 3.9   | Network Security Overview .....   | 28 |
| 3.10  | Secure Protocols in use .....   | 30 |
| 3.11  | List of Port Numbers .....  | 31 |
| 4     | Change Management .....   | 32 |
| 4.1   | Versioning and Operational Control: .....   | 32 |
| 4.2   | Baseline Inventory Controls: Bot Creators, Bot Runners and Bots .....                     | 32 |
| 4.3   | Change Control & Documentation: RBAC .....  | 32 |
| 4.4   | Change Control: Automated Configuration Changes .....                                     | 32 |
| 4.5   | Software Usage and License Management .....   | 33 |
| 4.6   | Change Management: Dual Authorization .....   | 33 |
| 5     | Identification and Authentication .....   | 33 |
| 5.1   | Multi-domain AD support .....   | 33 |
| 5.2   | User Authentication for Control Room Access .....   | 33 |
| 5.3   | Authentication Failure Messages .....   | 34 |
| 5.4   | Auto Log-off .....  | 34 |
| 5.5   | Authentication for Bot Creators .....   | 34 |
| 5.6   | Authentication for Bot Runners .....  | 35 |
| 5.7   | Authentication of Bot Runners: Dynamic Access Token .....                                 | 36 |

|       |  |    |
|-------|--|----|
| 5.8   | Integration with Third-Party Identity and Access Management Solutions..... | 37 |
| 6     | Compliance and Vulnerability Scanning .....                                | 38 |
| 6.1   | Windows Logo Certification .....   | 38 |
| 6.2   | Systems Security Analysis.....   | 39 |
| 6.2.1 | Static & Dynamic Code Analysis: Veracode Vulnerability Scanning.....       | 40 |
| 6.2.2 | Network Vulnerability Analysis: Nessus Vulnerability Scanning.....         | 40 |
| 6.2.3 | OWASP ZAP Vulnerability Scanning .....                                     | 41 |
| 6.2.4 | Penetration Testing.....   | 41 |
| 6.3   | FIPS 140-2 Compliance.....   | 41 |
| 7     | Auditing and Logging.....  | 41 |
| 7.1   | Audit Logs.....  | 41 |
| 7.1.1 | RBAC on Audit Log .....  | 42 |
| 7.1.2 | Control Room Bot Creator and Runner Activity Logging.....                  | 42 |
| 7.1.3 | Audit of Bot Runner Operations .....                                       | 42 |
| 7.1.4 | Audit Log Non-repudiation .....  | 43 |
| 7.1.5 | Export of Audit Logs .....   | 43 |
| 7.2   | Activity Logging.....  | 43 |
| 7.3   | Version Control.....   | 44 |
| 7.3.1 | Deployment of bots .....   | 45 |
| 7.4   | Email Alert notifications.....   | 45 |
| 8     | Additional Security Controls .....   | 46 |
| 8.1   | Restrict CR install from database system administrator account.....        | 46 |
| 8.2   | Auto lock the device .....   | 46 |
| 8.3   | Use of SHgetKnownFolderPath function.....                                  | 46 |
| 8.4   | API Level security.....  | 46 |
| 8.5   | Clean Uninstall.....   | 46 |
| 8.6   | Store data in “Program Data” folder .....                                  | 46 |
| 8.7   | Protected handling of MSVC DLL files.....                                  | 46 |
| 8.8   | Assembly Manifest.....   | 46 |
| 8.9   | Application path on network.....   | 47 |
| 8.10  | Auto-login without disabling legal disclaimer .....                        | 47 |
| 8.11  | Secure Java automation .....   | 47 |
| 8.12  | Automation in non-English languages .....                                  | 47 |
| 9     | Additional Security features in Next Release (v11).....                    | 47 |

|      |  |    |
|------|--|----|
| 9.1  | Native CyberArk integration.....                     | 47 |
| 9.2  | Support for SAML and Kerberos .....                  | 47 |
| 9.3  | Role Based Access Control (RBAC) On Credentials..... | 47 |
| 9.4  | Tighter integration with Active Directory .....      | 48 |
| 10   | Appendix .....                                       | 48 |
| 10.1 | Control Room Directories Listing.....                | 48 |
| 10.2 | Client Directories Listing .....                     | 50 |
| 10.3 | List of executables .....                            | 51 |
| 10.4 | List of Cryptographic Providers .....                | 51 |
| 11   | Glossary of terms.....                               | 51 |

## Executive Summary

Many of the world's largest financial enterprises rely on Automation Anywhere's secure digital workforce platform to automate security sensitive operations. Our security architecture is founded on Least Privilege principles and a strict Separation of Duty model with 41 technical controls implemented across seven NIST 800-53r4 Control Families. Controls are applied across our three components: the Control Room (CR), a Windows-based server, and Bot Creators (development systems) and Bot Runners (Bot execution run times) thru the Bot lifecycle from creation thru decommissioning. Our security architecture and underlying controls are mapped to industry best practices as defined by NIST and can be readily mapped to other frameworks such as CoBIT (SOX) and ISO 27002.

**Access Controls.** AAE limits and controls human and bot access to logical resources across components.

- Two independent control planes enforce least privilege. Only developers can read or write, while only authorized CR users can execute automations, (CR authorizes and executes) subject to fine-grained Role Based Access Controls (RBAC) down to individual automations (bot), Bot Runners and domains.
- Bot-level Separation of Duty is enforced. Each bot is obfuscated and executed by its corresponding authorized Bot Runner(s). Tampering at execution is prevented via Stealth Mode and complete user interface disabling.
- Bot execution is controlled via RBAC. Domain privileges are defined across groups of bot and Bot Runners.
- Security at-rest and in-transit. All access credentials are secured at-rest via a central credential vault with support for third-party credential stores such as CyberArk. All communications are secured in-transit via SSL and TLS and credentials are secured in memory with Microsoft's Data Protection API.

**Configuration Management** is controlled at both bots and Bot Runner levels.

- The CR authorizes, enforces and logs changes to all Bot Creators and Bot Runners,
- Bots are controlled via robust versioning system, for roll-back and full event logging,
- Bot change control on execution is enforced thru encryption and authentication,

**Identification and Authentication** is controlled thru Windows authentication services.

- Bot Creators use Active Directory for authentication
- Bot Runners have two levels of authentication, for auto-login initiate the runner and execution of Bots.
- Credentials are secured at-rest and in-motion thru our Credential Vault or integration with third party products.

**Risk Assessment** is undertaken on Static, Dynamic and Network-based Vulnerability Assessments. **Audit and Accountability** are established through event capture, logging and auditing on all three components with granular event capture at the bot level and non-repudiation. Bot Insight embedded analytics

# 1. Automation Anywhere Enterprise Overview

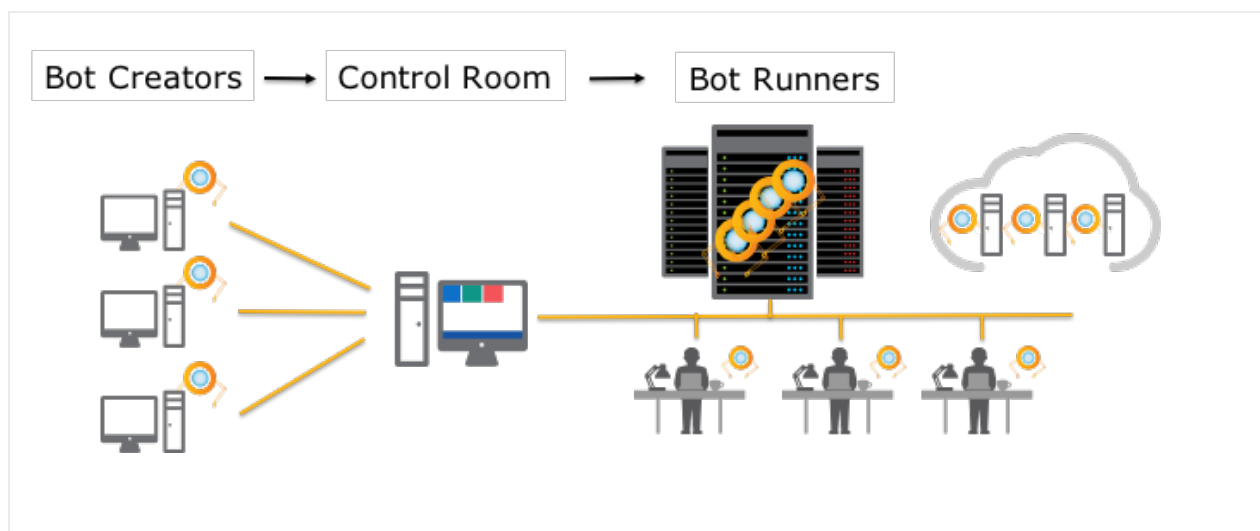
## Introduction

Over 500 Enterprises have chosen our Enterprise Digital Workforce platform as the foundation to deliver complex business work securely and at scale. Automation Anywhere leads the industry with the largest installed base of bots (automation tasks are called bots), many times larger than our nearest competitor. Our large installed base, and corresponding product maturity, translates into a more robust and resilient product with a depth and breadth of features unparalleled in the RPA space. Our bots have proven capable of automating thousands of applications across varying infrastructures and industries, including some of the world's largest Banks and Telecommunication companies. This is the reason seventeen of the top twenty Business Process Outsourcers (BPOs) have chosen us to deliver their Enterprise-class Digital Workforce.

Our solution is designed to gradually scale up to match each step of the RPA journey. We avoid imposing complexity in the early stages to ensure short time to benefit and reduced up-front expense. No heavy workflow methodologies or burdensome infrastructure requirements are there during the initial phases. Once initial pilots are ready to scale up, our centralized Control Room delivers the automated provisioning, orchestration and controls required for large scale deployments. From desktops to datacentre and cloud-based scale-ups, our solution distinguishes itself by delivering Enterprise-class management and control, without diminishing rapid time to value, across any infrastructure.

## Distributed Architecture

Automation Anywhere platform is deployed using a distributed architecture. Centralized management is accomplished via a web-based server, called the Control Room, to manage all development and execution of the digital workforce. The Control Room (CR) is connected to Bot Creators and Bot Runners. Bot Creators are Windows-based development systems used for authoring and tailoring of automations. Bot Runners execute the automations, they are run time systems installed on Windows machines. Bot Runners may be deployed on Desktops or on Virtual Machines (VMs) in Datacentres or Cloud. Only Bot Creators and Bot Runners that are authenticated to the CR may develop and execute automations.





## Solution Components

AAE's distributed architecture consisting of three components: the Control Room, a Windows-based server, and Bot Creators and Bot Runners, Windows-based clients. No confidential or sensitive personal data is retained or stored within the Control Room, Bot Creators or Bot Runners. Only Bot Insight, our embedded analytics can be configured to store data.

### Control Room (CR)

At the centre of our Digital Workforce is the Control Room, the brains of the bot workforce. The CR provides a single pane of glass for orchestrating your digital workforce. It runs on MS Internet Information Services (IIS) 8 or later in Windows Server 2012 R2 and uses an integrated MS SQL Server 2012 database. The CR delivers reliable, scalable, and secure, automation collaboration, bot deployment and execution. From the CR bots can be deployed from desktop to virtual and cloud-based infrastructure. Some processes start on desktops and, as RPA matures, migrate to batch processing in datacentres. While others may require persistent human interaction, or do not otherwise lend themselves to server-based processing. Regardless of how it is deployed, the CR delivers consistent and auditable management and control over the RPA infrastructure to ensure that any bot can be deployed and managed securely at enterprise scale. CR features can be broadly divided into following 3 categories:

#### Centralized Automation Deployment

- The CR acts as the single point of access and control for bot execution.
- All Bot Creators and Bot Runners must be registered to a control room before they are operable. Only the CR can execute Bots on Bot Runners.
- Only Bots loaded in the Control Room may execute on Bot Runners. All bots across the enterprise are first uploaded to CR. Every single automation activity must authenticate against CR.
- CR provides Bots upload & download and in-built version control features to facilitate seamless collaboration for end to end business process automation by multiple users.
- All scheduling is managed by the CR. bots are deployed on the Bot Runners either ad-hoc or on pre-defined schedules. Once the schedules are created, CR automatically and intelligently picks up the subsequent updates to bots, without any need to alter automation schedules.

#### Centralized Access Control and Collaboration

- Least Privilege and Access Controls user access are implemented in the CR via Role Based Access Control (RBAC).
- All Users and Roles are created and managed from the CR.

#### Centralized Workforce Management

- CR dashboards provides a single pane of glass of the automation infrastructure.
- CR receives real time heartbeat & telemetry from automations with events, exceptions or alerts.
- Unauthorized users cannot pause, resume or stop any of the ongoing automations on any bot runner.
- All historical automation data is logged in and available through CR Audit Logs.

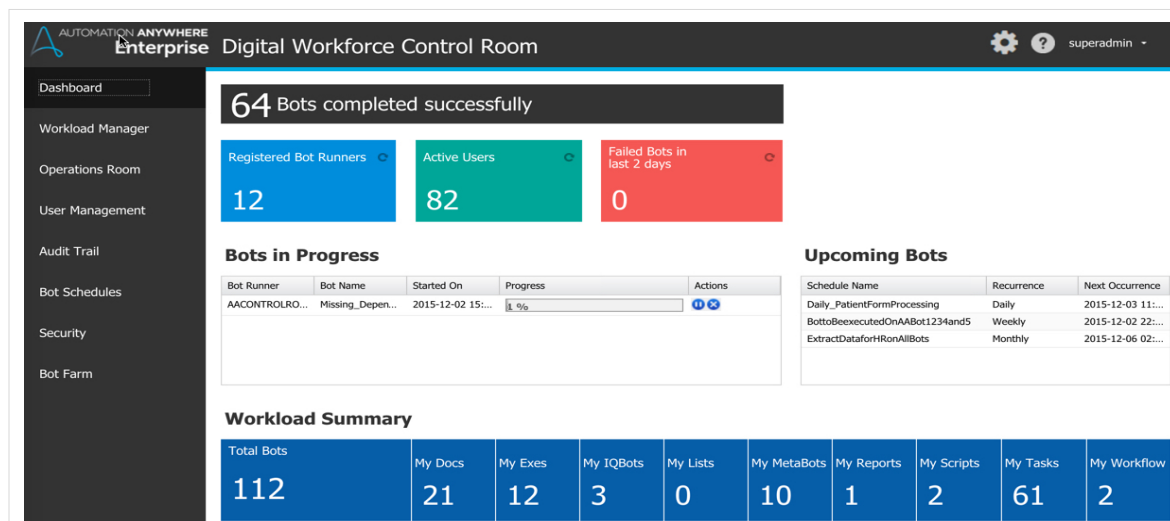


FIGURE 2: CENTRALISED CONTROL

## Bot Creator: Development Client

Bot creators are used for authoring and unit testing of automation. Bot creators run on windows and must authenticate against an active CR, on a failed closed principle. Bot Creators only access the CR to upload and download bots.

Developed bots are uploaded to the version Control System on the CR for application of Software Design Lifecycle Controls (SDLC) best practices to preserve Bot Integrity with features to check-in, check-out, rollback, compare across bot versions. Below is a brief description of the Development Client and associated nomenclature.

**TaskBots:** are used to automate rule based business processes which use structured data. TaskBots are built using the solution accelerators (such as SMART Recorder, extensive command library) available in Bot Creator. TaskBots are built for Object based, Image based or coordinate based automation, depending on the application type. As described above, TaskBot properties can also be configured to set automation triggers, execution priority, bot timeout, bot re-runs, alert notifications and security controls.

**MetaBots:** are reusable automation building blocks which deliver speed and scalability to your automation landscape. MetaBots are built once and can be re-used across hundreds of TaskBots and thus accelerate time to value. MetaBots enables compliance to organizational Standard Operating Procedures (SPOs) since MetaBots automation must be used as a black box and cannot be altered. MetaBots facilitates offline automation by allowing automation creation even when live application is not available. MetaBots provide extensibility to our platform by enabling API based integration with third party applications. Automation created using MetaBots can be calibrated against any changes in the business application. This means that entire automation need not be changed with change in application and makes the automation resilient.

## Bot Runners: Runtime Client

Bot Runners do not have any create or edit access on automation but only Bot Runners can execute bots. Once a bot is created and uploaded by the Bot Creator, the user on the CR can schedule and execute the Bot on an authorized Bot Runner(s).

Each of the Bot Runners must first authenticate against enterprise Active Directory by securely fetching the user credentials from Centralized Credential Vault (CV). Next level of authentication is performed against an available CR. CR centrally deploys the automation only to authorized and authenticated Bot Runners.

All credentials are secured at-rest and in-motion according to best practices as required by NIST SC-8, Transmission Confidentiality and Integrity. When the Bot executes on Bot Runners, they securely access enterprise business applications utilizing credentials encrypted and stored in the CV or in a third party credential store. Once the automation execution is finished, remote Bot Runners are reverted to their original state (i.e. locked or log off). All credentials provisioning is done over HTTPS. As soon as the credentials usage is over, credentials are encrypted using Microsoft Data Protection API, which uses AES-256 bit encryption. Only encrypted credentials stay in system memory. All credentials are secured through secure pre and post processing. See Section 3.5 for details on Credential Vault (CV).

## IQBots - Cognitive Bots

IQBots have their own development system and are called from TaskBots. They use user-defined semantic domain models to extract information from semi and un-structured data such as scanned documents and images (e.g. invoices, contracts). IQBots' cognitive capabilities deliver the combination of unsupervised and supervised learning to deliver maximum accuracy. IQBots are initially trained by humans, then they learn and grow smarter with every human validation. IQBots can be synergistically used along with IBM Watson to deliver comprehensive cognitive business process automation. Future IQBots will use natural language cognition to add a Sentiment Skill, enabling them to identify people's attitudes and emotional states from language to gauge context and drive appropriate actions.

## Bot Insight

Bot Insight is a fully embedded analytics platform for operational and business analysis, it is fully integrated into the CR (see Figure 3 below). Any TaskBot or MetaBot variable can be tagged and logged for analysis by Bot Insight, we recommend anonymizing data by omitting any Personally Identifiable Information at all times. Bot Insight performs the near real-time measurement of Bot operations and business process data and automatically creates dashboards using advanced algorithms specific to each bot. CR provides user management based on RBAC and administrators can customize dashboards and publish to a wider user audience.

From a security perspective, delivers near real-time analysis to conform to NIST SI-4 requirements for automated tools for near real-time monitoring of system information, creation of alerts and situational awareness. The use of tags, logs and filters facilitates extensive auditing for capture of significant events as required by NIST SI-7. Bot Insight provides fine grained event logs on Bot operations, permitting time series analysis, interactive drilldowns, ad-hoc discovery and insight into operational and business process. All data and analysis can be exported using CSV customized formats into a Security Information & Event Management (SIEM) for further analysis.

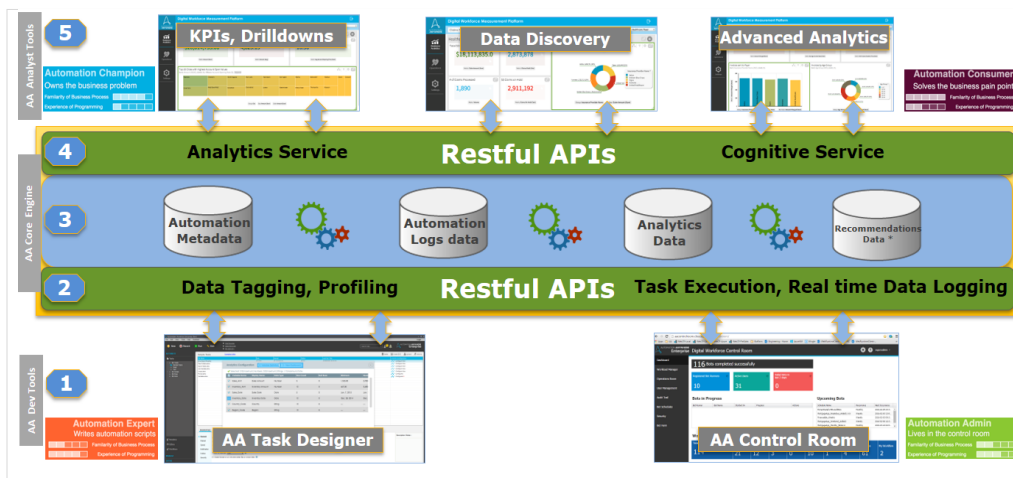


FIGURE 2: BOT INSIGHT

## BotFarm

BotFarm delivers RPA-as-a-service through Automation Anywhere platform. It provides on- demand digital workforce ramp-up and ramp-down with minimal supervision and addresses the seasonality in automation demand. It uses AWS public cloud to enable usage based metering and delivers unlimited horizontal scalability.

BotFarm's on-demand elastic digital workforce and the automation is continuously monitored and comprehensively audit logged for all compliance and logging purposes. BotFarm has multi-node high availability and powers SLA driven business continuity despite change in automation workload.

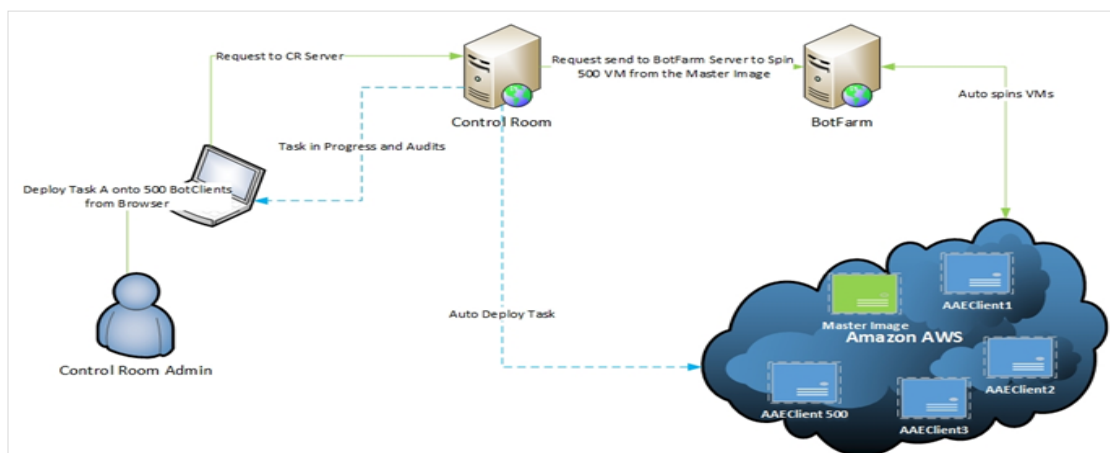
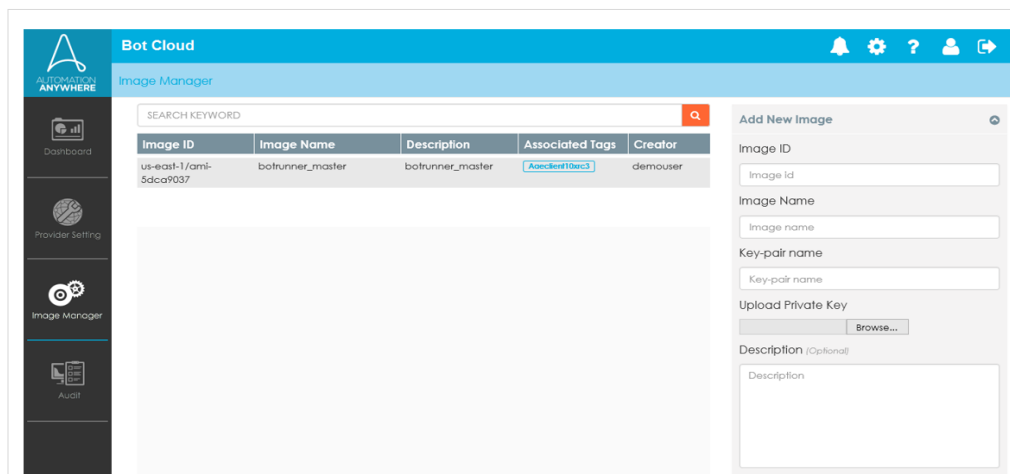


FIGURE 4: BOTFARM ARCHITECTURE



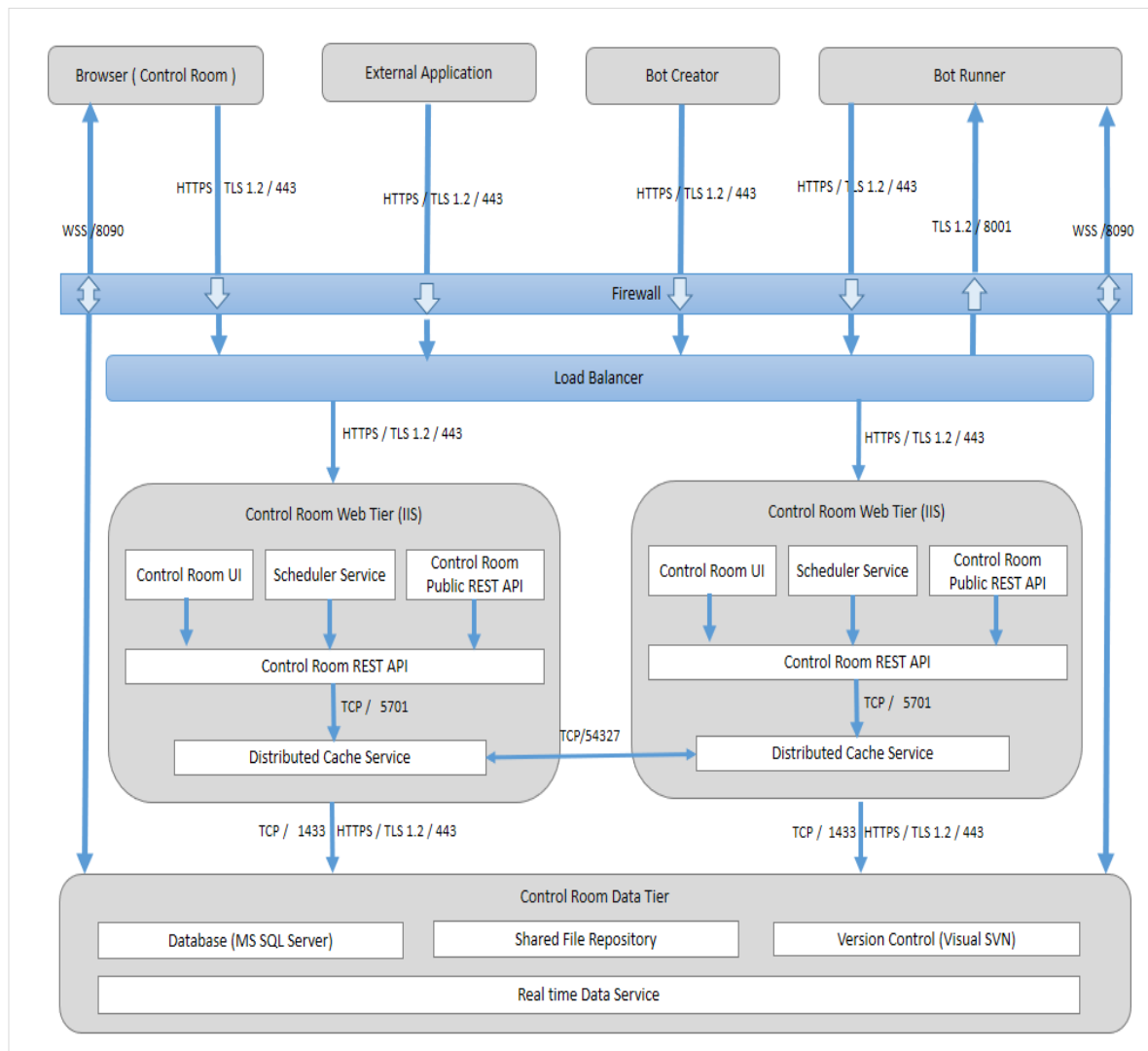
**FIGURE 5: BOTFARM**

## Network Architecture and Boundary Controls

All the communication between authorized and authenticated Bot Creators, Bot Runners, and the CR use secure protocols and pass through a Network Firewall and a Load Balancer, based on customer's deployment topology consistent with best practices as defined by NIST SC-7 Boundary Protections requiring connections to networks only through managed interfaces using devices such as firewalls and load balancers. All communications are denied by default and only allowed thru specific services outlined below (see Figure 2 below). See the Network Security section for more detailed discussion of the network configuration.

- Secure protocols (e.g. TLS 1.2, HTTPS, WSS) are used to communicate between different modules of Client and Control Room. Two Application Servers with Data tier are represented in above diagram to describe how load balancing and synchronization take place between servers.
- Distributed Cache Service is introduced from Control Room 10.3.0 version. It holds application specific data and shares with REST Service as well as the other instances of Cache Service. Internally it uses Hazelcast Distributed Cache mechanism.
- Real time Data Service is a common service for all Application Servers. It receives and broadcasts real time task progress data coming from each running Bot Runner. It listens on WSS protocol. It plays a mediator role between browser where Control Room is opened and a Bot Runner where task is running.
- Shared File Repository is a file system location where all the bots reside physically. It is shared across all the Application Servers, so that same repository view and operations become possible.
- Version Control System (VCS) and MS SQL Server are external software applications. They may or may not be on the Data tier depending upon the need.
- Data tier can be configured for failover cases separately if high availability is concerned. Refer the AAE Control Room 10.5.0 High Availability Configuration Guide for detailed information.
- All network connections are terminated at the conclusion of each session or within a specified time period, i.e. no network services are enabled via keep-alive processes.

Go to section 3 for the details on communication between client and server layers.

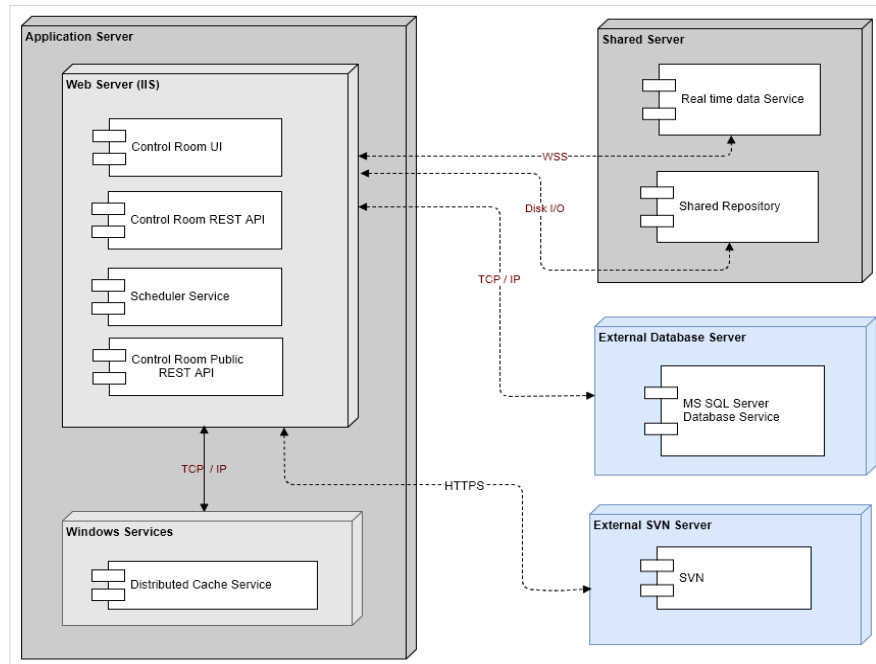


**FIGURE 6: PLATFORM ARCHITECTURE**

## Distributed architecture with HA/DR support

Automation Anywhere platform is self-contained within customer's environment and therefore mitigates the risk of "cross-contamination" from an unlikely event of a security breach in another customer's network. It supports distributed architecture to deliver the optimal performance and security. Following are the main distributable components of Control Room which can be clustered to achieve High Availability:

- Automation Anywhere Web Socket Server Service
- Subversion Service
- Microsoft SQL Server 2012 and higher



**FIGURE 7: DISTRIBUTED MODE**

## Distributed Cache

CR architecture uses distributed cache to update all other nodes as soon as any information is updated in one of the nodes. This ensures fastest data synchronization across all the nodes and delivers seamless user experience. Our platform uses clustering mechanism to implement distributed cache, to synchronize all data operations. For example, once the Credential Vault is opened from one node, it is automatically opened for all other nodes too.

## HA/DR Deployment

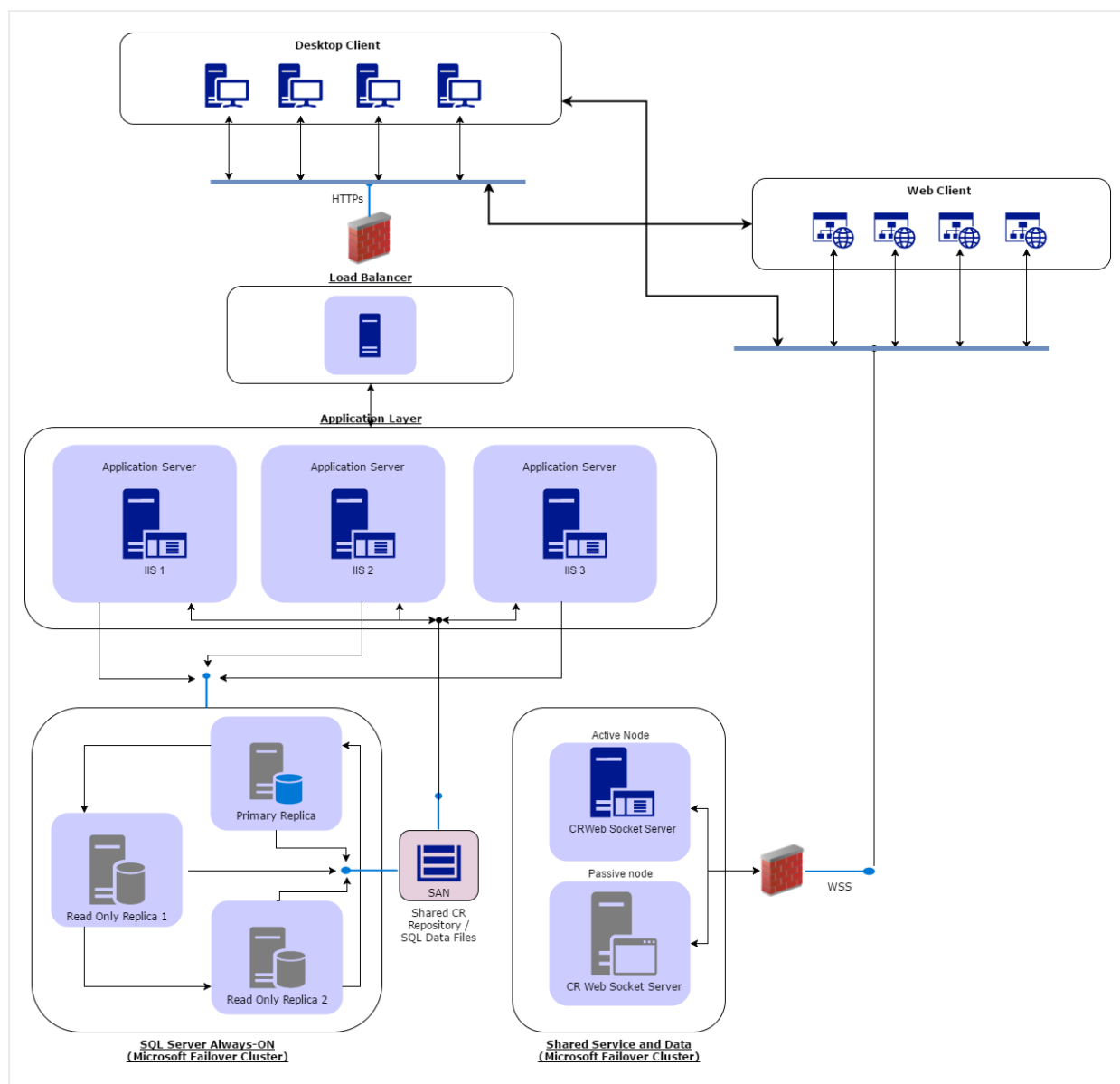
Due to the on-premise deployment, Automation Anywhere delivers solution High Availability (HA) and Disaster Recovery (DR) using customers' existing infrastructure and processes. This allows us to configure the platform components so that they easily integrate with customers' existing HA/DR infrastructure, avoiding the need to change existing process / solution. All communication from Automation Anywhere to customers' HA/DR infrastructure is over secure HTTPS and WSS layer.

High availability infrastructure support across multiple control rooms using a network load balancer. Network load balancing is achieved by running multiple instances of the CR, each in its own IIS web server and a load balancer in front of it handling web requests. The Load Balancer can either be an external appliance (such as an F5 load balancer) or you can use the built-in Microsoft Network Load Balancer (NLB) available on Server 2012. With NLB, the IIS Servers in the cluster also acts as the Load Balancer.

The IIS instances access a separate physical box containing shared services (that may be customer-specific), Web Socket Services, a shared MSSQL database, and the CR license manager.

Microsoft Windows Clustering can be configured to create a failover cluster for CR. Clustering provides high availability and high reliability minimizing the CR downtime. SQL Server high availability can be achieved using a SQL Server 2012 AlwaysOn Availability group. There is no single point of failure with always-on SQL Server and Socket Server active / passive configuration. HA/DR is also supported for single forest, multi-domain AD deployment.

Refer Automation Anywhere Failover Cluster configuration guide for details on HA/DR architecture.



**FIGURE 8: HA/DR DEPLOYMENT**



## Security Architecture

AAE security architecture is founded on Least Privilege principles and a strict Separation of Duty model with 41 technical controls implemented across seven NIST Control Families. We've selected the NIST framework as a foundation for best practices as a way to enumerate the controls implemented throughout. Translations from NIST to other control frameworks<sup>1</sup> are widely available, resources are provided at the bottom of this section.

The product security architecture is maintained by Product Management and forms part of a formal policy model as an integral part of our Development Roadmap. The table below enumerates the control families and the corresponding features and security impacts. Each Control family is then discussed in detail in the corresponding chapters with a detailed discussion of how our security architecture is implemented in our products.

| Control Family                    | Control Code      | Control Room Feature  | Security Impact  |
|-----------------------------------|-------------------|---|--|
| Access Controls                   | AC-3,6, 7,9,10,12 | Central Policy Control                                      | Enforce access restrictions for change control and Least Privileges on system components:<br>(1) Fine grained Access to Bots & Bot Runners is controlled via RBAC,<br>(2) Bot and Bot Runner Domains can be assigned to Roles via RBAC,<br>(3) RBAC Roles are fully audited    |
|                                   | AC-2,3,5,6        | Role-based Access Control                                   | Enables user access, restricts operational privileges, enforces least privilege principals   |
|                                   | AC-17             | Bot Repository  | Bot versioning system with access restrictions   |
|                                   | AC-3,7,9,10,11    | Bot and Bot Runner Encryption                               | Encryption and obfuscation of sensitive information at Bot level through Credential Vault and integration with Key Management systems  |
| Configuration (Change) Management | CM-2,5,6,7,9      | Centralized Bot Runner Control                              | Restrict functionality based on roles, domains, implement deny-all and allow-by exception  |
|                                   | CM-10             | Centralized Licensing                                       | Centralized provisioning, tracking and enforcement of Bot Dev and Bot Runner licensing   |
|                                   | CM-2,5,6,8        | Bot Operations Room   | Maintains centralized inventory control of all Bots and Runtimes   |
|                                   | CM-8              | Inventory Control   |  |
| Dev Config Management             | SA-10             | Bot Creator Management, Bot Check-in, Check-out             | Control Room applies software Life Cycle management to Bots from Dev, Test and Prod. Bot versioning enables change control of automations.   |
| Audit & Accountability            | AU-1 thru 15      | Audit Trail   | Automated event logs captured on three levels: Control Room, Bot Runners and Dev Clients. Non-repudiation is assured through read-only logs, all user identities are bound to actions, and   |
| Identification and Authentication | IA-1 thru 5       | Active Directory integration, Bot runner ID and Attestation | Implements Windows platform security including cryptographic bidirectional authentication, Bot runner identification and attestation, and password management policies. Credential Vault, with integration with Key Management systems, protects the integrity of credentials. |
| Incident Response                 | IR-4,6            | Incident Response   | Bot Insight embedded analytics capabilities can monitor events and generate alerts to SIEM systems for response.   |
| Controlled Maintenance            | MA-2              | Automated Maintenance                                       | Control Room versioning systems provides an automated mechanism to roll out updates to Bots, historical information is maintained.   |

(1) Resources: ISACA provides guides that map NIST SP800-53 to other security frameworks such as CoBIT (SOX), SANS Top20 (<http://www.counciloncybersecurity.org/critical-controls/tools/>) and ISO27002 (<http://www.bankinfosecurity.in/mapping-nist-controls-to-iso-standards-a-7251>).

## Access Control

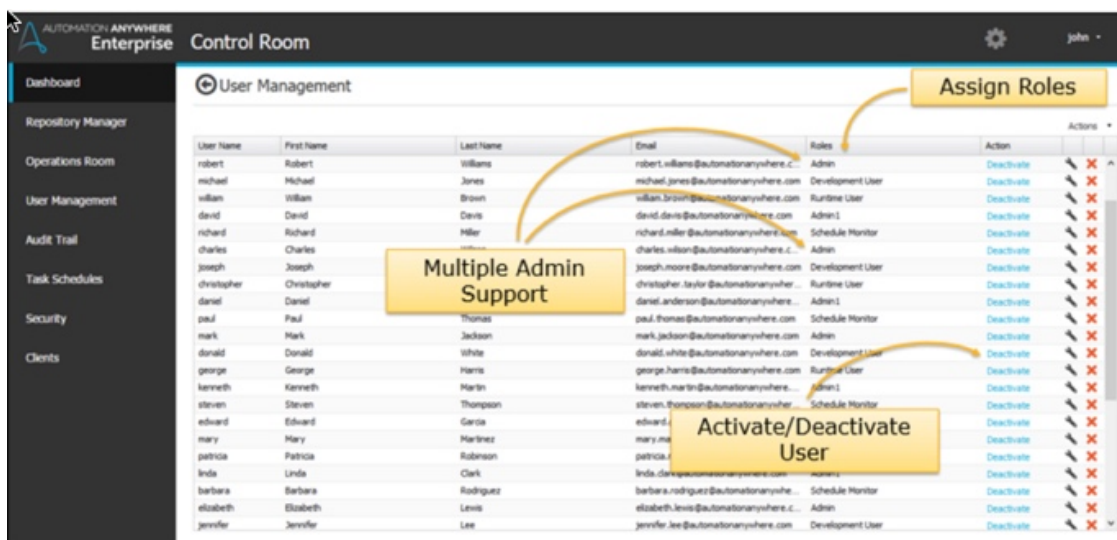
### Independent Control Planes: Bot Creators and Bot Runners

For logical separation of duties, CR divides automation users into 2 broad categories: Bot Creators (development) users and Bot Runners (runtime) users. Bot Creators exist on a separate windows system with its own credentialing system and can create, update and unit test the Bots on the Bot Creator. Bot Creators can only upload and download Bots to and from the versioning system on the CR. Users on the CR may be granted privileges to execute bots on Bot Runners but have no access to the Bot Creators. This separation of duty constitutes a dual authorization by requiring both the developer and the business user to create and execute the bot in conformance with NIST AC-3 best practices.

### Role Based Access Control (RBAC)

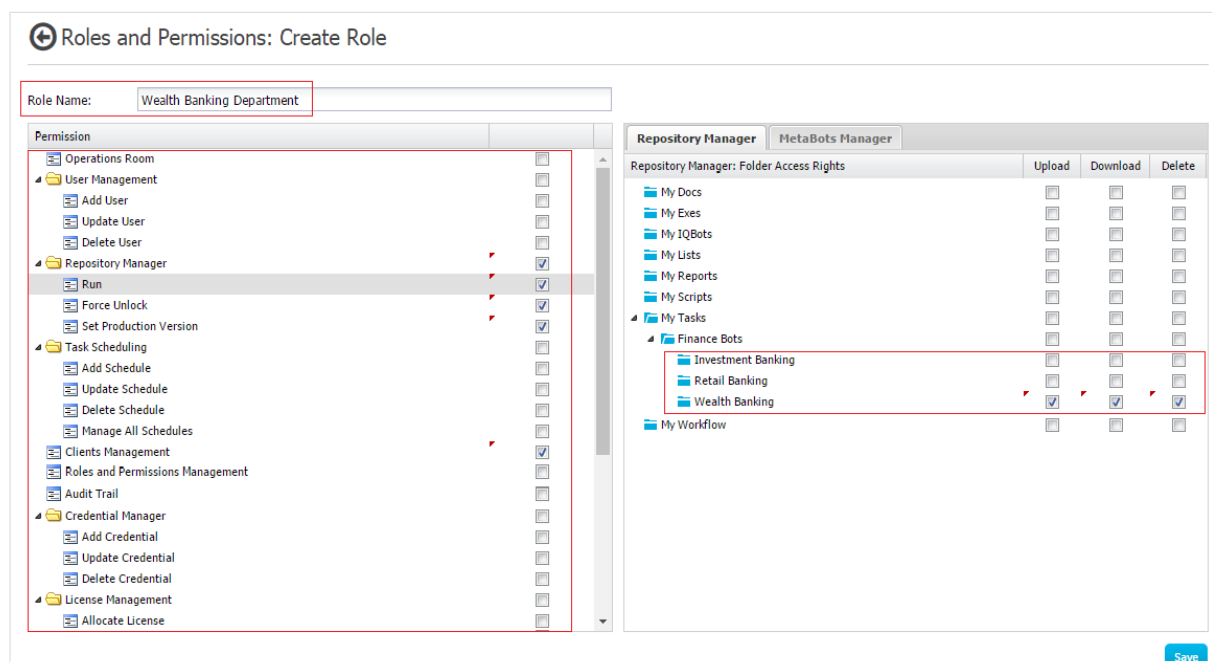
Control Room implements Least Privileges and Separation of Duties through a configurable Role Based Access Control (RBAC) capability that conform to requirements in NIST AC 2, 3, 5 and 6. All CR users must be assigned one or more roles. Access available based on the usage conditions assigned to each role that a user is a member of. Authorized users can also temporarily or permanently suspend users, per business needs. RBAC enforces session handling to prevent any unauthorized use. If an unauthorized user attempts to view session details or to gain unauthorized access, the CR will not allow the user to proceed and will immediately terminate unauthorized user's session. This user will be forced to login with his/her own credentials again. Inactive accounts can be disabled.

The administrator controls is responsible for all security functions consistent with best practices in NIST SC-3: Security Function Isolation. Users can be assigned admin privileges by assigning "Admin" role to them. Admin users can see all the data in CR, in our next release (v11) we will further segmenting administrative roles for separation of administrative tasks.



| User Name   | First Name  | Last Name | Email                                     | Roles            | Action     |
|-------------|-------------|-----------|---|------------------|------------|
| robert      | Robert      | Williams  | robert.williams@automationanywhere.com    | Admin            | Deactivate |
| michael     | Michael     | Jones     | michael.jones@automationanywhere.com      | Development User | Deactivate |
| william     | William     | Brown     | william.brown@automationanywhere.com      | Runtime User     | Deactivate |
| david       | David       | Davis     | david.davis@automationanywhere.com        | Admin            | Deactivate |
| richard     | Richard     | Hill      | richard.hill@automationanywhere.com       | Schedule Monitor | Deactivate |
| charles     | Charles     | Wilson    | charles.wilson@automationanywhere.com     | Admin            | Deactivate |
| joseph      | Joseph      | Moore     | joseph.moore@automationanywhere.com       | Development User | Deactivate |
| christopher | Christopher | Taylor    | christopher.taylor@automationanywhere.com | Runtime User     | Deactivate |
| daniel      | Daniel      | Anderson  | daniel.anderson@automationanywhere.com    | Admin            | Deactivate |
| paul        | Paul        | Thomas    | paul.thomas@automationanywhere.com        | Schedule Monitor | Deactivate |
| mark        | Mark        | Jackson   | mark.jackson@automationanywhere.com       | Admin            | Deactivate |
| donald      | Donald      | White     | donald.white@automationanywhere.com       | Development User | Deactivate |
| george      | George      | Harris    | george.harris@automationanywhere.com      | Runtime User     | Deactivate |
| kenneth     | Kenneth     | Martin    | kenneth.martin@automationanywhere.com     | Admin            | Deactivate |
| steven      | Steven      | Thompson  | steven.thompson@automationanywhere.com    | Schedule Monitor | Deactivate |
| edward      | Edward      | Garcia    | edward.garcia@automationanywhere.com      | Admin            | Deactivate |
| mary        | Mary        | Hartman   | mary.hartman@automationanywhere.com       | Development User | Deactivate |
| patricia    | Patricia    | Robinson  | patricia.robinson@automationanywhere.com  | Runtime User     | Deactivate |
| linda       | Linda       | Clark     | linda.clark@automationanywhere.com        | Admin            | Deactivate |
| barbara     | Barbara     | Rodriguez | barbara.rodriguez@automationanywhere.com  | Schedule Monitor | Deactivate |
| elizabeth   | Elizabeth   | Lewis     | elizabeth.lewis@automationanywhere.com    | Admin            | Deactivate |
| jennifer    | Jennifer    | Lee       | jennifer.lee@automationanywhere.com       | Development User | Deactivate |

FIGURE 9: USER MANAGEMENT MODULE



**FIGURE 10: EXTENSIVE RBAC AVAILABLE ON EVERY MODULE**

Controls are implemented at Control Room, Bot Creators and Bot Runners layers to implement NIST Access Controls (AC) and Change Management (CM) guidelines. Following technical controls are implemented to ensure access is governed through NIST Least Privileges:

### RBAC on Bots

Access is deny-all and allow by exception based on roles, except for admin roles, addressing NIST Access Control AC 17 and addressing NIST Change Management for establishing Base Line Configurations (NIST CM 2), access restrictions for configuration management (NIST CM 5 and 6) and Least Functionality (NIST CM 7) and monitoring Configuration Management for Bot activity across Dev, Test and Production environment of (NIST CM 9).

### RBAC on Bot Runners

Bots can only execute by command from the Control Room. Local Bot execution is prevented and protected through multiple layers of security designed to prevent abuse or tampering resulting from an escalation of privilege on a Windows system executing the Bot Runner addressing Access Control enforcement in accordance with NIST AC-3 Access Enforcement and AC-6 Least Privilege for Code Execution.

RBAC on Bot Runners facilitates complete isolation of one department's Bot Runners from rest of the departments' Bot Runners in a seamless manner. If a user role doesn't have access to a set of Bot Runners, the user will never see those Bot Runners when running/scheduling an automation on remote Bot Runners (See Role-based Processing Domains below).

### RBAC in Credential Vault: Credentials Management

Credentials are created from CR and are used across Bot Creators and Bot Runners. These credentials are securely stored in the centralized Credential Vault. Credential Management access is deny-all and allow by exception based on roles, domains as defined in RBAC.

This permission is further divided into following sub-permissions:

**Add Credential:** Only those users who have this permission can create new credentials from CR.

**Update Credential:** Only those users who have this permission can update existing credentials from CR.

**Delete Credential:** Only those users who have this permission can delete existing credentials from CR

Authorized users can assign various permutation and combinations of these accesses to different sets of users and roles, per the business need.

### Role-based Processing Domains

The Control Room RBAC applies least privilege principles to Domains by implementing Processing Domains, specifying role-based privileges and permissions at the Bots and Bot Runners level. RBAC is applied at a folder level to completely isolate one department's bots from rest of the departments' bots in a seamless manner. If user role doesn't have access to a set of bots, those bots will not exist for that particular user, this enables the separation of duties across different domains. For example, Finance and Accounting roles will have access only to Bots that automate Finance and Accounting functions and to specific Bot Runners that can execute these Bots. This is consistent with best practices as defined by NIST AC-4 Processing Domains.

This permission is further divided into following sub-permissions:

- **Run:** Only those users who have this permission can run the Bots from CR to remote Bot Runners.
- **Set Production Version:** Only those users who have this permission can mark a particular version of Bot as a production-ready version.
- **Force Unlock:** Only those users who have this permission can unlock a bot which is checked out by some user for editing.

Authorized users can assign various permutation and combinations of these accesses to different sets of users and roles, per the business need.

### RBAC on Audit Log

Audit is automated for all privileged and non-privileged roles to conform to best practices as defined in NIST AC-6. Access is view-only based on a deny-all and allow by exception based on roles and domains as defined in the Audit section 7 addressing Audit and Accountability (NIST AU 1 thru 15) and as required by NIST AC-2 Automated System Account Management. If a role does not have permission to view Audit Logs, "Audit Trail" tab will not be visible to all members of those roles. Audit automatically captures all events related to creation, modification, enablement, disablement and removal of users, bots, bot creators and bot runners. See Section 7 for a more detailed discussion on Audit Logs.

## RBAC on Operations Room

Operations Room display the real-time status of the automation running across the enterprise. Access to Operations Room is deny-all and allow by exception based on roles, domains as defined in RBAC. Two levels of checks are applied to access Operations Room data. First, user must be a member of a role which has access to view Operations Room in CR. For users who have got Operations Room access, they can only view the Bots belonging to their departments (as applied through RBAC on Bots).

## RBAC on User Management

Access is deny-all and allow by exception based on roles, domains as defined in RBAC. Only those users who have access to User Management will be able to manager users in system.

This permission is further divided into following sub-permissions:

**Add User:** Only those users who have this permission can create new users from CR.

**Update User:** Only those users who have this permission can update existing users from CR.

**Delete User:** Only those users who have this permission can delete existing users from CR

Authorized users can assign various permutation and combinations of these accesses to different sets of users and roles, per the business need.

## RBAC on Roles and Permissions Management

Access is deny-all and allow by exception based on roles, domains as defined in RBAC. Only those users who have access to “Roles and Permissions Management” will be able to create, update and delete roles in system. This permission is typically assigned to admins and power users from across the departments in an enterprise.

## RBAC on Bots Schedules

Access is deny-all and allow by exception based on roles, domains as defined in RBAC.

This permission is further divided into following sub-permissions:

**Add Schedule:** Only those users who have this permission can create new users from CR.

**Update Schedule:** Only those users who have this permission can update the schedules created by them.

**Delete Schedule:** Only those users who have this permission can delete the schedules created by them.

**Manage All Schedules:** Only those users who have this permission can manage (update, delete) all the existing schedules created by any user.

Admin can assign various permutation and combinations of these accesses to different sets of users and roles, per the business need.

## RBAC on License Management

Access to License Management is deny-all and allow by exception based on roles, domains as defined in RBAC. Only those users who have access to License Management permission can update the license from the CR. There is a common license for all the users across the enterprise, for a given CR. Updated license will be effective for all the Bot Creators and Bot Runners, registered with the corresponding CR.

## Secure Application Partitioning

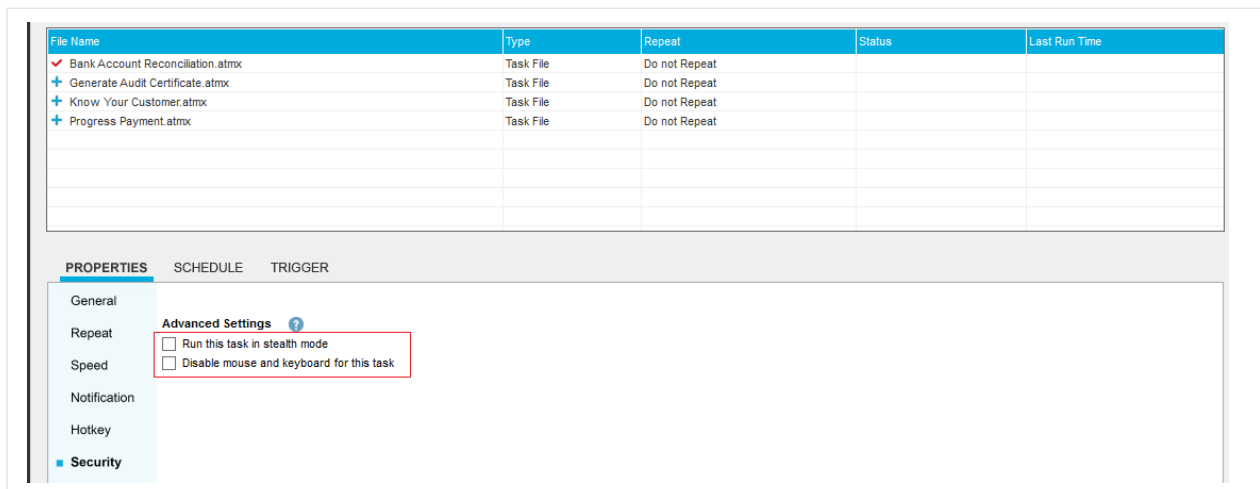
AAE provides security options to enable and enforce secure Bot execution consistent with best practices under NIST SC-2: Secure Application Partitioning. This is essential to the execution of automations in both attended or unattended

mode by preventing unauthorized access to confidential information or unauthorized tampering with the execution of Bots on Virtual Desktops or physical machines.

### Secure Application Partitioning: Stealth Mode for automation

AAE enables the option to run Bots in “stealth mode” where a logged in user would be unable to view the ongoing automation. The business application and any program windows are not displayed on the screen, ensuring confidentiality and privacy. In addition, Stealth Mode ensures that the user would have no control over the running automation, including disabling any ability to pause the automation, stop the automation or see the progress of the automation. This security capability addresses NIST SC-2 best practices for secure application partitioning.

This provides additional safeguards to a bot when a Bot Runner machine is running in (1) attended more (shared between a bot and a human) or (2) unattended where the Bot is executing on a VM. The risk in Unattended mode is from the Virtual Infrastructure Administrator who may have access privileges to the VMs executing the automation and may be able to view or record processes and thereby subvert other controls. Stealth mode eliminates any chances of unauthorized access and tampering with automations.



| File Name                          | Type      | Repeat        | Status | Last Run Time |
|------------------------------------|-----------|---------------|--------|---------------|
| ✓ Bank Account Reconciliation.atmx | Task File | Do not Repeat |        |               |
| + Generate Audit Certificate.atmx  | Task File | Do not Repeat |        |               |
| + Know Your Customer.atmx          | Task File | Do not Repeat |        |               |
| + Progress Payment.atmx            | Task File | Do not Repeat |        |               |

PROPERTIES

SCHEDULE

TRIGGER

General

Repeat

Speed

Notification

Hotkey

Security

Advanced Settings ?

☐ Run this task in stealth mode
 ☐ Disable mouse and keyboard for this task

**FIGURE 11: SECURITY SETTINGS**

### Secure Application Partitioning: Disable mouse and keyboard

Automation Anywhere Enterprise system enables the option to separate user functionality, specifically user interface services such as keyboard and mouse, to prevent unauthorized tampering when bots are running. This makes automation tamper proof ensuring the integrity of automation data. Like “stealth mode”, this provides additional safeguards to bot when a Bot Runner machine is shared between a bot and a human. For example, when this setting is on, user cannot input unwarranted texts anywhere (e.g. entering different credentials) or click on the places where user is not supposed to click. This security capability addresses NIST SC-2 best practices for secure application partitioning.

### Secure Application Partitioning: Configurable automation timeout

Automation Anywhere Enterprise system enables the option to configure the time-period after which a bot will be terminated if it has not finished its execution. It protects the Bot Runner system against any unauthorized usage when a running Bot can be paused by some unauthorized user and that user may try to run the automation with tampered data. It also saves the system memory when Bot is not able to finish execution for some reason. This security capability



addresses NIST SC-2 best practices for secure application partitioning and NIST SI-7 Time Limits on Process Execution without Supervision.

### Secure Application Partitioning: Centralized control on automation running remotely

Automation Anywhere digital workforce run across the enterprise, on remote Bot Runner systems. All the running automations are centrally managed only by authorized users from CR. Running Automation Anywhere bots can be centrally paused, resumed or completely stopped from the centralized CR. This security capability addresses NIST SC-2 best practices for secure application partitioning.

### Bot Execution Access: Dynamic Access Token

The CR implements and enforces a Trusted Path for registration and authentication of Bot Creators and Bot Runners in accordance with NIST SC-11 to protect against any attempt to execute unauthorized Bots. The CR issues new client access tokens, or identifiers thru hashing and signed by CR and sent to Bot Creators and Bot Runners over HTTPS. Every subsequent communication between CR and Bot Creator/Runner is serviced by CR only after validation of signature of latest access token sent by Bot Creator/Runner. Each access token is unique to every Bot Creator/Runner. This ensures that even if some unauthorized user could bypass enterprise security and able to access the system somehow, CR security will restrict any damage. This is discussed in greater detail in the Identification and Authentication section 5.7.

### Secure Credential Store: Credential Vault (CV)

Automation Anywhere platform provides a centralized Credential Vault to securely store all credentials and provision them to bots, on an on-demand basis. The Credential Vault may also be used to store other information deemed confidential or sensitive. The credential store implements NIST controls IA-2 to uniquely identify and authenticate organizational users (or processes acting on behalf of organizational users).

Since sensitive information need not be stored in Bots or on Bot Runner systems, the CV facilitates a logical separation of credentials from the bots..

CV variables are created from CR and are instantly available to all the Bot Creators and Bot Runners registered with the respective CR. CV adds flexibility and dynamic character to bots since only the credential references are present in the bots and not the credentials. When bots are moved from one environment to another environment, absolutely no change is needed in bots. Bots can seamlessly pick up the credentials values applicable for the new environment from CR of that environment. Additionally, CR automatically stores configuration related sensitive data into CV by default.

### Configuration of CV: Encryption

Automation Anywhere Control Room (CR) installation generates following 3 keys

- **Master(Private) key:** is managed by admin user outside of the system. Its only use is to unlock the CV. Admin must enter master key every time CR is restarted. Once the vault is open, the master key is immediately erased from memory and it is not stored anywhere within the AAE product.
- **Public key:** is stored in CR database and is used in conjunction with the master key manually entered by user to unlock the CV.
- **Data encryption key:** is stored in CR database along with public key and is used to encrypt and decrypt the credentials at the time of storage or provisioning. This key itself is encrypted using the master key. The data encryption key does not leave the credential vault at any time. Credential encryption and decryption are done at the credential vault.

To meet NIST IA-5 *PKI-based authentication requirements*, CR employs a *deliberate methodology to manage the content of PKI*. The CV gets automatically locked every time CR service is down. When CR service is started again, CV must be unlocked through a combination of public and private key. This ensures that there is no single point of failure, even when someone hacks into the enterprise database and gets access to the public key, credential security is not compromised since private key is stored externally.

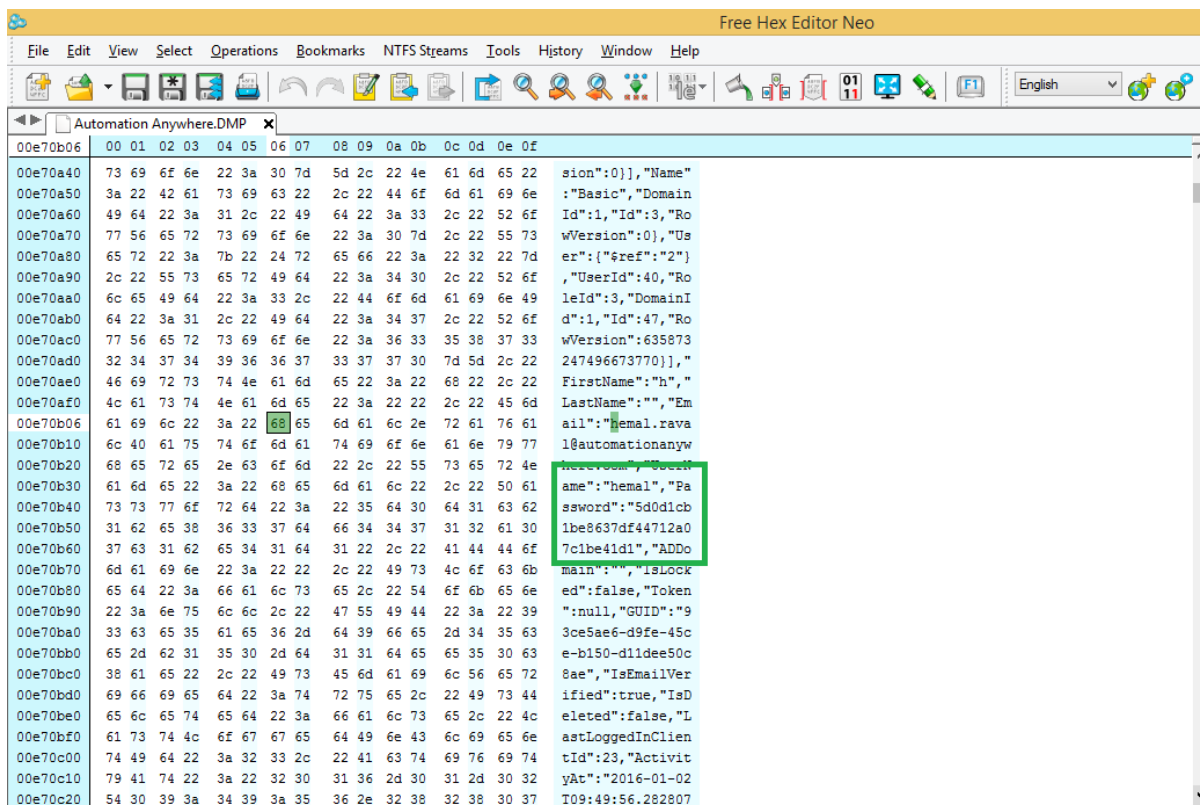
### Credential storage in CV

All credentials are created and managed through CR user interface. These credentials are encrypted to AES-256 bits format by Credential Vault Service to conform to NIST SC-28 to prevent unauthorized access or disclosure of credentials. Only encrypted credentials travel from the CR to the Database server and are stored in database in encrypted form. The data encryption key encrypts all credentials using AES 256-bit key using a FIPS 140-2 Level 1 validated algorithm available from MS Windows to meet the NIST IA-7, SC-12 and 13 requirements for implementation of mechanisms for authentication to a cryptographic module that meet the requirements of applicable federal laws. Please see the [list of cryptographic providers](#) in the appendix for more details.

### Credential Provisioning to bots

Bots Runners or bots do not store credentials locally. Credentials are only provisioned during execution of automation. When the credentials are requested by Bot Runners, encoded (64 bit) credentials travel from CR to Bot Runner over HTTPS protocol. As soon as the credentials usage is over, credentials are encrypted using Microsoft Data Protection API, which uses AES-256 bits encryption. Only encrypted credentials stay in system memory. This ensures that memory mapping tools cannot hack the credentials from system memory. When the bots finish execution, credentials are erased from the memory.





**FIGURE 12: ENCRYPTED PASSWORD IN SYSTEM MEMORY**

## Security at-rest

### Credentials storage in CV

All sensitive data is stored in Credential Vault using AES-256 encryption. Along with the credentials created from CR, following data is also encrypted and securely stored in CV

- User AD credentials for auto-login to Bot Runners are stored in CV
- Connectivity details for Version Control System
- Connectivity details of SMTP server

### PGP Command

Automation Anywhere “PGP” command adds security to your unsecured data by encrypting it. Encryption and decryption is done using “Passphrase” or with public/private keys. Symmetric algorithm is used for both the options. Users can use choose to use any of the symmetric algorithm out of the 8 supported algorithms, including AES-256.

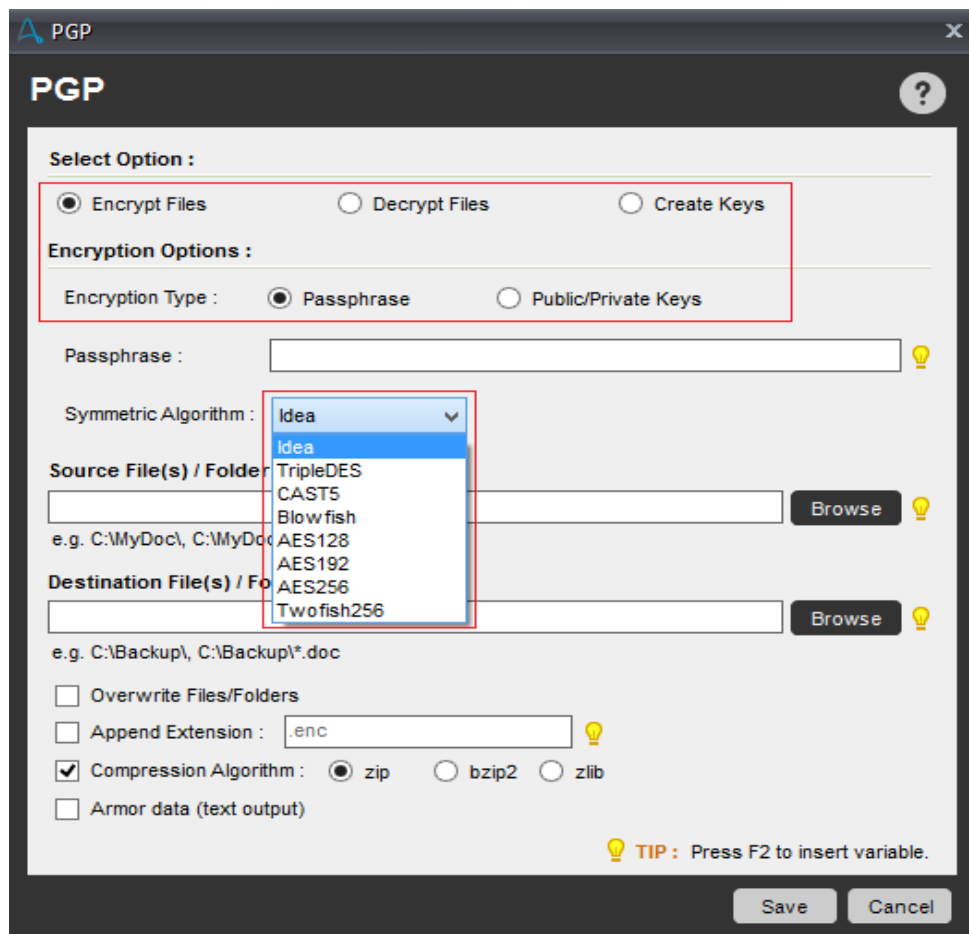


FIGURE 13: PGP COMMAND

## Secure Recording

Automation Anywhere platform allows enabling or disabling of capture of images of business applications, at the click of a button. When secure recording is enabled, no Bot Creator or Bot Runner can capture the application images or values or texts. This results into no sensitive data, intentionally or unintentionally, being stored within bots, in the form of images. Rest of the automation data (such as UI object details are captured) continues to be captured and automation works seamlessly. This setting is applied from CR and is applicable for all Bot Creators and Bot Runners across the enterprise.

## Protection of software binaries

All binary files which are installed with our platform are digitally signed, obfuscated and protected using Intellilock licensing mechanism. This adds to security at binary level. It also allows product files to be not detected as a virus by your enterprise anti-virus.

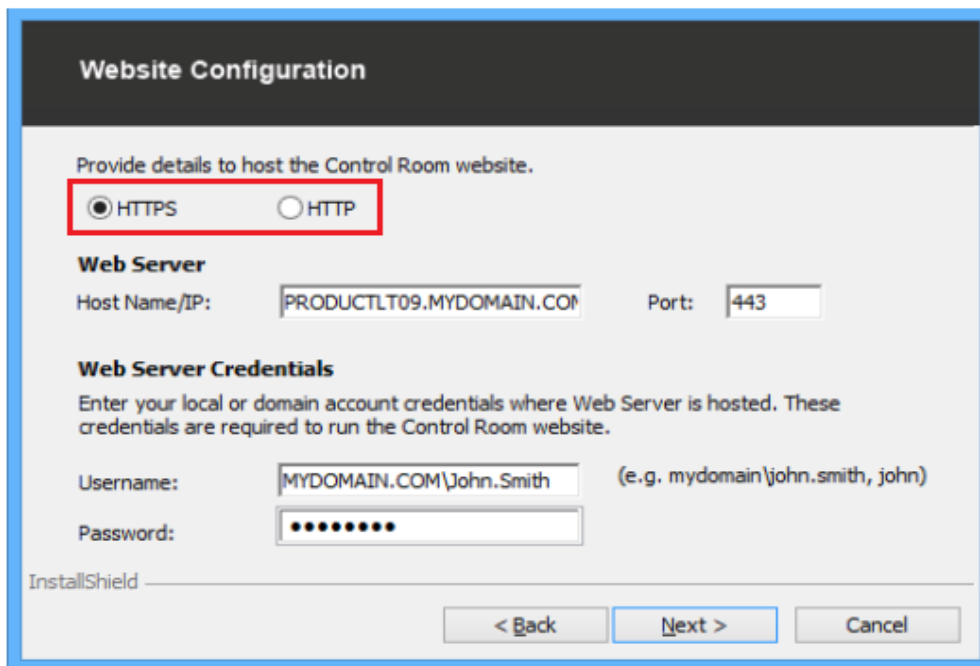
## Security in-transit: Support for secure protocols

Automation Anywhere platform supports secure protocols such as TLS 1.2, HTTPS, WSS data transfer.

- Deployment of bots from CR to remote Bot Runners is done over TCP + TLS 1.2
- Upload and download of bots from Bot Creator to CR is done over HTTPS.

- Transfer of any information from CR to database and vice versa is done over TDS + TLS 1.2
- Transfer of encoded credentials from CR to Bot Runners is done over HTTPS
- WebSocket communication with the real time data service in CR is done over WSS

User can configure to choose the communication protocol between CR server and clients. We recommend using HTTPS in a production environment.



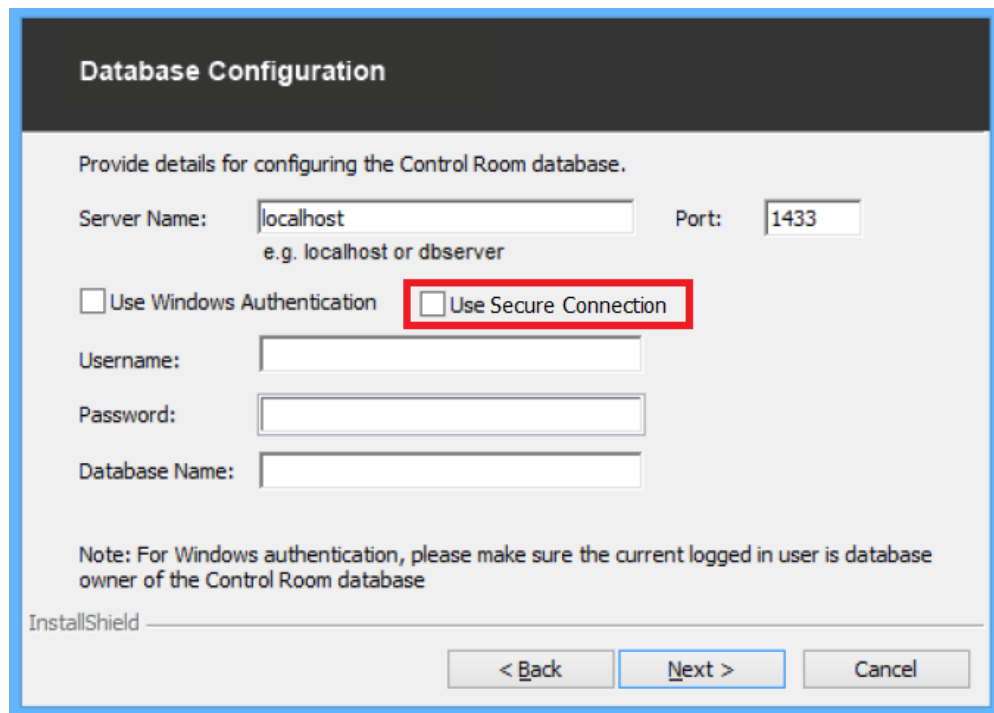
**FIGURE 3: PROTOCOL CONFIGURATION DURING CONTROL ROOM INSTALLATION**

## Authentication with Control Room

When a Bot Creator or Bot runner tries to connect to Control Room, the user's credentials are encrypted using AES (256 bits key length) and RSA (2048 bits key length) and then transmitted to Control Room, on top of the existing layer of TLS. These credentials are decrypted by Control Room and authenticated against the hashed (SHA256 algorithm) user passwords.

## Communication between CR and database

Automation Anywhere Control Room allows users to configure to secure connection between Control Room and database. Secure connection encrypts all communications between Control Room and database and provides additional security to all data exchanged.



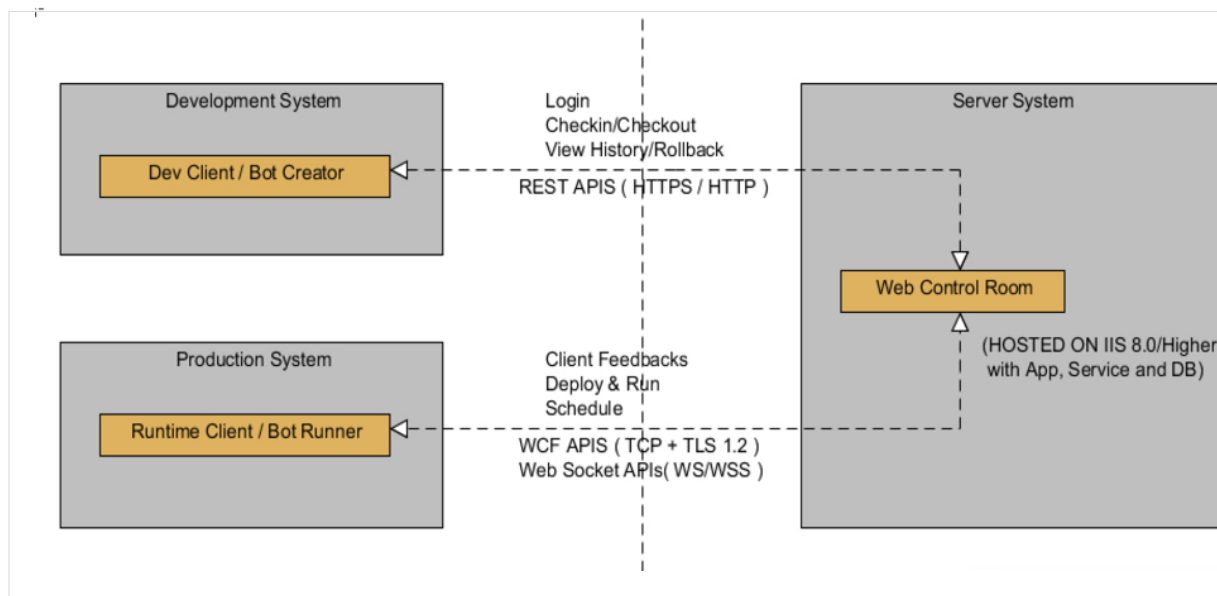
**FIGURE 4: SECURE COMMUNICATION BETWEEN CR AND**

## Password Hashing

Users' CR passwords are concatenated with a salt and then hashed using the SHA256 algorithm before being stored into the database. Every time a Bot Creator or Bot Runner authenticates against CR, its credentials are authenticated against the hashed credentials. All algorithms are FIPS 140-2 Level 1 compliant. Password hashing performs a one way, permanent transformation of the passwords of CR users, in line with standard password management practices

## Network Security Overview

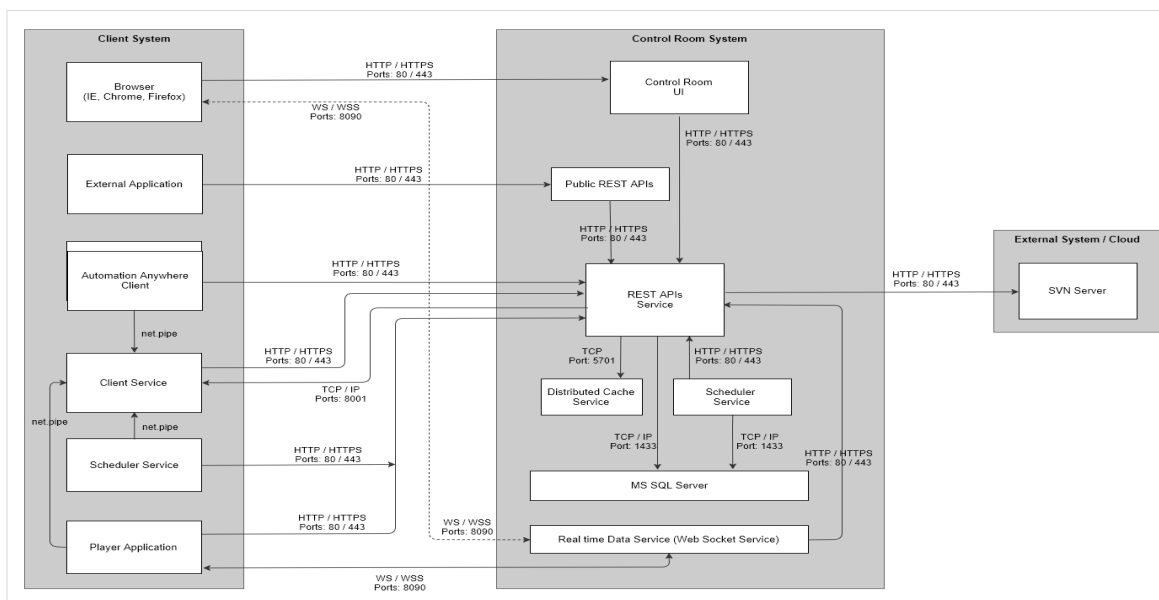
All communication between the Control Room, Bot Creators and Bot Runners is performed using outbound WCF TLS 1.2 communications and inbound HTTPS TLS 1.2. Bot Deployment to remote bot runners, provisioning of credentials, automation scheduling and event capture are performed exclusively through the CR. It requires only standard HTTP & HTTPS ports, no additional ports or services are enabled. We recommend only HTTPS to be used in production environment. Below is the Web Control Room architecture and network diagram.



**FIGURE 5: NETWORK COMMUNICATION**

TLS Communication between CR and other components can be turned into domain based transport layer security by setting the appropriate flags.

Below Figure explains component level communication channels with protocols and ports. Ports and protocols are same for standalone and distributed installations except the Cache service.



**FIGURE 67: NETWORK PROTOCOLS AND**

- REST APIs uses Distributed Cache Service to get shared cached data which is required for specific functionality. It also accesses SVN server for various task versioning activities like check-in/out, get latest, rollback etc.

- Scheduler Service makes REST API calls to run a task on a specific client machine at specific time. It also uses database (MS SQL Server) to get details about task scheduled for clients.
- Real time Data Service makes REST API calls to authenticate incoming connection requests. It receives task execution progress updates by Bot Runners and sends that information to all connected browser clients using WSS protocol.
- Automation Anywhere Client makes REST calls for user authentication and some repository operations like upload a task, download a task or compare two tasks.
- Client Service makes REST calls to validate user session at some regular interval. Control Room deploys and runs a task on a specific client using Client Service. It uses a TCP/IP channel.
- Scheduler service makes REST calls to get the machine's credential in order to perform an auto-login into the system. It also communicates to the Client Service to get license and user session related information.
- Player makes REST calls to get auto-login credentials for a logged in client. It also communicates to Client Service to get license and user session related information.

### Secure Protocols in use

Following table contains list of the protocols Automation Anywhere Enterprise products utilize to enable various task automation

| Sr. No | Protocol       | Deployment            |
|--------|----------------|-----------------------|
| 1      | SNMP           | Client                |
| 2      | IMAP           | Client                |
| 3      | FTP / SFTP     | Client                |
| 4      | POP3           | Client                |
| 5      | HTTP/HTTPS     | Client & Control Room |
| 6      | WS / WSS       | Client & Control Room |
| 7      | TCP/IP         | Client & Control Room |
| 8      | TLS            | Control Room          |
| 9      | SMTP           | Client & Control Room |
| 10     | SOAP           | Client                |
| 11     | Named Pipes    | Client                |
| 12     | NTLM / NTLM v2 | Control Room          |

**FIGURE 78: LIST OF PROTOCOLS**

## List of Port Numbers

Below table lists the configurable ports used by our platform

| Port             | Description       | Deployment             | Used for  |
|------------------|-------------------|------------------------|---|
| 110<br>995       | POP3              | Client                 | "Email Automation" command to retrieve emails from mail server.   |
| 143<br>993       | IMAP              | Client                 |   |
| 21               | FTP/SFTP          | Client                 | "FTP/SFTP" command  |
| 25<br>465<br>587 | SMTP              | Client                 | Client:<br>"Send email" command<br>"Error handling" command<br>"Email notification" feature<br>Control Room:<br>Send email when user created, password set/reset, role changed etc. |
| 161              | UDP               | Client                 | "SNMP" command  |
| 22<br>23         | Terminal Emulator | Client                 | "Terminal Emulator" command   |
| 80               | HTTP              | Client<br>Control Room | Hosting of Control Room in IIS (Not recommended)  |
| 443              | HTTPS             | Client<br>Control Room | Hosting Control Room in IIS (Recommended)   |
| 8001             | TCP/IP            | Client<br>Control Room | Client Service to receive deploy/run/schedule requests from Control Room.   |
| 8090             | WS / WSS          | Client<br>Control Room | Operation room data and pause/resume/stop actions.  |

|       |        |  |  |
|-------|--------|--|--|
| 1433  | TCP/IP | MS SQL Server                            | Default port used by MS SQL Server   |
| 4530  | TCP/IP | Client (AAProxyServer.exe)               | It's used to communicate between client to respected plugins via TCP socket for AAE Client, Editor or Player |
| 5701  | TCP/IP | Control Room (Distributed Cache Service) | Control Room uses to connect to Cache Service.   |
| 54327 | TCP/IP | Control Room (Distributed Cache Service) | Uses for distributing cache to other cache service instances.  |

**FIGURE 89: LIST OF PORT NUMBERS**

## Change Management

### Versioning and Operational Control:

From a security perspective, the versioning system previously described establishes a Base Line Configurations (NIST CM 2), access restrictions for configuration management (NIST CM 5 and 6) to deliver controlled maintenance (NIST CM 8) and Development Configuration Management (NIST SA 10). The Bot Creator performs check-in, check-outs into the Control Room-based VisualSVN system. The Control Room enforces version control as Bots are deployed to Bot Runners. The versioning system in the Control Room maintains a common Bot baseline, with detailed configuration logs, and roll-back capability. This is commonly used for access restrictions to the Bot Repository for change control and restricting Least Privileges for operational control.

### Baseline Inventory Controls: Bot Creators, Bot Runners and Bots

The CR provides a single-pane-of-glass on all automation operations and infrastructure providing a way to baseline the configuration of the environment. Inventory controls are maintained through the application of RBAC and the use of the Bot Repository, Operations Room and License Management to establish a single point of control for Base Line Configurations (NIST CM 2), access restrictions for configuration management (NIST CM 5 and 6). Automated baseline reporting can be configured using the auditing and reporting systems in the CR.

### Change Control & Documentation: RBAC

The CR RBAC provides a point of access control and management for all changes to CR, CV, Bot Creators, Bot, and Bot Runners with an automated mechanism to prohibit changes and report on any attempts to make unauthorized changes. The logging and auditing system on the CR provides the reporting mechanism for change management to conform to best practices as described in NIST CM-3 through 5.

### Change Control: Automated Configuration Changes

The Bot Creator Metabot capability can be utilized to update all bots with any authorized automated implementation changes required to conform to change management best practices as described in NIST CM-4.



## Software Usage and License Management

The Control Room provides an automated mechanism for tracking and controlling the use of licensed software across Bot Creators and Bot Runners, addressing NIST Change Management CM 10.

## Change Management: Dual Authorization

Separation of duties is implemented at multiple levels. Dual authorization is achieved thru separation of control planes for the Bot Creator and Bot Runners. Only bots created by an authorized Bot Creator can be executed by a separately authorized Bot Runner. And only then by a User who has been explicitly given privileges to do so by a separate administrator.

## Identification and Authentication

All automation actions (create, view, update, deploy, delete etc.) across the enterprise can be performed only after CR authentication is successful completed. Once authentication is successful, platform applies a second mandatory level of access control enforcement in the form of fine grained Role Based Access Control (RBAC) described in Section 4.3.

AAE offers seamless integration with Windows Active Directory for access to the CR, Bot Creators, and Bot Runners. When CR is integrated with AD, all the AD users with basic details are directly available in CR without need of any extra configuration. For AD integration, user passwords stay in AD only and are not saved in our platform.

In addition to AD authentication, CR has its own controls to prevent unauthorized access to any automation data. Refer to the [Dynamic Access Token section](#) 5.7. For CR login, users can configure to authenticate against AD and completely bypass login screen. Bot Runners users can also configure their AD credentials for Bot Runners machine auto-login. These credentials are saved in centralized Credential Vault.

## Multi-domain AD support

Automation Anywhere architecture supports single-forest multi-domain Active Directory integration. CR can be configured with Active Directory Global Catalogue Server in a way that CR, Bot Creators and Bot Runners can all be in same or different AD domains of a single forest. This gives added flexibility and control for large scale complex deployment where users are spread across geographies.

Multi-domain support is provided out of the box and no extra configuration is needed. CR User provisioning from different AD domains is also seamless. It enables CR admin to centrally orchestrate the digital workforce running across the globe.

## User Authentication for Control Room Access

CR users must authenticate against Active Directory for any administration and orchestration purpose. Single Sign On (SSO) can be configured for CR users, against the enterprise Active Directory. In 4Q we will provide additional support for Security Assertion Markup Language (SAML) v2. Using SAML, AAE platform users will be able to perform Single Sign On (SSO) to AAE platform by authenticating against the third-party enterprise systems. When an enterprise configures SSO for our platform, user passwords will not be stored in Control Room.

CR authentication is single-factor and integrates with Active Directory (AD). Login and password management defaults to the AD domain. Additional factors can be implemented via the Azure Multi-Factor Authentication Server available for all

Control Room supported Windows platforms. Additional factors create one more layer of defence against unauthorized access.

When the Control Room is running standalone (no AD integration) it implements automated password policy enforcement to partially conform to NIST IA -5 by providing case sensitivity, number of characters, numbers and special characters. The next release (v11) will implement password complexity, minimum character change on password changes to fully comply with NIST IA-5.

Control Room passwords are secured both [at-rest using CV](#) and [in-transit using HTTPS](#).

## Authentication Failure Messages

In the event of a failed authentication attempt, our platform doesn't specifically state if the username is incorrect or the password is incorrect. It only states that the supplied credentials are incorrect. This is one critical information security requirement for our enterprise customers and defends the system against a brute force attack.

This check is present at all places where authentication is performed, such as

- Bot Creator, Bot Runner connection to CR
- User login to CR from browser
- Connection from CR to SQL Server

All failed authentication attempts are logged, see Section 7 on Audit Logs. Audit Log access is provided as per RBAC and audit logs are made available to user on a read-only basis for all users.

## Auto Log-off

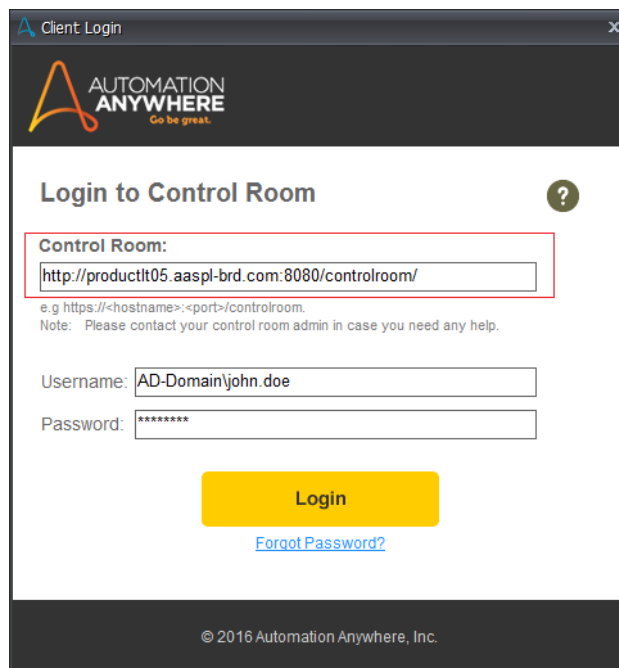
Automation Anywhere platform complies to information security guidelines regarding user session timeout, in case of inactivity. An active session with CR is terminated after pre-defined time interval, when it is not being used. For CR users, they will be automatically logged-off after an idle period of 20 minutes. Similarly, Bot Creator session is terminated after 8 hours.

Bot Creators and Bot Runners have an additional checkpoint in the form of dynamic access token. This ensures that security is not compromised in the event of user not actively using the system.

Refer to the [Dynamic Access Token section 5.7](#) for details on access tokens.

## Authentication for Bot Creators

Bot creators must authenticate against CR for any operations on the Bots. The system obscures feedback of authentication information during the authentication process to protect the information from possible exploitation/use by unauthorized individuals as required by NIST IA-6.



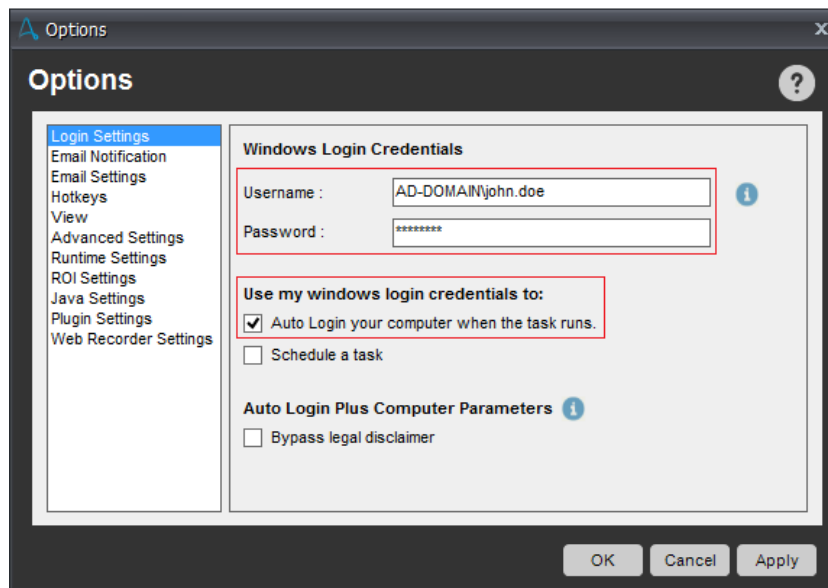
**FIGURE 20: BOT CREATOR AND BOT RUNNER  
AUTHENTICATION**

Users' passwords to connect to Control Room are encrypted during transit, implemented using hybrid security with RSA (2048) + AES (256) + HMAC (SHA256) algorithm.

## Authentication for Bot Runners

Two layers of authentication are present for deploying the bots on remote Bot Runners. First a logged off/disconnected/locked Bot Runner is logged on/connected/unlocked using the configured credentials. These credentials are securely fetched from the centralized Credential Vault over HTTPS. This first level of authentication is performed against the enterprise Active Directory domain. This is done automatically on behalf of the user and is called Bot Runner auto-login. (shown in *Figure 10*). Refer to the [Credential Vault section 3.5](#) for details. The current default authentication utilizes TLS/SSL and is currently based on NTLM, with Kerberos authentication scheduled for release in 4Q of this year for stronger authentication between Bot Creator/Bot Runner and Control Room/Third party systems.

Once first level of authentication is successful, second level authentication kicks in where user must authenticate against CR to run the bot on the respective Bot Runner. Once authenticated, Bot Runners can be authorized to execute Bots independently and asynchronously.

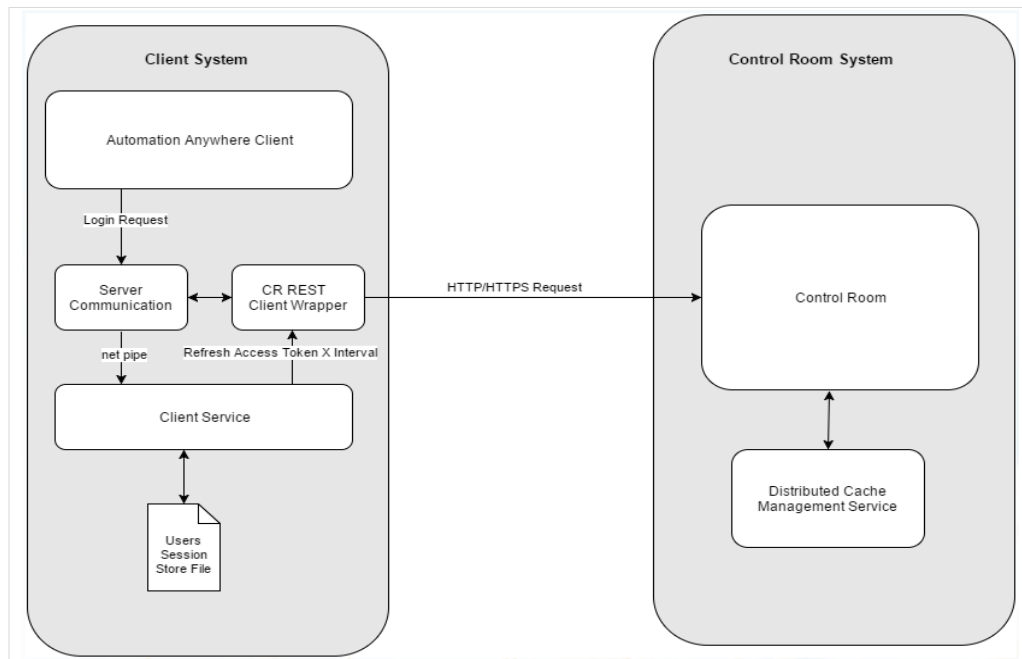


**FIGURE 21: BOT RUNNER AUTO-LOGIN**

## Authentication of Bot Runners: Dynamic Access Token

The CR implements and enforces a Trusted Path for registration and authentication of Bot Creators and Bot Runners in accordance with NIST SC-11. Even after Bot Creators and Bot Runners have authenticated themselves, our platform protects the automation data against any attempt to subvert the path. CR issues new client access tokens, or identifiers, after a pre-defined time interval. These tokens are protected to conform to NIST IA-5 by being signed by CR and sent to Bot Creators and Bot Runners over HTTPS. Every subsequent communication between CR and Bot Creator/Runner is serviced by CR only after validation of signature of latest access token sent by Bot Creator/Runner.

The access token is unique to every Bot Creator/Runner. Access tokens are unique from one Bot Creator/Runner, it cannot be used in any other Bot Creator/Runner. For that single system also, it will be expired after the pre-defined time interval. This protects the system from an unauthorized attempt to bypass enterprise security and execute an unauthorized bot consistent with best practises to conform to NIST IA-9 Service Identification and Authorization. These controls implement IA-3 for cryptographically based bidirectional authentication and attestation of Bot Runners and Bot Creators before establishing connections. It also addresses requirements around unique, automated, identifier management IA-4 for multiple forms of authorization and identification. Identifiers are dynamically managed for audit and control purposes. Identifiers are used as authenticators and managed for verification on initial deployment, revoke, and prevent reuse. There are no static, unencrypted, identifiers in use by Bot Creators or Bot Runners and cached tokens are cleared periodically.



**FIGURE 22: DYNAMIC ACCESS TOKEN**

## Integration with Third-Party Identity and Access Management Solutions.

In cases where a client wants to use their own secure credential store, Automation Anywhere platform supports seamless integration with privileged access management solutions such as CyberArk, TPAM, Thycotic etc. Our extensible MetaBot engine is used to make API calls to any of these privileged access management solutions. Our enterprise customers use MetaBots to securely fetch the credentials stored in CyberArk, TPAM and use those credentials to authenticate against business applications, during execution of automation.

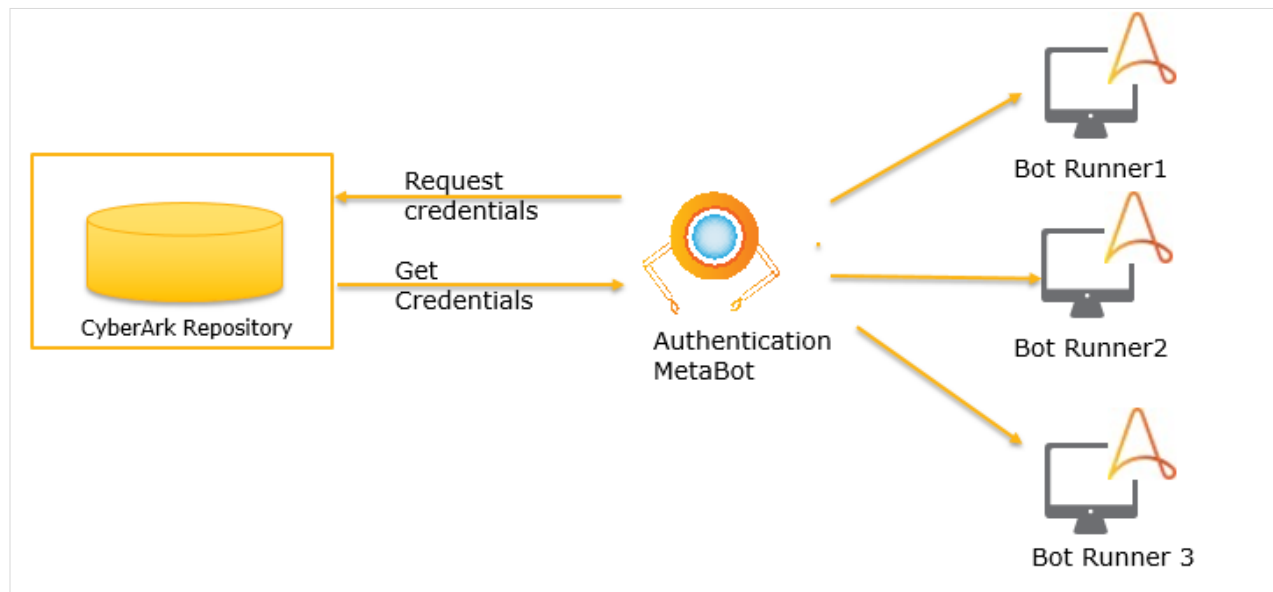


FIGURE 23: CYBERARK INTEGRATION

Extensible MetaBot engine has a special *Execute* type permission which is introduced to handle such specific secure integration scenarios. Such MetaBots are shared with authorized users by providing them *Execute* permission on integration MetaBots. Users with only *Execute* permission on MetaBot can only use such MetaBot as black boxed bots, within TaskBots. By no means can such user see the content of such MetaBot or edit those MetaBots.

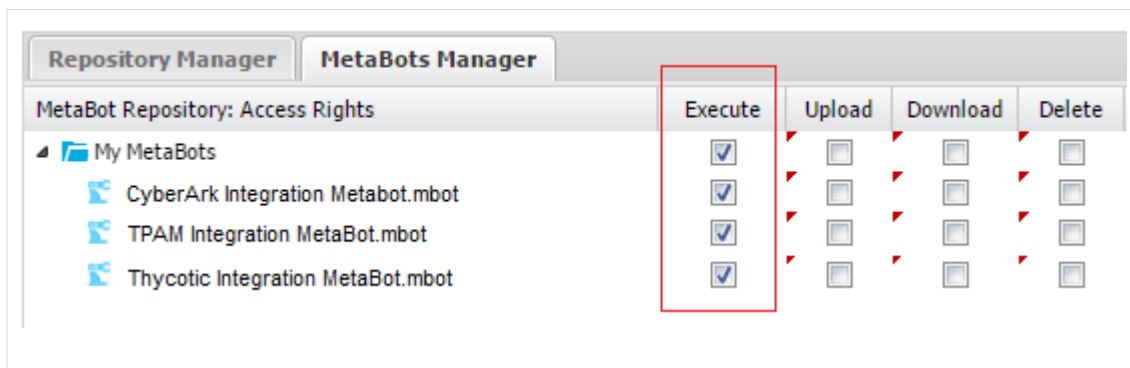


FIGURE 24: EXECUTE PERMISSION TYPE FOR  
METABOT

## Compliance and Scanning

### Windows Logo Certification

Microsoft acknowledges our commitment to quality since our platform meets stringent Microsoft standards on stability, compatibility, reliability, performance, and quality. Our Bot Creators and Bot Runner software can be showcased in the Microsoft Windows Compatibility Centre and we are authorized to display the certification logo on our website.

## Vulnerability

Windows logo certification enables our enterprise customers to host our Bot Creator, Bot Runner product on their internal app stores. This facilitates trusted distribution of software within the enterprise.

Software used for Bot Creators and Bot Runners is Automation Anywhere Enterprise Client 10.5. This software is certified to use Windows logo for following versions:

Automation Anywhere Enterprise Client 10.5 We are certified by Microsoft to use Windows logo for all Windows 7, Windows 8.1 and Windows 10 platforms.

| # | OS Name     | OS Version  | OS Architecture |
|---|-------------|---|-----------------|
| 1 | Windows 7   | Microsoft Windows 7 Professional (6.1.7601.65536) | X64             |
| 2 | Windows 8.1 | Microsoft Windows 8.1 Pro (6.3.9600.0)            | X64             |
| 3 | Windows 10  | Microsoft Windows 10 Pro (10.0.10586.0)           | X64             |

Our Bot Creator and Bot Runner software passes following Microsoft tests to secure logo certification

1. Do not force an immediate reboot during installation
2. Do not force an immediate reboot during uninstallation
3. Write appropriate Add/Remove Program values
4. Single user registry check
5. Install to Program Files
6. Install platform specific files, and drivers
7. Proper OS version checking
8. User account control run level
9. Don't block reboot
10. Do not load services and drivers in safe mode
11. Crashes and hangs
12. Compatibility fixes
13. User mode hooking using AppInit\_DLLs
14. Compatibility manifest
15. Attack surface analyser

## Systems Security Analysis

Automation Anywhere has implemented a Development Security Plan and Protocol that defines a specific depth of testing/evaluation to be performed by the Developer and Quality Assurance teams on each Bot conforming with best practices as defined by NIST SA-11 Developer Security Testing and Evaluation and NIST SA-15, Development Process, Standards and Tools. The Development and QA Teams are trained on the protocols and procedures summarized below.

## Static & Dynamic Code Analysis: Veracode Vulnerability Scanning

On each weekly build during the development process and before every release, all Automation Anywhere software is scanned for flaws using the Veracode tool. Our product currently meets the requirements for Veracode Level 3, which is defined as no “Very High” or “High” level vulnerabilities. We currently have documented less than 80 findings of medium severity. In addition, Automation Anywhere conducts manual code reviews using organization-wide best practices procedure and techniques. These policies conform to NIST RA-5 requirements

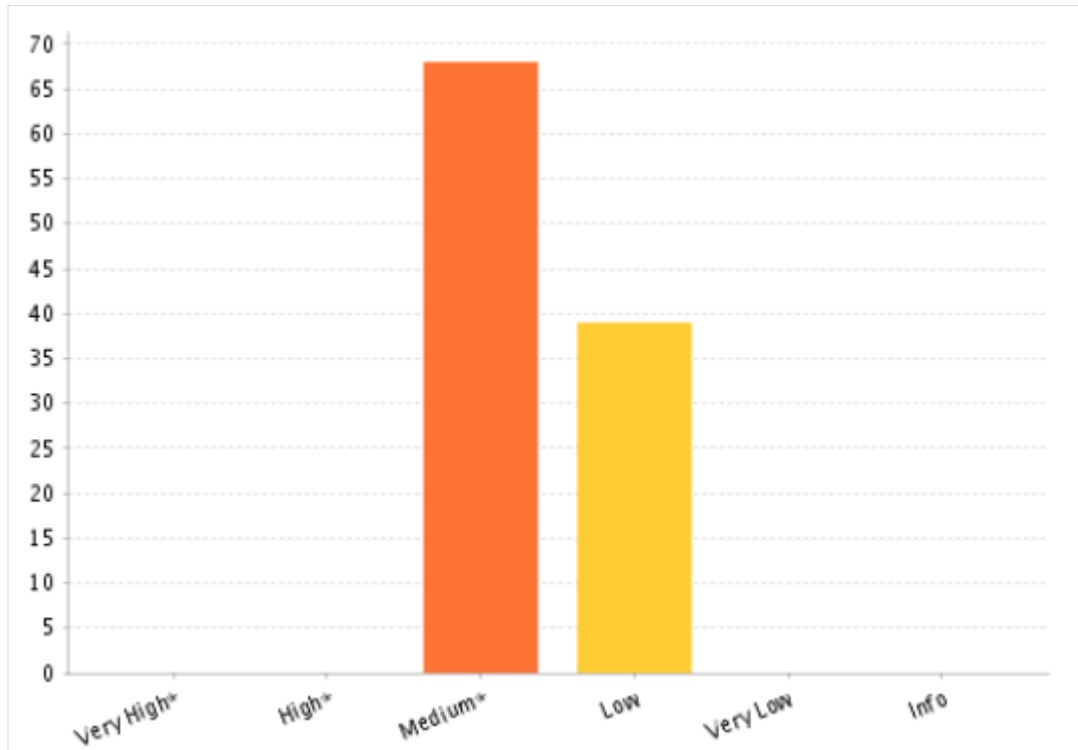


FIGURE 25: SUMMARY OF FLAWS FOUND BY SEVERITY

## Network Vulnerability Analysis: Nessus Vulnerability Scanning

Automation Anywhere platform undergoes automated Nessus vulnerability scanning before every release to identify the vulnerabilities, policy-violating configurations and malware that attackers could use to penetrate Automation Anywhere Enterprise. Results are immediately fed back to the development plan and critical and high vulnerabilities are corrected prior to release. Medium and low vulnerabilities are prioritized based on schedules and resources constraints.

| Results Summary |      |        |     |      |       |
|-----------------|------|--------|-----|------|-------|
| Critical        | High | Medium | Low | Info | Total |
| 0               | 0    | 8      | 2   | 86   | 96    |

FIGURE 26: PRE-SCAN NISSUS REPORT NO “CRITICAL” OR “HIGH” SEVERITY VULNERABILITIES WERE FOUND BY NISSUS IN THE CURRENT RELEASE AS NOTED BELOW



As a matter of policy, Automation Anywhere is committed to make reasonable efforts to mitigate or remediate critical and high vulnerabilities within 30 days of identification on of a new vulnerability on any supported products. These policies conform to NIST RA-5 requirements. These policies conform to NIST RA-5 requirements.

### OWASP ZAP Vulnerability Scanning

OWASP ZAP Vulnerability Scanning helps automatically find security vulnerabilities in web applications during testing. Automation Anywhere conducts a OWASP ZAP scan before every release. Currently, there were no “Critical” or “High” risk vulnerabilities reported. As a matter of policy, Automation Anywhere is committed to make reasonable efforts to mitigate or remediate critical and high vulnerabilities within 30 days of identification on any supported products. These policies conform to NIST RA-5 requirements.

### Penetration Testing

Automation Anywhere performs internal penetration tests and attack surface reviews before each release. The scope of security testing is previewed as part of each release plan. In addition, we benefit from feedback from penetration tests conducted by our customers, which include some of the largest financial institutions in the world. Automation Anywhere intends to contract an independent third-party penetration test for Version 11.

### FIPS 140-2 Compliance

Automation Anywhere credentials encryption utilizes a Microsoft Windows standard FIPS 140-2 Level 1 crypto modules based on the Advanced Encryption Standard (AES) with a 256-bit key. Users’ CR passwords are concatenated with a salt and then hashed using a FIPS compliant SHA256 algorithm before being stored into the database. Please see the [list of cryptographic providers](#) in the appendix for more details.

## Auditing and Logging

### Audit Logs

Automation Anywhere provides a comprehensive and centralized audit logging of all automation activities, to authorized users. Role based access control to Audit Log is managed through Control Room. 60 + actions are audit logged.

All valid and invalid attempts of actions are audit logged. Events are logged by:

- Performer of the action, i.e. Username
- Source of the action, i.e. Bot Runner or Control Room
- What type of event, i.e. event description
- When the event occurred, i.e. date and time
- Where the event occurred, i.e. machine name
- Outcome of the event, i.e. description and status of event

Some of the key audited actions are as below

- Login and logout to centralized Control Room
- Create, Update and Delete Users
- Activation, deactivation of the CR users
- Any change of password for any user

- Create, Update and Delete Roles (helps in tracking changes to security policy, change in user access privileges)
- Create, Update and Delete Schedules
- Connection to Credential Vault
- Create, Update and Delete Credentials
- Set production-ready version of bots
- Deploy bot from CR to Remote Bot Runners
- Pause, Resume and Stop ongoing automations
- Any upload and download from Bot Creators and Bot Runners
- Any check-in, check-out of bots from Bot Creators and Bot Runners
- Update Email settings, Version Control Settings etc.
- Enable, disable secure recording
- Changes to license
- Create Bot Runner instance on BotFarm, release virtual machine, terminate virtual machine

### RBAC on Audit Log

Audit is automated for all privileged and non-privileged roles to conform to best practices as defined in NIST AC-6. Access is view-only based on a deny-all and allow by exception based on roles and domains as defined in the Audit Section 7 addressing Audit and Accountability (NIST AU 1 thru 15) and as required by NIST AC-2 Automated System Account Management.

If a role does not have permission to view Audit Logs, “Audit Trail” tab will not be visible to all members of those roles. Audit automatically captures all events related to creation, modification, enablement, disablement and removal of users, bots, bot creators and bot runners. See Section 7 for a more detailed discussion on Audit Logs.

### Control Room Bot Creator and Runner Activity Logging

For every Bot Creator and Bot Runner, Automation Anywhere performs a comprehensive activity logging for Bots, Workflows, Reports etc.

Some of the key activities logged are as below

- Task Creation, Update, Deletion (task is a type of bot)
- Task Run
- Workflow Creation, Update, Deletion
- Workflow Run
- Report Creation, Update, Deletion
- Report Run
- Change in bot properties

### Audit of Bot Runner Operations

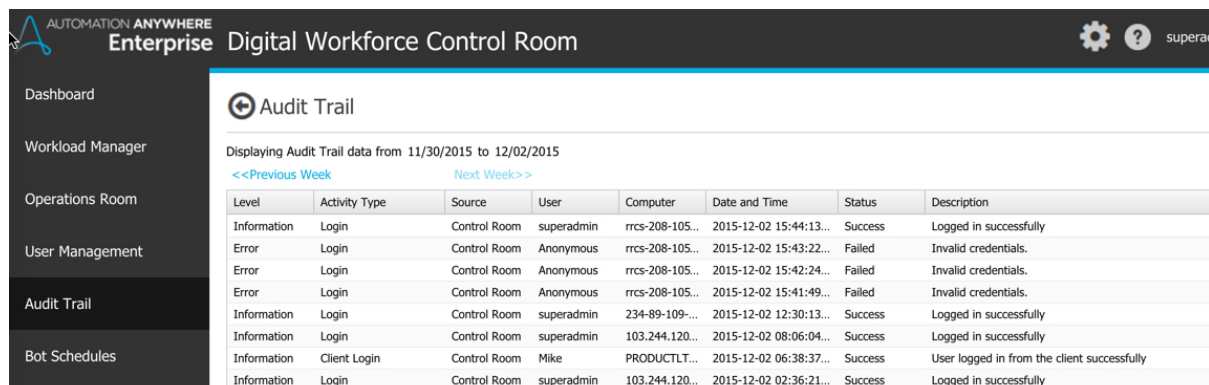
Bot Insight captures additional Bot Runner events for review and analysis of audit records for indications of inappropriate or unusual activity. The Bot Insight logs can be exported for further analysis. Automated dashboard and reports are available and can be customized to identify and alert on anomalous activity. These capabilities conform to best practices as defined in NIST AU-6 Audit Review Analysis and Reporting.

## Audit Log Non-repudiation

The logs are protected against an individual (or process acting on behalf of an individual) falsely denying having performed authorized actions through read-only privileges, automated event capture, and binds the identity of the user to the actions, in conformance with best practices as defined in NISGT AU-10 Non-repudiation and AU-11 Association of Identities.

## Export of Audit Logs

All CR and Bot Insight Bot Runner logs can be exported to a Security Event Information Management Systems for further analysis to support the organizations incident response efforts in accordance with the NIST AU-6 and IR-5 requirements.



The screenshot shows the 'Audit Trail' section of the Automation Anywhere Enterprise Digital Workforce Control Room. It displays a table of audit logs for the period from 11/30/2015 to 12/02/2015. The table includes columns for Level, Activity Type, Source, User, Computer, Date and Time, Status, and Description. The logs show various login attempts, including successful logins for 'superadmin' and 'Mike', and failed logins for 'Anonymous' users.

| Level       | Activity Type | Source       | User       | Computer        | Date and Time          | Status  | Description                                 |
|-------------|---------------|--------------|------------|-----------------|------------------------|---------|---|
| Information | Login         | Control Room | superadmin | rrcs-208-105... | 2015-12-02 15:44:13... | Success | Logged in successfully                      |
| Error       | Login         | Control Room | Anonymous  | rrcs-208-105... | 2015-12-02 15:43:22... | Failed  | Invalid credentials.                        |
| Error       | Login         | Control Room | Anonymous  | rrcs-208-105... | 2015-12-02 15:42:24... | Failed  | Invalid credentials.                        |
| Error       | Login         | Control Room | Anonymous  | rrcs-208-105... | 2015-12-02 15:41:49... | Failed  | Invalid credentials.                        |
| Information | Login         | Control Room | superadmin | 234-89-109-...  | 2015-12-02 12:30:13... | Success | Logged in successfully                      |
| Information | Login         | Control Room | superadmin | 103.244.120...  | 2015-12-02 08:06:04... | Success | Logged in successfully                      |
| Information | Client Login  | Control Room | Mike       | PRODUCTLT...    | 2015-12-02 06:38:37... | Success | User logged in from the client successfully |
| Information | Login         | Control Room | superadmin | 103.244.120...  | 2015-12-02 02:36:21... | Success | Logged in successfully                      |

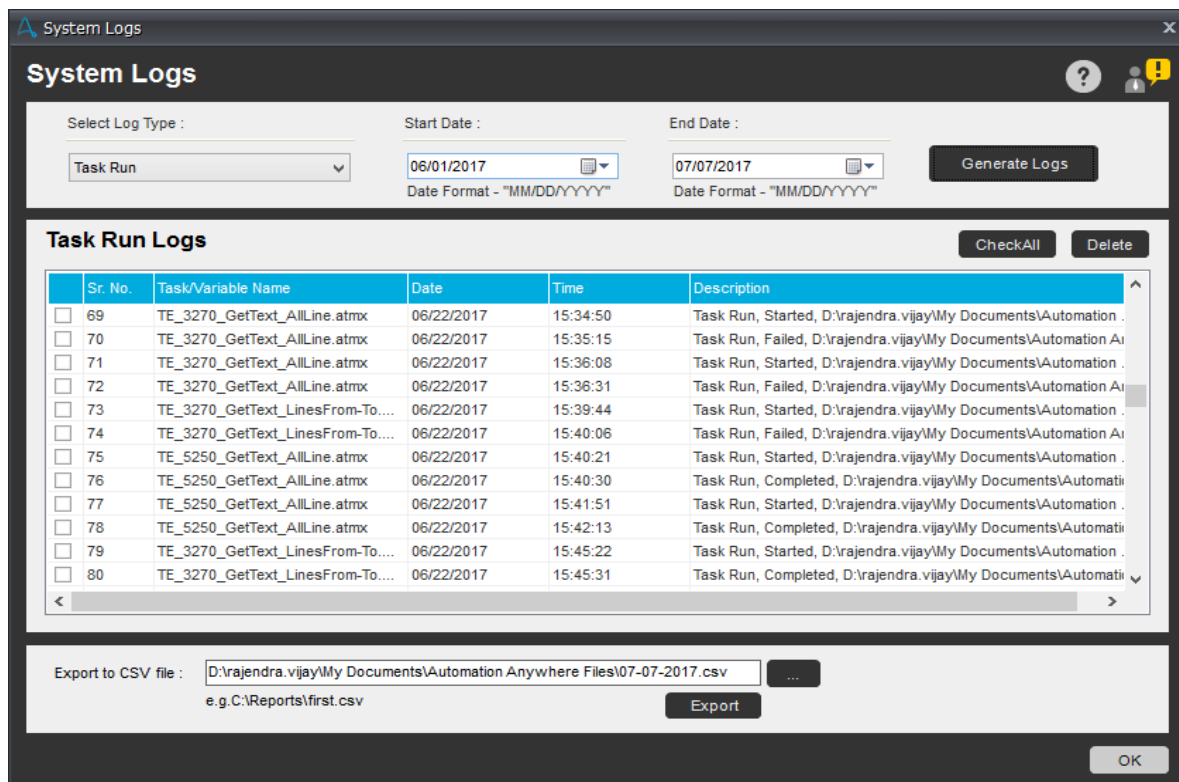
**FIGURE 27: AUDIT LOG**

## Activity Logging

For every Bot Creator and Bot Runner, Automation Anywhere performs a comprehensive activity logging for Bots, Workflows, Reports etc.

Some of the key activities logged are as below

- Task Creation, Update, Deletion (task is a type of bot)
- Task Run
- Workflow Creation, Update, Deletion
- Workflow Run
- Report Creation, Update, Deletion
- Report Run
- Change in bot properties



**FIGURE 28: ACTIVITIES LOGGED**

## Version Control

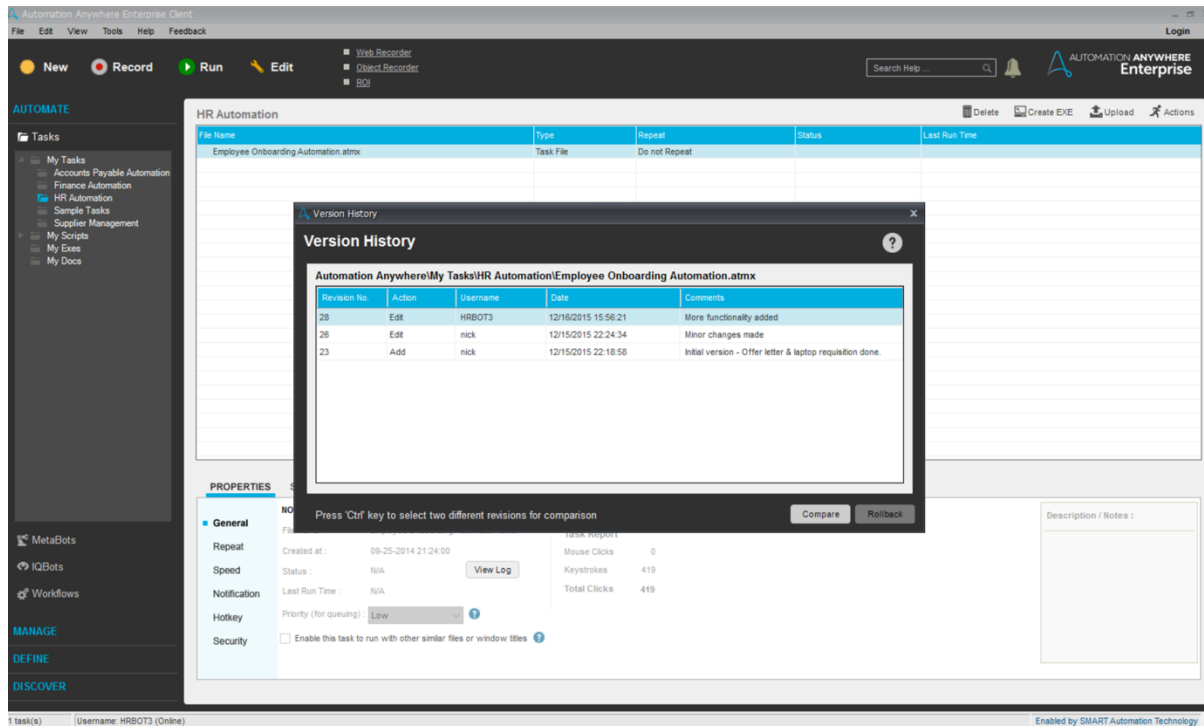
The Control Room offers a full-fledged version control system for TaskBots, MetaBots, Workflows, files etc. Version Control is provided via integration with Visual Subversion (SVN) to provide enterprise strength version management necessary to meet NIST SA-10 Developer Configuration Management requirements.

Version control is essential to change management, to ensure that the developer performs Bot configuration management especially when automating complex processes where automation must be co-authored by many automation experts. Version control provides a control point consistent with NIST SA-10 by providing a single point of control to changes to Bots to ensure that changes are consistent with integrity and quality consistent with organizationally-defined processes such as Trusted Generation procedures. Version control is enabled in the central Control Room via a simple enablement switch and specifying connection details to the SVN server. The SVN server can be provisioned on a separate machine or a clustered environment to eliminate single point of failure.

Once version control is enabled, all the rules of version management are enforced by the central Control Room:

- At a time only one user can work on updating a bot (check-out).
- When the user is done modifying a bot, s/he must 'Check In' the bot, which automatically creates a new version of the bot.
- Users can enter comments when checking a bot in. These comments will be displayed when viewing the version history.
- Users can compare any two versions of a bot to understand the changes and then take appropriate decisions.
- Users can rollback to any of the previous versions of a bot, whenever needed.

It is recommended to backup SVN server regularly using external backup mechanisms. This will ensure archival of all versions of all bots in the central Control Room. Enterprises are strongly advised to enable version control in order to maintain traceability of changes to their bots in order to systematize their automation efforts.



**FIGURE 29: VERSION CONTROL SYSTEM**

While the bots are versioned, the credentials used in bots are version agnostic. All versions of the bots use the same set of latest credentials stored in Credential Vault. This ensures that even when user switches to old versions of the bots, latest credentials will be used and bots will not fail.

## Deployment of bots

Automation Anywhere bots can be marked as “production-ready” for a selected version. Capability to mark “production-ready” version is available through Role Based Access Control (RBAC). Authorized user can select any version of a bot and designate it as production-ready. This allows automation experts to continue to collaborate on upgrading that bot through upload/download via CR. This ensures that a work-in-progress bot is not unintentionally deployed on remote Bot Runners.

## Email Alert notifications

Events captured and logged can also be configured as alerts. Email notification can be setup to have users notified when an event is detected. This feature can be setup for multiple users/ process stakeholders. E-mail alerts can help address incident response and handling (NIST IR 4, 6). The Auto Login feature in Control Room is now enhanced to send email notification in case Windows Auto Login fails on the client machine. This helps stream appropriate TaskBot failure information to the Control Room user who has deployed and run the TaskBot.

## Additional Security Controls

### Restrict CR install from database system administrator account

Automation Anywhere Control Room restricts the database connection configuration with system administrator (SA) account. All the database level transactions must be performed with a non-system administrator account. This is one of the reasons our CR installer passes the SQL Server 2012 certification test.

### Auto lock the device

When Automation Anywhere bots are deployed from Control Room to remote Bot Runners, they revert the Bot Runner system to its original state. For example, if the Bot Runner machine was logged off and our bot logged into the machine, it will log it off as soon as the automation execution is finished. Similarly, the bot will lock or disconnect the machine, per the original state. This ensures that system level security is not compromised.

### Use of SHGetKnownFolderPath function

Automation Anywhere software uses SHGetKnownFolderPath function and Knownfolder\_ID to determine the full path to the special folders. This is a recommended practice from Microsoft and use of this function ensures that system will never redirect automation data to any other folder, even if someone attempts to hack the function call. This is also one of the InfoSec requirements of our enterprise customers.

### API Level security

Automation Anywhere software performs authentication and authorization level checks at API level. API calls are serviced only for those users who have permission on the automation data. Unauthorized users cannot bypass system security through rogue API calls.

### Clean Uninstall

When automation Anywhere client software is uninstalled, it leaves no trailing files or folders behind. This clean uninstall of AAE Client software complies to enterprise InfoSec policies.

### Store data in “Program Data” folder

Automation Anywhere client software allows storing automation data into “Program Data” folder, for the files which need to be edited by end users. Permissions are also set on the directory during the installation so that user can edit the content of the folder. This complies with the InfoSec requirements of our enterprise customers.

### Protected handling of MSVC DLL files

Automation Anywhere client software uses MSVCxxx.dll files for automation purposes, but it does not install these files by itself.. Client software directly uses the DLL files installed by Microsoft operating system only. This ensures that client software does not overwrite the DLL files installed by Microsoft and our customers do not have to worry about performing one more cycle of checking for any vulnerabilities which might have introduced.

### Assembly Manifest

All the executable (.exe file) of Automation Anywhere Control Room and Client software contain the manifest files which describe assembly metadata such as file name, version number and culture. This makes our platform comply with organizational InfoSec policies.

## Application path on network

Automation Anywhere supports configuration of reading and writing automation data to a location on network drive. This enables users to keep all automation data at one place and authorized users can control the enterprise wide automation in a better way.

## Auto-login without disabling legal disclaimer

When Automation Anywhere bots are deployed from Control Room to remote Bot Runners, our customers need not change security settings such as disable login page or disable legal disclaimer or disable screensaver. Automation deployment works seamless without disabling such settings.

## Secure Java automation

Our platform can securely automate even those difficult-to-automate business applications which download java runtime environment (jre) during automation execution. Whenever such applications are started, an Automation Anywhere agent gets associated with java executable non-invasively and automates the business application. As soon as the automation is finished, our agent gets automatically terminated.

## Automation in non-English languages

Users can securely use keyboards characters of German, French, Italian and Spanish languages through the embedded automation commands in Bot Creators. This enables users to write data into these languages. Our customers need not depend on less secure third-party libraries for such automation.

## Additional Security features in Next Release (v11)

This summarizes the security enhancements scheduled to be release in v11 already covered in other sections of this document.

### Native CyberArk integration

Automation Anywhere 11.0 release will provide native integration with CyberArk, the leading privileged account security solution. Enterprises can decouple business applications' credential management from our platform. Bot Creators and Bot Runners will be able to use the credentials stored in CyberArk repository. Using CyberArk credentials, Automation Anywhere bots will authenticate against business applications.

### Support for SAML and Kerberos

Automation Anywhere 11.0 release will support Security Assertion Markup Language (SAML). Using SAML, AAE platform users will be able to perform Single Sign On (SSO) to AAE platform by authenticating against the third-party enterprise systems. When an enterprise configures SSO for our platform, user passwords will not be stored in Control Room.

Kerberos will be used as network authentication protocol to provide even stronger authentication between Bot Creator/Bot Runner and Control Room/Third party systems. Using Kerberos's strong cryptography, Bot Creators, Bot Runners and Control Room will prove their identities to one another, even over an insecure network.

### Role Based Access Control (RBAC) On Credentials

We will further segment administrative roles for separation of administrative tasks to more tightly control the Credential Vault data by applying RBAC on credentials. This will enable complete separation of one department's credentials with other department's users. E.g. Finance department users will be able to see and use only the credentials created by



Finance department admin. Finance users will not be able to see and hence won't be able to use credentials belonging to other departments.

## Tighter integration with Active Directory

Automation Anywhere platform will be tightly integrated with Active Directory. Bot Creators and Bot Runners will be able to perform domain based authentication to business applications using their credentials stored in Active Directory. This will ensure business applications' credentials need not be maintained outside of Active Directory, wherever they are present in AD. This will also enforce implementation of enterprise AD policies to Bot Creators and Bot Runners (e.g. same password policies for bots too)

## Appendix

### Control Room Directories Listing

**Installation Directory Structure:** sample output of the "dir /s/b" command

```
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Automation.CR.Core.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Automation.CR.Web.SocketServer.BusinessLogic.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Automation.CR.Web.SocketServer.exe
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Automation.CR.Web.SocketServer.exe.config
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Automation.Generic.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Automation.Generic.Security.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Automation.Legacy.NativeAPI.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Automation.Util.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Fleck.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\LightInject.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\LightInject.xml
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\log4net.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\Newtonsoft.Json.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services\RestSharp.dll
C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Control Room\Services
```

### Repository Directory Structure:



Default Path: <drive>:\Users\Public\Documents\Automation Anywhere Server Files

Sample Output of the “dir /s /b” command

```
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My Docs  
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My Exes  
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My IQBots  
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My Lists  
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My MetaBots  
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My Reports  
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My Scripts  
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My Tasks  
C:\Users\Public\Documents\Automation Anywhere Server Files\Default\Automation  
Anywhere\My Workflow
```

## Client Directories Listing

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\32-Bit

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\64-Bit

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AA.EditorX.Controller.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AA.EditorX.UI.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AA.Integrator.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AA.SchemaXML.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AA.Settings.xml

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AAAutoLoginService.exe

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AAAutoLoginService.exe.config

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AAAvatarAccessibilityBridge.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AAClientService.exe

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AAClientService.exe.config

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AA\_MyJS.js

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\AA\_MyVB.vbs

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\Automation Anywhere

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\32-Bit\Automation.CredentialProvider.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\32-Bit\sas.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\64-Bit\Automation.CredentialProvider.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\64-Bit\sas.dll

C:\Program Files (x86)\Automation Anywhere Enterprise 10.5\Client\64-Bit\sas1.dll

## Repository Directory Structure

```

D:\{user name}\My Documents\Automation Anywhere Files\AA.CloudSettings.xml
D:\{user name}\My Documents\Automation Anywhere Files\AA.Settings.xml
D:\{user name}\My Documents\Automation Anywhere Files\AA.Trigger.xml
D:\{user name}\My Documents\Automation Anywhere Files\AALog.txt
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere
D:\{user name}\My Documents\Automation Anywhere Files\ClientConfiguration.xml
D:\{user name}\My Documents\Automation Anywhere Files\Notification
D:\{user name}\My Documents\Automation Anywhere Files\ROILog.xml
D:\{user name}\My Documents\Automation Anywhere Files\ROIReportLog.xml
D:\{user name}\My Documents\Automation Anywhere Files\SystemLog.xml
D:\{user name}\My Documents\Automation Anywhere Files\TA.ObjectMapper.xml
D:\{user name}\My Documents\Automation Anywhere Files\Technology.xml
D:\{user name}\My Documents\Automation Anywhere Files\UsageStatisticsData.json
D:\{user name}\My Documents\Automation Anywhere Files\User.Settings.xml
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Scripts
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Tasks
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Scripts
\Sample Scripts
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Scripts
\Sample Scripts\CheckFileCreationDate.vbs
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Scripts
\Sample Scripts\CheckFileExists.vbs
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Scripts
\Sample Scripts\CheckFileSize.vbs
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Scripts
\Sample Scripts\CheckFolderExists.vbs
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Scripts
\Sample Scripts\CheckFolderSize.vbs
D:\{user name}\My Documents\Automation Anywhere Files\Automation Anywhere\My Scripts
\Sample Scripts\GetFileExtension.vbs

```

## List of executables

Automation Anywhere Client executables are listed below:

Automation Anywhere, AAPlayer, AAPuginInstallation, AASilverlightInjector, AATaskEditor, Automation.HelpViewer, Automatio.ChatViewer, Automation.SchedulerStartupApp, AutomationEventWatcher, AutomationScheduleMigration, AAProxyServer, JavaPath, RegisteredDLL, RemoteClientViewer, ReportManager, AANotification.

Windows Services utilized by the Clients are: AAAutologinService, AAClientService, AAESchedulerService.

## List of Cryptographic Providers

- AES256 – Microsoft .NET AesCryptoServiceProvider
- RSA2048 – Microsoft .NET RSACryptoServiceProvider
- SHA256 – Microsoft .NET HMACSHA256
- OpenPGP – Bouncy Castle

## Glossary of terms

| Acronym | Definition                |
|---------|---------------------------|
| CR      | Control Room              |
| RBAC    | Role Based Access Control |
| VCS     | Version Control System    |
| CV      | Credential Vault          |
| SVN     | Apache Subversion         |
| BC      | Bot Creators              |
| BR      | Bot Runners               |

|      |                                    |
|------|------------------------------------|
| UI   | User Interface                     |
| API  | Application Programming Interface  |
| OS   | Operating System                   |
| HA   | High Availability                  |
| DR   | Disaster Recovery                  |
| LB   | Load Balancer                      |
| MSFC | Microsoft Failover Cluster         |
| AD   | Active Directory                   |
| DC   | Domain Controller                  |
| SAN  | Storage Area Network               |
| KMS  | Key Management System              |
| API  | Application Programming Interface  |
| UX   | User Experience                    |
| JRE  | Java Runtime Environment           |
| SAML | Security Assertion Markup Language |
| SSO  | Single Sign On                     |