Security Overview

Kryon RPA

VERSION 20.9.8

Document revision: 29-Sep-2021

Contents

Introduction

Overview

Deployment & Scalability

Network

Data Privacy & Segregation

Password Policy

Authentication & Authorization Policy

Encryption

Auditing & Logs

Data Retention

Credentials Vault

Security Compliance & Penetration Testing

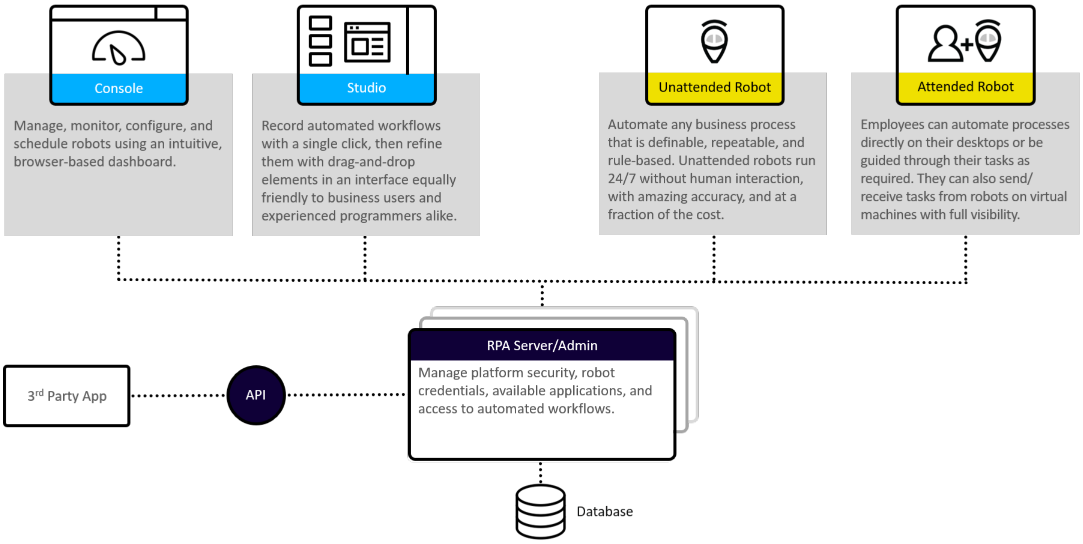
Introduction

The purpose of this document is to provide a high-level overview of the Kryon RPA Platform's system components and security features.

Unless stated otherwise, information in this document refers to common RPA use cases, including unattended, attended, and hybrid.

Overview

The Kryon RPA Platform



Manage, monitor, configure, and schedule robots using an intuitive, browser-based dashboard.

Record automated workflows with a single click, then refine them with drag-and-drop elements in an interface equally friendly to business users and experienced programmers alike.

Automate any business process that is definable, repeatable, and rule-based. Unattended robots run 24/7 without human interaction, with amazing accuracy, and at a fraction of the cost.

Employees can automate processes directly on their desktops or be guided through their tasks as required. They can also send/ receive tasks from robots on virtual machines with full visibility.

Manage platform security, robot credentials, available applications, and access to automated workflows.

Unattended robot

A lightweight desktop client installed on a virtual machine that runs a predefined sequence of actions on target applications with no human intervention.

Attended robot

A lightweight desktop client that runs a predefined sequence of actions on users' desktops. A Kryon attended robot performs actions for the user by moving his or her mouse and navigating the target application to complete the desired task. It can also educate the user on new applications, by navigating the user through the application and pointing to the exact location where he or she needs to click or enter text for each step of the task.

Studio

Studio is an Integrated Development Environment (IDE) that enables easy editing of simple and advanced automation workflows. Automation developers can add and edit desired workflows, comments, instructions, and links to other knowledge sources using an intuitive interface and robust toolbox of available commands.

RPA server

A software-based application server whose primary function is to serve as a central repository for automation workflows. It also functions as the core application for administration, provisioning, monitoring, authorizing permissions, managing licenses, and collecting end-user statistics.

ConsoleX

The central orchestration unit that manages unattended robot execution, prioritizes tasks, and schedules recurring processes. ConsoleX displays monitoring information and status of all task executions and active robots, along with historical aggregations of successful completions, exceptions, and execution time.

API integrations

As an I/O integration platform, Kryon I/O acts the hub of all cross-departmental applications regardless of their technology or formal APIs. With Kryon I/O at the center, legacy sub-systems (including ERP, CRM, billing and mainframe applications) can seamlessly collaborate.

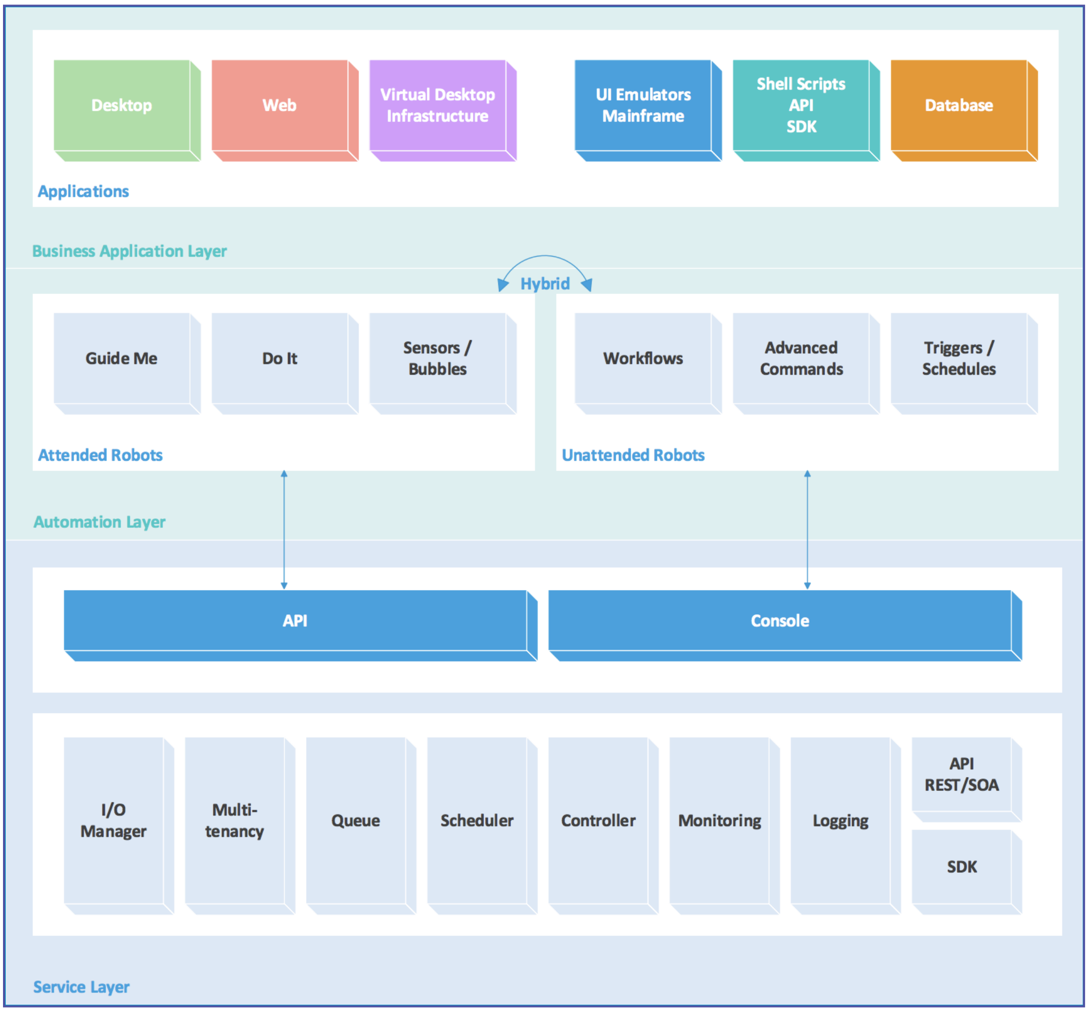
Kryon I/O uses proprietary service-bus and control-flow technology to connect any legacy or mainframe application to modern applications, including: SaaS applications, desktop applications (MSOffice), web applications (HTML/JavaScript), modern APIs such as REST, WSDL web services, and SAP applications.

Deployment & Scalability

Kryon's topology is micro-service based and configurable to support scalability, redundancy, high availability and security. It can, for example, be distributed over a dozen servers, or on a single machine – depending on need. The system supports customization of network security requirements (including TLS1.2 and HTTPS) and authentication via SSO with Active Directory and access groups. All ports can be customized to the target environment.

Kryon's platform is based on an N-tier technology. Client robots must be connected to the server. The server is a collection of services which, for large-scale deployments, are installed independently on dedicated servers; or for smaller deployments, on fewer machines.

Topology



NOTE

- For layered security services, should be deployed in a 3-tier configuration
- For redundancy services, should be deployed in N+1 redundancy

Network

Security

Using secure network protocols among all system components is recommended. The Kryon platform is regularly updated with the latest security fixes to ensure data safety and that privacy is not compromised.

Certificates are used to encrypt the transport protocol. The supported network protocols include HTTPS, TLS1.2, and secured Net.TCP. Unattended robots do not support HTTP/HTTPS.

- The certificate must be provided by customer
- Unsecured network protocols are also supported (HTTP/Net.TCP)

Firewall & port configuration

Kryon port configuration and network protocols can be configured to support all common firewall requirements. Configuration on the network may be simplified for some customers by selecting Net.TCP as the default communication protocol. To do so, IT should open ports for the local LAN or external WAN Net.TCP communication protocol.

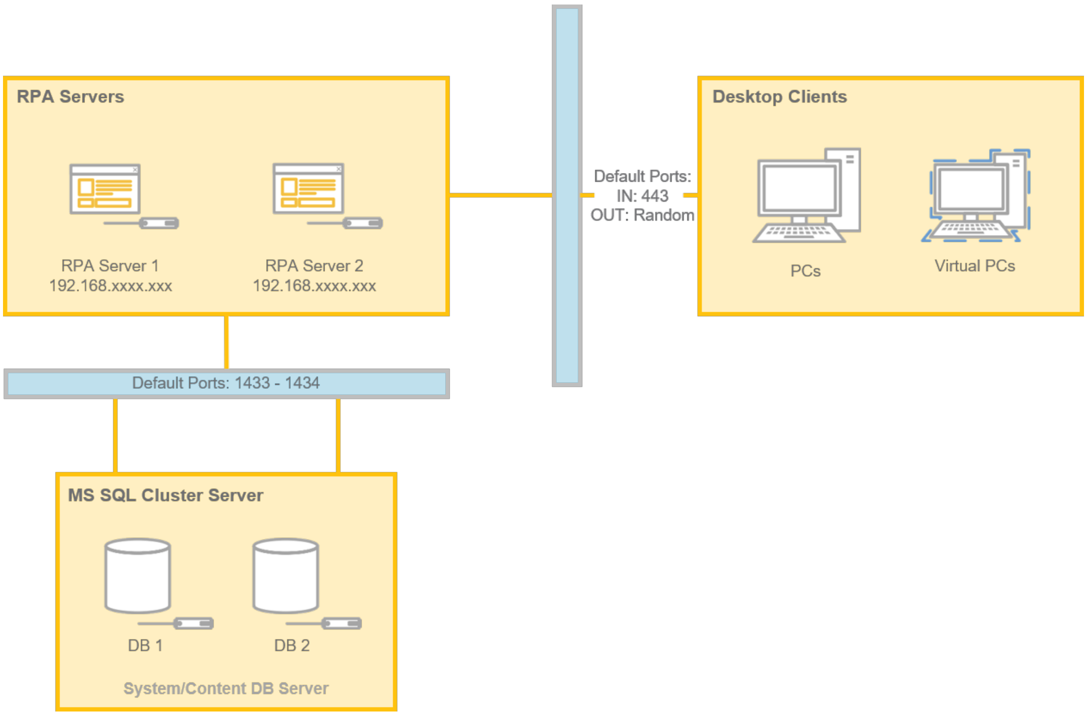
Kryon’s default port configuration is as follows (all can be customized in accordance with the organization’s security requirements):

- Between unattended robots and RPA Server: 443
- Protocol for robot instructions: Secured Net.TCP

NOTE

- Default network protocol is Net.TCP
- For production environments, the recommended protocol is HTTPS SSL/TLS 1.2 or Secured Net.TCP

  Only SSL/TLS v1.2 is supported.


RPA Servers
RPA Server 1
192.168.xxxx.xxx
RPA Server 2
192.168.xxxx.xxx
Default Ports:
IN: 443
OUT: Random
Desktop Clients
PCs
Virtual PCs
Default Ports: 1433 - 1434
MS SQL Cluster Server
DB 1
DB 2
System/Content DB Server

Data Privacy & Segregation

Administrators can manage roles and permissions for organizational security and scaling of RPA operations. Kryon's multi-tenant architecture allows administrators to configure separate working environments and teams for different business units and allocate resources such as RPA developers, managers, and robots to each team – facilitating in-team collaboration.

Multi-tenancy

Enterprises can create fully-separate and secure teams and environments. Resources including robots, RPA developers, and more can be allocated to each team. Administrators can provision multiple companies within an organization, and for each provisioned company, separate the data across all stages of the development and production automation lifecycle.

Authorization & access permissions

Only administrators have the privileges to authorize access permissions, which can be managed in several levels of granularity:

- RPA Developers can be restricted to specific companies (groups), robots, workflows
- Robots can be assigned permissions only to specific companies (groups), workflows and data
- Workflows can be restricted to specific companies (groups), embedded workflows and data

Administrators manage segregation of duties for Studio users as well as robots:

- Administrators can manage both permissions and users/robots associated with the automation workflow catalog
- Studio users can be set in groups and roles. Each group/user can have independent segregation between various departments based on organizational needs
- Administrators can share or block automation workflows per the group/user permissions definition
- Similarly, Administrators can set permissions for robots, at a group/robot level, allowing/blocking workflow execution

Groups

Administrators can group robots by task types, business units, or other criteria to manage the virtual workforce more efficiently. Instead of assigning a task to a specific robot, a task is assigned to a group and Kryon's server optimizes the task assignment, sending it to the first available robot. Administrators can add or remove robots from a group based on workload.

Password Policy

Administrators can control access to various password attributes and assign specific policies to users or groups:

- Administrators can force users to change their passwords upon first login
- Administrators cannot view users' reset password text
- Administrators can enforce the minimum password character length
- Administrators can enforce a policy requiring the use of complex passwords (uppercase letters, special characters, numbers)
- Administrators can enforce the number of invalid attempts prior to lockout
- Administrators can enforce the duration of inactivity on a workstation before the screen locks

Passwords are stored in a non-reversible format, SHA-256 or better.

Authentication & Authorization Policy

Kryon supports both Active Directory SSO authentication and secured credentials-based authentication. During the authentication process, credentials are encrypted throughout communication.

Active Directory SSO

Role-based access control can be manually configured by the IT team or automatically via the organization's Active Directory, allowing single sign-on (SSO). Kryon supports Kerberos authentication.

Credentials-based authentication

See Password Policy for details.

Encryption

Encryption is enforced by default for all sensitive data both in-transit and in-rest.

All data stored on the server side is encrypted using the Advanced Encryption Standard (AES) 256-bit key and kept in the Kryon database. All decryption is done on the client side only, so data is transmitted encrypted. No data is stored on the client side: once a robot is finished executing the relevant process and the session ends, all sensitive data is cleared.

- The encryption mechanism and key length for all encryption processes used within this product, including data in transit, data at rest (stored within the application), and any special storage (such as passwords), are as follows:
  - AES-256
  - SHA-256
- When the product communicates with itself, a client system, or another third-party system, the encryption options available to facilitate the communication in a secure manner are as follows:
  - HTTPS TLS1.2
  - Secured Net.TCP
- Passwords are stored in a non-reversible format, SHA-256 or better
- All sensitive configuration data parameters, such as passwords and connection strings, can be encrypted
- The key exchange procedure used is Diffie-Hellman Key Exchange
- See Credentials Vault for details regarding the encryption methodology used by the Kryon Credentials Vault.

FIPS compliance

The Federal Information Processing Standard (FIPS) Publication 140-2, (FIPS PUB140-2), is a U.S. government computer security standard used to approve cryptographic modules.

All encryption methods used by Kryon, both client and server-side, are FIPS compliant.

Auditing & Logs

Kryon audits all mission-critical events and changes with complete traceability into robot history and activities. The platform ensures the highest level of compliance by securing the logs so they cannot be altered or changed by individuals with malicious intent:

- Audit logging is enabled by default and not cannot be disabled by any user or role
- Event types which are logged include: logon, logoff, unsuccessful logon, access changes, task schedule transmission, critical transactions, and others
- Log file access is restricted by Windows file system permissions according to organization policy, to restrict unauthorized modification, access to, or deletion of the log file
- All logs are stored in an encrypted format

Data Retention

The Kryon RPA Platform is a client/server solution that stores no data from customer applications. All data used for RPA by the robots is encrypted on the client side and process measurements are transmitted encrypted until stored encrypted in the database.

Kryon does not persist any session data on the client machine. However, a custom purging process can be automated at session termination to clean any metadata that may reside in cookies and browser cache due to third-party website policies.

Credentials Vault

The Kryon Credentials Vault is a module within the Kryon RPA Platform used to manage login credentials for systems that Kryon robots need to access when performing automation tasks. Usernames and passwords are stored securely using data encryption.

The Kryon Credentials Vault enables automation developers to utilize the required credentials for any process without hard coding specific usernames and passwords into automation scripts and without exposing credentials to the developers.

Moreover, the Kryon Credentials Vault allows for the management of password updates in a single, centralized location – eliminating the need to update a changed password in every workflow in which it is used.

Architecture

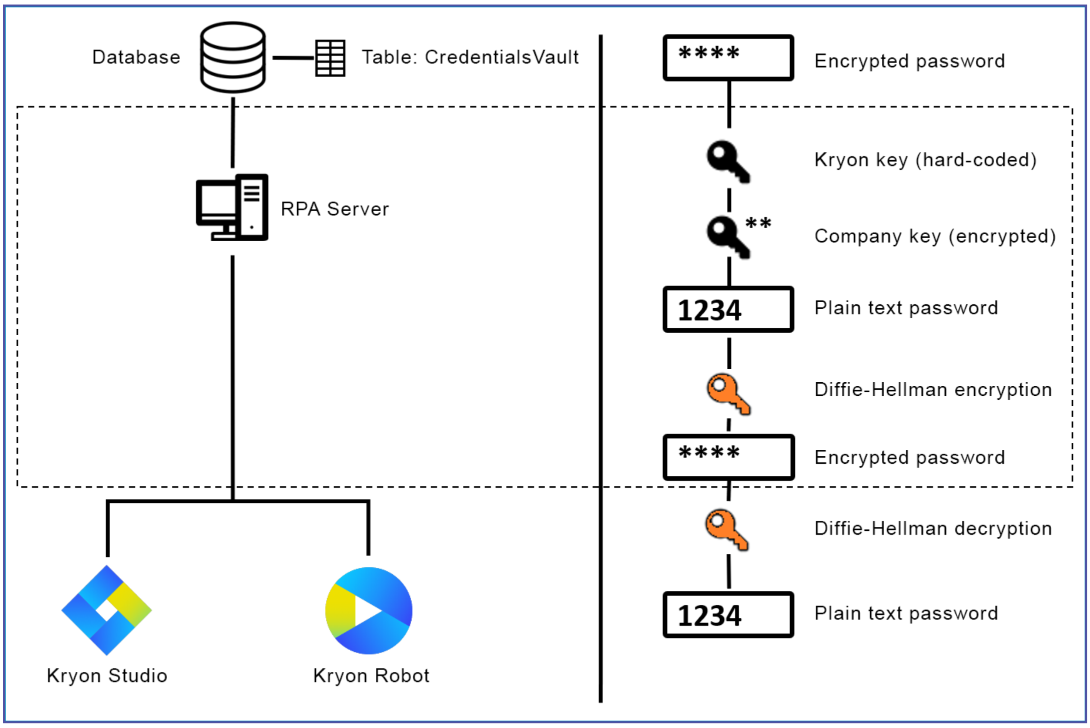
Credentials Vault data is saved in an encrypted format in the Kryon database, utilizing a 2-phase encryption process.

The AES encryption algorithm is used to encrypt/decrypt the data saved in the database, using the 256-bit key, CBC cipher mode.

A per-company encryption key is auto-generated on the server side upon each new company creation.

Upon each Credentials Vault data exchange between client and server, credentials are encrypted using Diffie-Hellman key exchange cryptographic protocol that allows two parties with no prior knowledge of each other to jointly establish a shared secret key, which is unique for each logged-in Kryon use.

A third party (potential attacker) intercepting the public keys but lacking knowledge of either private key cannot generate the shared secret key. Therefore, data encrypted with the resulting shared secret key is secure while in transit.



Diffie-Hellman key exchange usage

The shared secret key is unique and randomly generated by the Diffie-Hellman (DH) algorithm every time a particular user logs in to the system, and it is deleted when the user logs out.

The DH algorithm begins with a large prime (P) and a generator (G). These don't have to be secret and may be transmitted over an insecure channel.

The Kryon Credentials Vault utilizes the Chilkat DH component. This third-party library provides the ability to use known "safe" primes, as well as a method to generate new safe primes. (Generating new safe primes is a time-consuming, CPU-intensive task and is normally handled offline.)

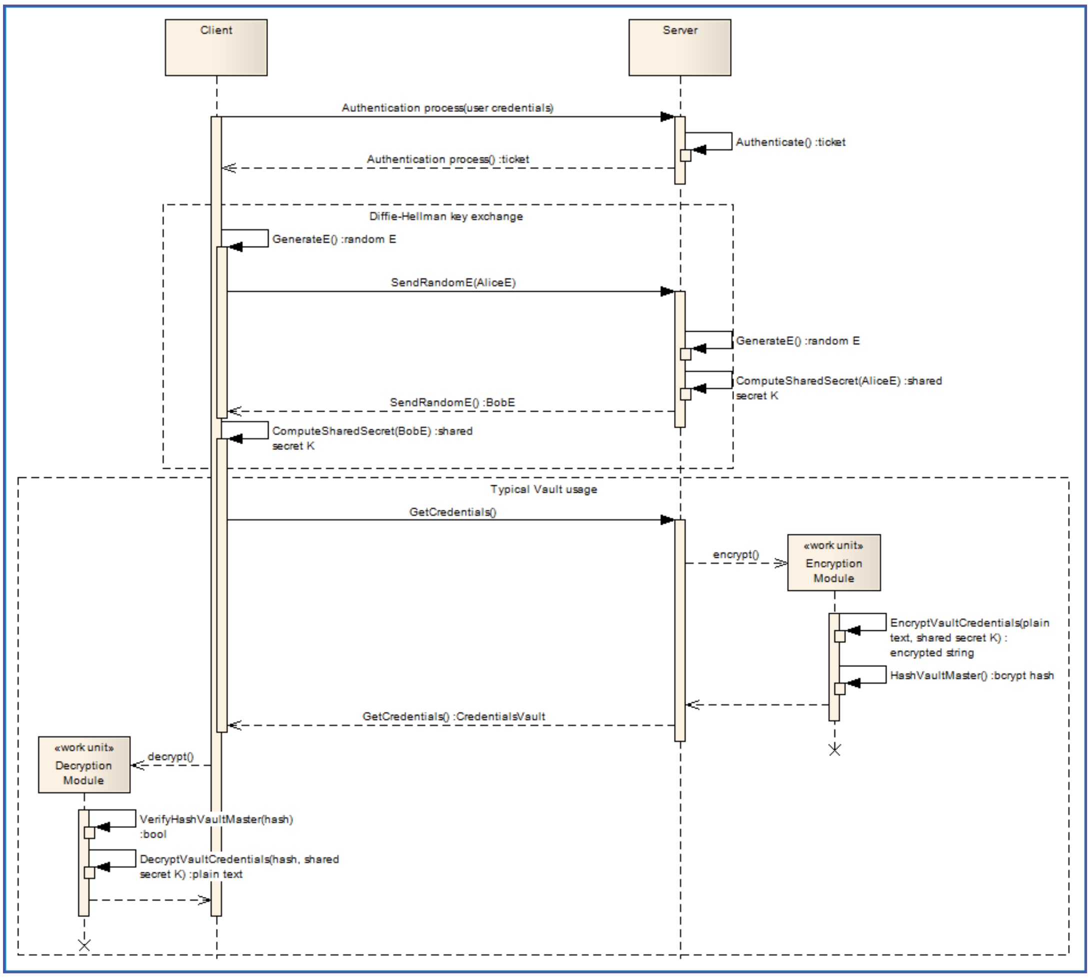
Credentials Vault uses Chilkat DH component's 8 pre-chosen safe primes.

1. Each side begins by generating an ***E*** value
   - ***Bob*** (Kryon server) generates a random ***E*** (which has the mathematical properties required for DH)
   - ***Alice*** (Kryon client) does the same
2. The ***E*** values are sent over the secured (optional) channel
   - Bob sends his ***E*** to Alice, and Alice sends her ***E*** to Bob
3. Each side computes the shared secret key ***K***

- Bob computes the shared secret from Alice's ***E***
- Alice computes the shared secret from Bob's ***E***

NOTE

The shared secret keys of both sides must be identical, otherwise a man-in-the-middle (MITM) attack could result.


Client
Server
Authentication process(user credentials)
Authenticate() :ticket
Authentication process() :ticket
Diffie-Hellman key exchange
GenerateE() :random E
SendRandomE(AliceE)
GenerateE() :random E
ComputeSharedSecret(AliceE) :shared secret K
SendRandomE() :BobE
ComputeSharedSecret(BobE) :shared secret K
Typical Vault usage
GetCredentials()
encrypt()
«work unit» Encryption Module
EncryptVaultCredentials(plain text, shared secret K) : encrypted string
HashVaultMaster() :bcrypt hash
GetCredentials() :CredentialsVault
«work unit» Decryption Module
decrypt()
VerifyHashVaultMaster(hash) :bool
DecryptVaultCredentials(hash, shared secret K) :plain text

Credentials Vault master password

Bcrypt, a password hashing function, is used to protect the Credentials Vault master password. The bcrypt cost factor (work factor, number of rounds of hashing) can be set to any value from 4 to 31. The Kryon Credentials Vault uses a value of 10.

Upon each client request to access the Credentials Vault, the server computes a hash for the master password and encrypts it using the Diffie-Hellman algorithm.

Upon receipt, the receiver (Kryon Studio/Kryon robot) computes a bcrypt hash value (the user's typed phrase) and compares the computed value to the received hash. If the two values are the same, the password authenticity is valid.

NOTE

Hash output will be different for each use because the bcrypt method automatically generates a random salt.

Security Compliance & Penetration Testing

Kryon is certified for ISO 9001:2008 compliant procedures. All employees and sub-contractors are trained according to Kryon's quality standards prior to and during their engagement at the company and are reviewed on an annual basis.

FIPS

The Federal Information Processing Standard (FIPS) Publication 140-2, (FIPS PUB140-2), is a U.S. government computer security standard used to approve cryptographic modules.

All encryption methods used by Kryon, both client and server-side, are FIPS compliant.

Security & penetration testing

Security and penetration testing is performed regularly both internally and by third-party auditors.

Kryon applications are security tested against the latest security protocols, including OWASP, WASC, blackbox testing, graybox testing, and whitebox testing, to discover and resolve security flaws or insecure coding practices such as buffer overflows, injection flaws, and improper error handling, etc.

Based on the results of and external testing and verification process provided by a third-party security auditor, there are no open critical-risk or high-risk vulnerabilities.