

Security Guide | PUBLIC

Document Version: 1.0.0 – 2024-04-04

# SAP Intelligent RPA Security Guide

# Content

<b>1</b>	<b>Security Guide. . . . .</b>	<b>4</b>
<b>2</b>	<b>Before You Start. . . . .</b>	<b>6</b>
<b>3</b>	<b>Technical System Landscape. . . . .</b>	<b>7</b>
3.1	On-premise Design. . . . .	8
3.2	Cloud Design. . . . .	8
3.3	Authentication to Cloud Factory. . . . .	9
3.4	Storage in Cloud Factory. . . . .	9
3.5	Configuration and Orchestration. . . . .	10
3.6	Agent Registration. . . . .	11
3.7	On-premise Execution. . . . .	12
3.8	API Triggers and Notifiers. . . . .	15
3.9	Monitoring. . . . .	16
<b>4</b>	<b>Security Aspects of Data, Data Flow, and Processes. . . . .</b>	<b>17</b>
4.1	Development of an Automation Project. . . . .	17
4.2	Project Upload and Configuration. . . . .	19
4.3	Job Execution. . . . .	19
4.4	Security Aspects of SAP GUI Connector. . . . .	20
<b>5</b>	<b>User Administration and Authentication. . . . .</b>	<b>21</b>
<b>6</b>	<b>Authorizations. . . . .</b>	<b>22</b>
6.1	Standard Roles. . . . .	22
	IRPAProjectMember/Delegate and Privileges. . . . .	23
	IRPAOfficer. . . . .	27
	IRPAAgentUser. . . . .	28
	IRPAParticipant. . . . .	28
6.2	Obsolete roles. . . . .	29
<b>7</b>	<b>Data Storage Security. . . . .</b>	<b>30</b>
7.1	Desktop Studio Data. . . . .	30
7.2	Cloud Studio Data. . . . .	30
7.3	Cloud Factory Data. . . . .	30
7.4	Desktop Agent Data. . . . .	31
	Desktop Agent Work Directory. . . . .	31
	Storage of Authentication Tokens. . . . .	31
	Access to the File System. . . . .	31

	Mass Registration. . . . .	32
	Local Agent Variables and Credentials. . . . .	32
<b>8</b>	<b>Data Protection and Privacy. . . . .</b>	<b>33</b>
8.1	Introduction. . . . .	33
8.2	Glossary. . . . .	33
8.3	Legal Ground for Personal Data. . . . .	34
8.4	Read Access Logging. . . . .	34
8.5	Change Log. . . . .	35
8.6	Deletion of Personal Data. . . . .	35
8.7	Automated Decision. . . . .	36
<b>9</b>	<b>Security-relevant Logging and Tracing. . . . .</b>	<b>38</b>
<b>10</b>	<b>Frequently Asked Questions. . . . .</b>	<b>39</b>
10.1	Data Security. . . . .	39
10.2	Agent Permissions. . . . .	41
10.3	Network Security. . . . .	42
10.4	Customer Data Export. . . . .	43

# 1 Security Guide

The Security Guide provides an overview of the security-relevant information that applies to SAP Intelligent Robotic Process Automation.

## Caution

This guide does not replace the administration or operation guides that are available for productive operations.

## Why this document?

With the increasing use of distributed systems and the Internet for managing business data, the demands on security are also on the rise. When using a distributed system, you need to be sure that your data and processes support your business needs without allowing unauthorized access to critical information. User errors, negligence, or attempted manipulation of your system should not result in loss of information or processing time. These demands on security apply likewise to SAP Intelligent Robotic Process Automation. To assist you in securing SAP Intelligent Robotic Process Automation, we provide this Security Guide.

## Target audience

- Technology consultants
- Security consultants
- System administrators

This document is not included as part of the Installation Guides, Configuration Guides, Technical Operation Manuals, or Upgrade Guides. Such guides are only relevant for a certain phase of the software life cycle, whereas the Security Guides provide information that is relevant for all life cycle phases.

## Overview of the Main Sections

The Security Guide comprises the following main sections:

- **Before You Start:** This section contains information about why security is necessary, how to use this document, and references to other Security Guides that build the foundation for this Security Guide.
- **Technical System Landscape:** This section provides an overview of the technical components and communication paths that are used by SAP Intelligent Robotic Process Automation.
- **Security Aspects of Data, Data Flow and Processes:** This section provides an overview of security aspects involved throughout the most widely-used processes within SAP Intelligent Robotic Process Automation.

- User Administration and Authentication: This section provides an overview of the following user administration and authentication aspects.
- Authorizations: This section provides an overview of the authorization concept that applies to SAP Intelligent Robotic Process Automation.
- Data Storage Security: This section provides an overview of any critical data that is used by SAP Intelligent Robotic Process Automation and the security mechanisms that apply.
- Data Protection: This section provides information about how SAP Intelligent Robotic Process Automation protects personal or sensitive data.
- Security-Relevant Logging and Tracing: This section provides an overview of the trace and log files that contain security-relevant information, for example, so you can reproduce activities if a security breach does occur.

## 2 Before You Start





### Fundamental Security Guides

SAP Intelligent Robotic Process Automation was designed to be integrated to the [SAP Business Technology Platform](#) and, to be more precise, to Cloud Foundry. That is why the corresponding Security Guides also apply to SAP Intelligent Robotic Process Automation.

For anything related to authentication and authorization in Cloud Foundry environment, check out the [SAP Business Technology Platform Security Guide](#).

### Additional Information

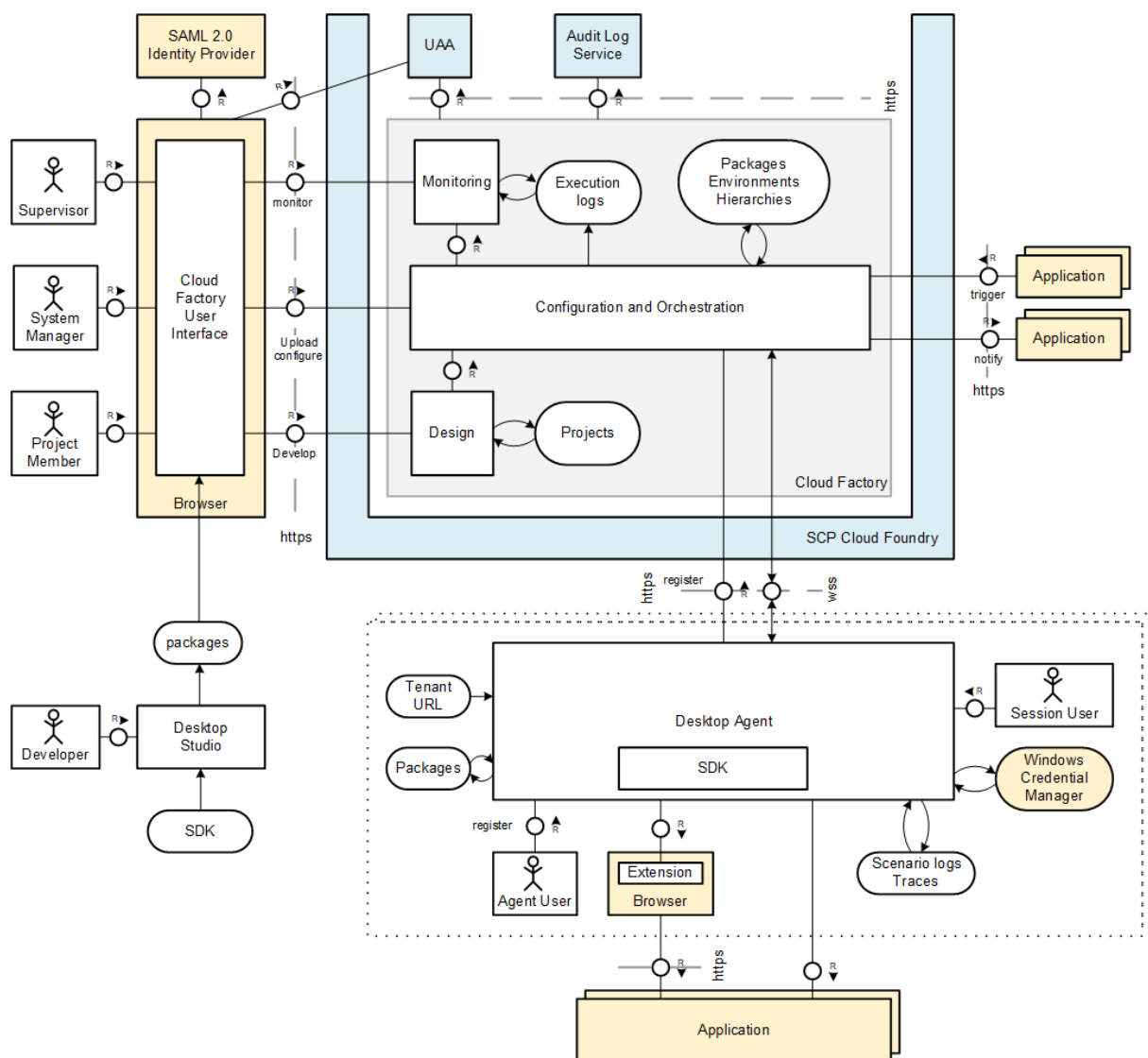
For more information about specific topics, see the Quick Links as shown in the table below.

Content	Address
Security	<a href="https://www.sap.com/community/topics/security.html">https://www.sap.com/community/topics/security.html</a> 
Related SAP Notes	<a href="http://support.sap.com/notes">http://support.sap.com/notes</a> 
Released platforms	<a href="http://support.sap.com/pam">http://support.sap.com/pam</a> 
SAP Solution Manager	<a href="http://support.sap.com/solutionmanager">http://support.sap.com/solutionmanager</a> 

### 3 Technical System Landscape

SAP Intelligent Robotic Process Automation has a hybrid architecture with a significant on-premise component dedicated to interactions with on-premise business applications and a central component deployed in the cloud.

On-premise software components include the SAP Intelligent Robotic Process Automation Desktop Studio for designing automation scenarios as well as the SAP Intelligent Robotic Process Automation Desktop Agent for executing them. The cloud part is concerned with the configuration, orchestration and monitoring of deployed scenarios. It also includes the definition of projects and the design of processes.



The on-premise components SAP Intelligent Robotic Process Automation Desktop Studio and SAP Intelligent Robotic Process Automation Desktop Agent are located at the bottom of the diagram. The Cloud part

is depicted at the top of the diagram, outlining its design, configuration, orchestration, and monitoring capabilities.

The high-level workflow is as follows:

1. Build automation projects in the Desktop Studio or processes in the Cloud Studio.
2. Export the project from the Desktop Studio, deploy it, and configure it in the SAP Intelligent Robotic Process Automation Factory.
3. Execute the project whereupon it will be pushed to the Desktop Agent for actual execution.

## 3.1 On-premise Design

Once installed, any developer can use the Desktop Studio to develop desktop scenarios. At the end of the development process, the developer generates a package containing the JavaScript code corresponding to the project.

This package contains integrity markers that will be checked by Desktop Agent before loading the package for execution, to ensure that the content of the package has not been altered.

Once a project is ready for execution, it must be uploaded to the Cloud Factory. The Cloud Factory is a multi-tenant SAP Business Technology Platform Cloud Foundry application. Customers subscribe to this application and use it in the limits of allocated resource quotas. Computing resources are shared between the different customers using this application, but we make sure that no data can leak from one customer to another. Projects developed using the Desktop Studio are called “Desktop Projects”.

Once uploaded to the Cloud Factory, the package can be directly configured for execution, or used in a process via the Cloud Studio.

## 3.2 Cloud Design

Users design processes and data types using the Cloud Studio, which is a part of the Cloud Factory.

In the Cloud Studio, users can create projects, and develop data types or processes within these projects. They can also import desktop packages so that desktop scenarios can be triggered within process executions.

At the end of the creation of a project, a package is generated containing the information required for the execution of the processes and / or scenarios. Packages are immutable and identified by a unique version number.

Interaction with the SAP Intelligent RPA web application is done via a web browser. All communications between the user browser and the backend use HTTPS (TLS 1.2) on the standard 443 port.

Projects developed using the Cloud Studio are called “Cloud Projects”.



## 3.3 Authentication to Cloud Factory

The authentication to the SAP Intelligent RPA Cloud Factory is done using the SAML 2.0 protocol. Customers may register any SAML 2.0 compliant identity provider to handle authentication. If the registered SAML 2.0 identity provider is within the customer's network, the user credentials will always stay within that network. For more information, see [Authorization and Trust Management](#).

Internally, Cloud Factory uses OAuth to ensure that any request to a service is properly authenticated. Identity information is provided to the services via a JSON Web Token containing information about the authorizations of the identified user. Each service verifies the validity of the token and checks that the user has the proper authorizations to execute the request.

Permissions are associated to users in the SAP BTP (Business Technology Platform) cockpit, by associating applicative roles to individual users or group of users. For more information, see the [SAP BTP documentation](#).

### Note

The SAP Intelligent RPA Cloud Factory never accesses directly the Identity Provider and cannot even list the users.

## 3.4 Storage in Cloud Factory

### Multi-tenancy

The SAP Intelligent RPA Cloud Factory is a multi-tenant application, which means that all customers share the same applicative components, the same databases and the same network.

Two kinds of databases are used:

- Relational databases
- Object stores

To make sure that customers are properly isolated from one another, strict rules are applied:

- In a relational database, customer data is isolated in distinct schemas
- In an object store, customer data is isolated in distinct paths or collections

The choice of a path or a schema is enforced by our development framework to avoid any risk of error.

## Storage Provisioning

The databases are provided by SAP BTP Cloud Foundry services, and the credentials required to access these databases are provided automatically by the infrastructure, without being seen or manipulated by administrators.

Databases are not Internet-facing, and only cloud operations administrators can access these databases. The list of authorized administrators is regularly reviewed.

## Data Encryption

The physical storage used by the databases is encrypted. However, some additional encryption is performed at the application level for data that might be sensitive: application captures and screenshots, job inputs and outputs, Desktop Studio variables of type [Credential](#) and Cloud Studio environment variables.

These data are encrypted using AES256 symmetric encryption. The keys used for encryption are specific to the customer. These data are decrypted on demand. If they are required for the execution of an automation, their values are decrypted in the cloud. Then they are transferred to the agent via the secure web socket (and thus re-encrypted during the transfer), and accessed on the agent.

The customer specific encryption keys are currently stored using the [SAP Credential Store](#) provided by the SAP BTP.

## 3.5 Configuration and Orchestration

A package must be associated with an environment to execute its processes or scenarios. The environment links the package to [environment variables](#) and to desktop agents (for the execution of scenarios).

Environment variables may be of several types:

- They may be simple textual information (like a system URL) representing different settings to be used by the processes or scenarios
- They may be credentials that will be needed during the execution to connect to specific systems

The values of these variables are stored in the environment. Simple cloud variables are stored directly, but credential variables are stored using AES 256 encryption.

Environments are associated to groups of Desktop Agents. Agent groups define matching criteria for agents (currently based on user login or machine name). Only these agents may be used to execute jobs within that environment.

Scenarios may be executed in two different modes:

- **Attended:** the Windows User will start the execution manually from her or his PC. To be able to so, the required packages must first be deployed to the agent. Package deployment is configured within the environment by defining time slots when a given package should be available on the agents of the environment.

- Unattended: the execution is triggered from the cloud, using a schedule or an API that can be invoked from external applications or systems. See Section on authentication for details on the required authorization mechanism in this case. Unattended execution means that projects may be sent to a PC and executed without notifying the Windows User or letting this user control what is being done, which may cause security issues if not intended. That's why this mode must be **explicitly enabled** on the Desktop Agent and can be disabled at any time.

## 3.6 Agent Registration

### Registration Process

A Desktop Agent needs to be registered in SAP Intelligent RPA Factory before any package can be delivered and executed on it:

1. On agent startup, a user with the role `IRPAAgentUser` authenticates to the Cloud Factory.
2. After authentication, a JSON Web Token (JWT) is generated that uniquely identifies the agent (namely, the machine used AND the Windows session user), and sent back to the agent.
3. The JWT is stored securely in the Windows Credential Manager on the agent machine. The algorithm used to locally encrypt the JWT depends on the Windows system version (AES256 on Windows 10).
4. If the agent is disconnected, it can reuse that token to re-establish the connection to the Cloud Factory.

The JWT is valid forever. However, it can be revoked at any time from the Cloud Factory: if so, the agent has to be registered again to be usable.

The SAP Intelligent RPA user performing the registration and the Windows session user can be different persons. However, the registration process must always be performed within the Windows session of the agent user and hence requires that the session user has allowed the SAP Intelligent RPA user to perform this operation. On the Cloud Factory side, the agent is identified using the Windows session user login and a machine hardware unique identifier (currently computed using a unique BIOS identifier and the **Boot volume Device ID**). It is not possible to register a given agent from a different computer or session.

### Mass Registration

The registration process as described earlier requires that each agent is registered individually, and each time a user needs to provide credentials. This is adapted to deployments with only a few agents, or for development. Mass registration provides a way to deal with large deployments. The process is different and as follows:

1. An Intelligent RPA Factory user creates a "registration" via the user interface.
2. The registration is copied in the customer's network, so it can be accessed from each agent location.
3. Agents can be configured to check that location, and use the registration to automatically register to the Factory.
  - The user who created the registration, and users designated as "co-managers" in the registration have MANAGE privilege on these agents;

- These agents are automatically shared with the agent group that was provided in the registration;
4. The users with MANAGE privilege on the agent group has the possibility to accept the registered agents, or reject them.

For more information on mass registration process, see [Mass Agent Registration](#).

## Tenant URL File

To be able to authenticate to the Cloud Factory, the Agent User needs a URL. This URL is either provided directly by the user (first registration) or read from a json file located in `C:\ProgramData\SAP\Intelligent RPA\Tenants`.

### Note

From agent version 3.16, the new file location is  
`C:\Users\USER_NAME\AppData\Roaming\SAPDesktopAgent\Store\Tenants`.

## 3.7 On-premise Execution

### Communication Between Desktop Agent and Cloud Factory

Once an agent is registered, it can communicate with SAP Intelligent RPA Cloud Factory. Communication between the Desktop Agent and the Cloud Factory is required when the jobs start and at the end of their execution. In between, it will only be required if variable values are needed or if Business Activity Monitoring events are issued.

#### Communication Protocol

Communications between the agent and the Cloud Factory use the Secure WebSocket protocol, which means that all of them will be encrypted by the TLS 1.2 layer. WebSockets provide a persistent connection between Desktop Agent and Cloud Factory that both parties use to start sending data at any time.

Although the execution of the jobs is logically triggered by the orchestrator on the cloud side, the communication between an agent and the cloud application is always initiated by the agent. This way, there is no need to open an inbound port from the Internet.

### Note

When running behind a proxy or a firewall, make sure that the WebSockets secure protocol (wss) is properly supported and not blocked.

The JWT stored in the Windows Credential Manager is used to authorize the agent when it needs to communicate with Cloud Factory. Agent authentication is done by retrieving and communicating the current

windows user login and a unique machine identifier (based on the operating system UUID and the boot device ID). Authorization is only provided if user login and machine identifier match what is stored in the JWT.

## Exchanged Data

Once a secure communication channel with the Cloud factory is set, the agent can exchange commands or events with the Cloud Factory.

- Commands indicate actions to be performed by the agent or by the Cloud Factory. For instance, the Cloud factory can ask the agent to execute a given scenario, or the agent can ask the Cloud Factory to provide the value of an environment variable.
- Events are emitted by the agent to indicate some state change. For instance, the agent will indicate to the Cloud Factory that some execution has been started or is finished.

Data can be exchanged as a result of these commands and events:

- Before executing a job, the package needs to be downloaded from the Cloud Factory
- When a job is triggered from the Cloud Factory, input data will be provided by the Cloud Factory to the Desktop Agent (except when the execution is triggered locally, in attended mode).
- During the execution of a job, the job may need to access the values of environment variables. This is controlled by the project. Credential variables are decrypted in the cloud, and sent over the wss connection, that re-encrypts them until they reach the agent.
- During the execution of a job, the job may send Business Activity Monitoring events to Cloud Factory. This is controlled by the project
- At the end of the execution of a job, the job output will be sent back to Cloud Factory.
- For cloud projects using the Core SDK version 1.11 or above, traces and logs are first saved encrypted on the agent's workstation, then sent to the cloud where they can be visualized by authorized users. For other projects, logs and traces stay unencrypted on the agent's workstation

No other business data are exchanged between the Desktop Agents and the Cloud Factory.

## Packages

Packages are received by the Desktop Agent and stored locally. When a package is built, some integrity markers are computed and inserted into it. Before loading a package, the agent will check these markers, and refuse loading modified packages, ensuring that the code of the package has not been altered since its creation.

In V2, packages are also signed using a private key that is generated specifically for the customer, and whose public counterpart is sent to the agent. This way, the agent can verify the authenticity of the checksum.

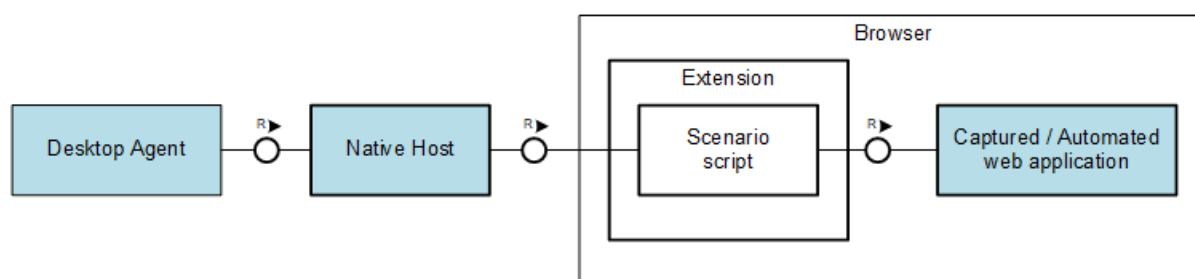
## Scenario execution

A scenario is a JavaScript program used to automate tasks on some applications.

The way to automate applications depends on the technology of the application and is implemented inside the agent via technology-specific connectors. These connectors take advantage of the technology to provide automation mechanisms that are as stable and reliable as possible.

Automating Web applications cannot be done reliably without accessing the structure of the application pages. That's why using a browser extension for that is mandatory. These browser extensions are used at capture-time to inspect the automated application pages and unambiguously describe the elements required for the automation. Browser extensions are used at execution time to act on the captured elements.

The communication between the extensions and the agent is mediated on the Desktop agent machine by a process called "Native Host". This process is started by the browser when the extension is loaded, and only the communication (via stdin / stdout) between the extension and that process is authorized. The Desktop Agent communicates with the native host using the Windows `wm_copydata` function. All these communications can only happen within the same machine.



The extensions provided as part of SAP Intelligent RPA are distributed through the official distribution channels of each supported browser. They are signed to make sure their content cannot be altered.

## Impersonation

Desktop Agents, even in unattended mode, run inside a Windows session, and inherit the privileges of the session's user. They can do whatever that user can do.

In order to authenticate to automated applications, Desktop Agents may leverage credential variables. Credential variables are securely stored in the Cloud Factory, and securely transmitted to the agents. Credential variables are very useful for development, or in the unattended case, when the session user is just a technical user that has no permissions on automated applications.

Credential variables are not handy in the attended case, where each Desktop Agent is run by a distinct person, who should authenticate directly to the automated applications. In this use case, it is usually more convenient to deploy certificates on the agent machines: since the Desktop Agent runs in the user session, it can access the Windows Credential Manager and use credentials or certificates that are stored there.

This approach can also be used in the unattended case. It should be preferred since it doesn't require credentials to be stored outside of the customer's network and facilitates credential management.

## Execution Data

### Job Outputs

At the end of the execution of a job, the result of the job execution (as defined by the scenario code) is sent back to the cloud, where it is stored.

Since job outputs may contain personal or sensitive data, they are stored in the database using AES 256 encryption. The keys used for encryption are specific to the customer. These data are kept at most 30 days within the Cloud Factory.

### **Logs, Traces, and Business Activity Monitoring Events**

For cloud projects using the Core SDK version 1.11 or above:

- Logs generated by the packages (using the “Log Message” activity) are sent to the Cloud Factory. Before being sent, they are cached locally, but they are encrypted beforehand, and will only be decrypted when authorized users need to visualize them from the cloud studio, or download them. Their content depends on the executed project and is completely controlled by the project developers.
- Traces are also sent to the Cloud Factory, and like logs, they are stored encrypted before being sent, and can only be decrypted by authorized users. They contain very detailed information about the execution of the jobs, including potentially sensitive or personal information.
- Business Activity Monitoring events are also controlled by the project developers, and sent to the Cloud Factory, so that they can be collected and analyzed. Like the other logs, they are encrypted before being cached locally.

For other projects :

- Logs generated by the packages (e.g., using the `ctx.log()` SDK function in Desktop projects) are stored unencrypted on the Desktop Agent machine. They are not sent to the Cloud Factory.
- Traces are also stored unencrypted on the Desktop Agent machine. Since they may contain sensitive information, and are not encrypted, they should only be used for debugging purposes in test environments.
- Business Activity Monitoring events are sent to the Cloud Factory, but they are cached locally unencrypted.

## **3.8 API Triggers and Notifiers**

Scenario execution can be triggered externally by an API, and execution results may be sent to other applications via notifiers

When using an API trigger, two different authentication mechanisms are being used:

- The first mechanism allows a client application to connect to SAP Intelligent RPA and to get an authorization token. To do so, an application needs a client ID and secret. This client ID/secret pair is common to all APIs. It enables the connection to an authorization server which is dedicated to the API triggers, and is never used for any other purpose. Even though these credentials are not enough to give access to any sensitive part of SAP Intelligent RPA, they should be handled securely.
- The second authentication mechanism is the `irpa-trigger-token` header which is dynamically generated when the API trigger is created. This token is specific to one API trigger. Together with the client ID and secret (password), it allows an application to use that API trigger to start jobs. This token should also be handled securely, and never disclosed outside the context of the client application runtime.

## 3.9 Monitoring

Monitoring collects events coming from the Desktop Agents and presents them to the user. These events represent the state of the agents, the status of the jobs that have been run, and the Business Activity Monitoring events.

Job events contain the outputs of the jobs. Since job outputs may contain business data, they are stored encrypted in the database, using an AES 256 encryption key. The Job outputs may only be viewed by users with the `IRPAPersonalDataAccess` role.



## 4 Security Aspects of Data, Data Flow, and Processes

This section describes security aspects relating to the use and storage of sensitive or personal data.

### Related Information

[Development of an Automation Project \[page 17\]](#)

[Project Upload and Configuration \[page 19\]](#)

[Job Execution \[page 19\]](#)

[Security Aspects of SAP GUI Connector \[page 20\]](#)


### 4.1 Development of an Automation Project

This step is achieved using the Desktop Studio and Cloud Studio under the developer's identity.

Developers may include any JavaScript in their projects, and the Cloud Factory cannot detect whether a part of that code could cause confidentiality, integrity or availability issues on the agents on which a project would be deployed. It is thus very important to review the project's code carefully before uploading it. After the upload, SAP Intelligent Robotic Process Automation Desktop Agent makes sure that the code of the project being executed has not been modified locally.

When executing Desktop Projects, or Cloud Projects using the core SDK 1.10 or older, explicit calls to the logging function `ctx.log` in the Desktop Studio and the [Log](#) activity in the Cloud Studio should be carefully reviewed to make sure that no sensitive or personal information can be disclosed in the logs. For other projects, the risk is mitigated by the local encryption of logs, and the access control mechanisms to view these logs, although personal information still needs to be properly managed if it is present in the logs.

#### ⚠ Caution

Criteria to recognize applications, screens or fields (in applications or PDF files) may be based on **regular expressions**. Regular expressions may be misused and lead to an exponential usage of computing resources (for more information, see the [OWASP site](#) ).

SAP Intelligent RPA will warn you about some dangerous regular expression patterns, but we cannot guarantee that the detection is complete. Always double-check the regular expressions that are used within projects.

### ⚠ Caution

The review process of automation projects should pay particular attention to four topics: presence of sensitive or personal data, logging, usage of local agent variables or credentials and code correctness.

Local agent variables are stored in the Windows registry and completely handled by the automation code. It is important to note that misusing these variables can lead to an instability of the operating system. Also, note that these variables and their values will not be removed if the agent is uninstalled. Therefore, it is important not to store personal or sensitive information.

Similarly, local agent credentials must be used with care. By removing the default prefix, existing credentials can be erased, which can cause issues on the computer.

Project files should not contain any kind of sensitive or personal information, like credentials to connect to an external application. SAP cannot provide Data Protection Regulations compliance mechanisms for sensitive or personal data contained within a project. Other mechanisms are provided to securely use credentials to connect to applications ("credential variables").

Explicit calls to the logging function `ctx.log(...)` in the Desktop Studio and the log activity in the Cloud Studio should be carefully reviewed to make sure that no sensitive or personal information can be disclosed in the logs.

Project files essentially consist of JavaScript code but there are no enforced restrictions on language:

- Follow secure coding best practices while coding your project such as input validation.
- Pay attention to information written to files, or more generally output using some communication channel. This information may be visible to the user that runs the agent or disclosed to a third party.
- Javascript may be used to dynamically evaluate user input. Allowing the usage of functions such as `eval()` is a risk that needs to be evaluated for fix or acceptance.
- It is possible to build a project in such a way that user input is used in launching OS commands. This needs to be properly reviewed.

### ⚠ Caution

The Cloud Studio 2.0 release provides the ability to create automations, and to capture applications. Until now, this was only possible using the Desktop Studio. Application captures include screenshots and the detailed structure of the captured screens (which is required to declare the page elements), and these captures might contain confidential information. Although these captures are treated as confidential information, our recommendation is not to use production systems for the capture, but development or test systems, so that sensitive data don't end up in the cloud.

### ℹ Note

Screenshots and captures are encrypted in the cloud using a customer specific AES-256 key. When projects or packages are exported, screenshots and captures are decrypted before being stored in the zip file that will be downloaded. When the project or package is imported, screenshots are encrypted in the target customer space.

## 4.2 Project Upload and Configuration

Before uploading the project, make sure that its contents have been properly sanitized both in terms of content and behavior.

Credentials should only be provided using credential variables, or in [password fields of cloud variables](#) in the deployment configuration. It is the only way to ensure their proper encryption in the cloud.

### ⚠ Caution

Variables containing sensitive information are protected in the cloud and in transit to the agent. However, they are **not** protected on the client machine. Always consider that the information contained in these variables may be disclosed to the user of the session running the agent. If you don't want sensitive information to be visible from all users, please use unattended automations on workstations that will be operated and accessed by a limited number of authorized users.

## 4.3 Job Execution

### Session Management

Agents run in a Windows session and inherit the privileges of the session's user. Make sure that this user only has the privileges required to access the automated applications and the agent. Do not run an agent in an administrator's session. To use the "auto-start" feature, the session user needs to enter his or her credentials to let the Agent service restart the session. These credentials are stored in the "Session 0" credential store, and never sent to the cloud. The same credentials are used to unlock the screen when the running automation mandates that. The screen unlocking time should be limited to the minimum: the default values are 3s for the unlock timeout, and 10s for the re-lock timeout. In some cases, these values may be too small: they may be changed, but should be kept as small as possible.

Note that in some cases, the screen won't be able to relock, for instance if an open application prevents the screen from relocking, or if the node runtime crashes.

### ActiveX Execution

Intelligent Robotic Process Automation Agents are implemented using COM components. It's recommended to disable ActiveX execution from Internet Explorer to prevent unwanted usage of these components from third party web sites using ActiveX technology.

## Sensitive Information

The project files are sent from the cloud application to the agent. These project files should not contain any sensitive or personal information.

## Inputs and Outputs

Input parameters are provided to the agent to let it execute a scenario. Input parameters are provided through the triggering mechanism, or via an [addJob](#) instruction created by another scenario. Input parameters are defined by the scenario developer depending on the use-case. If transferring information between the agent and the cloud is a legal or a security problem, you can replace this information by a reference to a file or any other location within the customer's network, and let the agent read information from that location. In this case, the information provided as input is limited to the location, and no sensitive information is transferred.

Outputs are sent by the agent at the end of its execution. If the job fails, these outputs may be used for debugging. They will only be visible to users having the `IRPAPersonalDataAccess` role (see [Authorizations \[page 22\]](#)). Outputs are also required when they are used as input to other jobs. Like for inputs, if transferring information between the agent and the cloud is a legal or security problem, you can replace this information by a reference to a file or any other location within the customer's network, and let the agent write information to that location. In this case, the information provided as output is limited to the location, and no sensitive information is transferred.

## 4.4 Security Aspects of SAP GUI Connector

You can capture SAP applications with Desktop Agent using the SAP GUI connector.

SAP GUI for Windows is a front-end application that you can use to access SAP applications such as SAP ERP, SAP Business Suite, and so on. It is designed for the Windows operating system and provides a Windows-like user experience and integration with other applications based on OLE interfaces or ActiveX controls.

For automation purposes, you need to enable and use the SAP GUI Scripting API. You need to enable both [client](#) and [server](#) scripting. A JavaScript library (`SAPScripting.js`) is available to implement specific behaviors.

SAP GUI Scripting is by default deactivated. You can activate it for selected users who need to use the SAP GUI connector for automation.

If you have questions about the security of SAP GUI Scripting, please see the section [Security Q&A](#) in the *SAP GUI Scripting Security Guide*.

## Related Information

[SAP GUI Scripting Security Guide](#)

## 5 User Administration and Authentication

SAP Intelligent Robotic Process Automation doesn't implement any specific user management and authentication mechanism. It completely relies on the user management and authentication mechanisms provided with the [SAP Business Technology Platform](#). Therefore, the security recommendations and guidelines for user administration and authentication as described in the [SAP Business Technology Platform Security Guide](#) also apply to SAP Intelligent Robotic Process Automation.

### ⚠ Caution

In case a web API trigger token is compromised, it should be immediately revoked from the API trigger user interface.

To limit the impact of a token being compromised, distinct client applications should use distinct API triggers.

### ⚠ Caution

To be able to authenticate to the Cloud Factory, the agent user needs a URL. This URL is either provided directly by the user or read from a json file located in `C:\ProgramData\SAP\Intelligent RPA\Tenants`. Since this json file may be modified by all users of the agent machine, it is always important to check that it is correct: Users should always carefully check that the URL indicated in the authentication window is the URL of their tenant.

From agent version 3.16, the new file location is

`C:\Users\USER_NAME\AppData\Roaming\SAPDesktopAgent\Store\Tenants`. Also, file modification no longer affects other users.

### ⚠ Caution

API keys can be defined at the environment level to let external applications or systems trigger automations or processes. Even if these keys can only be used to trigger specific skills, and in no way to connect to the factory, they must be handled securely to prevent any malicious user to deliberately misuse the triggers. This recommendation also holds for the client id and secret that are needed to retrieve an authentication token, and for the token itself.

To help keep your internet-facing triggers secure, follow these best practices:

- Do not embed API keys, client id, and secret, or authentication token directly in code. Instead of embedding these credentials in your applications, use a password vault to retrieve them, or at least, store them in environment variables or in files outside of your application's source tree.
- Delete unneeded API keys to minimize exposure to attacks.
- Rotate your API keys periodically. To do that, simply create a new API key in your environment, and reconfigure the applications and systems so that they can use the new key. Once all the calling applications have been updated, you can safely delete the previous key.

# 6 Authorizations

SAP Intelligent Robotic Process Automation uses the authorization concept provided by the [SAP Business Technology Platform](#) and Cloud Foundry. Therefore, the recommendations and guidelines for authorizations as described in the [SAP Business Technology Platform Security Guide](#) also apply to SAP Intelligent Robotic Process Automation.

The [SAP Business Technology Platform](#) authorization concept is based on assigning authorizations to users based on roles. Roles are defined by SAP Intelligent Robotic Process Automation and cannot be modified.

## 6.1 Standard Roles

SAP Intelligent Robotic Process Automation uses the standard roles below.

Role	Description
IRPAProjectDelegate	<p>This role is used to monitor the executions (to use with read privilege upon a set of agents and/or environments).</p> <p>This role cannot create any project, entity, or register an agent.</p> <p>The projects, packages, environments, agent groups, and registered agents must be created before sharing with the corresponding rights (for example, view and edit).</p>
IRPAProjectMember	<p>Has all <code>IRPAProjectDelegate</code> rights plus creation and registration rights.</p> <p>Creates projects (directly or via desktop package import), and acts upon projects depending on privileges.</p> <p>Creates packages, and acts upon packages depending on privileges.</p> <p>Creates environments, and acts upon environments depending on privileges.</p> <p>Registers agents, and acts upon agents depending on privileges.</p> <p>Creates agent groups, and acts upon agent groups depending on privileges.</p>

Role	Description
IRPAOfficer	<p>Has all IRPAProjectMember rights.</p> <p>Has implicit MANAGE privileges on all objects (projects, packages, environments, agents, agent groups).</p> <p>Removes references to former employees.</p>
IRPAAgentUser	Registers and runs agents.
IRPAParticipant	Performs actions during a user task or a process execution.
Document_Information_Extraction_UI_Templates_Admin	This role must be assigned to users who build automations using the Document Information Extraction activities. Without access to this, user will be unable to access the Document Information Extraction interface for annotating templates.

## 6.1.1 IRPAProjectMember/Delegate and Privileges

### IRPAProjectDelegate

This role is used to monitor the executions (to use with read privilege upon a set of agents and/or environments).

The IRPAProjectDelegate role cannot create any project, entity, or register an agent. The projects, packages, environments, agent groups, and registered agents must be created before sharing with the corresponding rights (for example, view and edit).

### IRPAProjectMember

The IRPAProjectMember role has all IRPAProjectDelegate rights plus creation and registration rights.

IRPAProjectMembers can create any kind of object, an object being either a project, a package, an environment, an agent, or an agent group. Access to these created objects is governed by **Privileges**.

## Privilege

A Privilege is a permission given to a specific user or group of users on a specific object. Privileges make it possible to control precisely who has access to or can use specific projects, environments, or agents.

- **READ:**  
The READ privilege on an object gives the ability to inspect or use that object, but not to make any modification on it.
- **EDIT:**  
The EDIT privilege on an object includes the permissions of the READ privilege, and gives the ability to modify the object.
- **MANAGE:**

The MANAGE privilege on an object includes the permissions of the EDIT privilege, and gives the ability to share the object, and to manage its lifecycle (for instance, “releasing” a package requires a MANAGE privilege on the package).

- VIEW JOBS DATA:

The VIEW JOBS DATA privilege allows users to view the outputs of the jobs executed in the context of that environment. If the job outputs contain sensitive data, it may be necessary to precisely control who has access to these outputs.

**Note**

The “View Jobs Data” requires another privilege (READ, EDIT, MANAGE) on the environment, otherwise the user cannot access the jobs at all.

Here is the detail of the operations each privilege permits:

On Projects

Privilege	Authorized actions
READ	View the contents of the project, and the contents of its artifacts.  Copy a project or copy project artifacts.  See the defined privileges.
EDIT	Have the READ privilege.  Rename / change the description.  Delete the project.  Modify / delete the contents.  Test the project.  Create a PREVIEW package.
MANAGE	Have the EDIT privilege.  Share the project with teams or individuals.

On Packages

Privilege	Authorized actions
READ	View and use a package (e.g., within a project).  Use the public artifacts of the package in other projects when the user uses a package.  See the defined privileges.  Deploy the package in an environment.
EDIT	Have the READ privilege.  Rename / change the description.



Privilege	Authorized actions
MANAGE	<p>Have the EDIT privilege.</p> <p>Manage the package lifecycle: PREVIEW, RELEASED, DEPRECATED, DECOMMISSIONED.</p> <p>Share the package with teams or individuals.</p>
On Environments	
Privilege	Authorized actions
READ	<p>Environments:</p> <ul style="list-style-type: none"> <li>List the environments.</li> <li>View environments details.</li> <li>View environments privileges.</li> </ul> <p>Triggers and notifiers:</p> <ul style="list-style-type: none"> <li>List triggers and notifiers.</li> <li>View triggers and notifiers details.</li> <li>Run an existing trigger.</li> </ul> <p>Deployed packages:</p> <ul style="list-style-type: none"> <li>List deployed packages.</li> <li>View deployed packages details.</li> </ul> <p>Variables:</p> <ul style="list-style-type: none"> <li>List variables.</li> </ul> <p>Agents:</p> <ul style="list-style-type: none"> <li>List agents.</li> </ul> <p>Jobs (monitoring):</p> <ul style="list-style-type: none"> <li>List jobs.</li> <li>View jobs details.</li> <li>Download BAM CSV.</li> </ul>

Privilege	Authorized actions
EDIT	<p>Environments:</p> <ul style="list-style-type: none"> <li>Modify and delete environments.</li> </ul> <p>Triggers and notifiers:</p> <ul style="list-style-type: none"> <li>Create, modify, activate and delete triggers.</li> </ul> <div> <p><b>Note</b></p> <p>The Create action also requires the READ privilege on the package, if it has not been already deployed in the environment.</p> </div> <ul style="list-style-type: none"> <li>Create, modify and delete notifiers.</li> </ul> <p>Deployed packages:</p> <ul style="list-style-type: none"> <li>Deploy, modify, activate and undeploy packages.</li> </ul> <div> <p><b>Note</b></p> <p>(The Deploy action also requires the READ privilege access on the package, if it has not been already deployed in the environment.</p> </div> <p>Variables:</p> <ul style="list-style-type: none"> <li>Create, modify and delete variables.</li> </ul> <p>Agents:</p> <ul style="list-style-type: none"> <li>Add / remove agents and agent groups (provided the user has also the MANAGE privilege on the agent or agent group).</li> </ul> <div> <p><b>Note</b></p> <p>To be able to add an agent (or an agent group), the user must have the MANAGE privilege on it or it must be shared with the environment.</p> </div> <p>Jobs:</p> <ul style="list-style-type: none"> <li>Cancel jobs.</li> </ul>
MANAGE	<p>Have the EDIT privilege.</p> <p>Share the environment with teams or individuals.</p>
VIEW JOB DATA	View job logs.
On Agents	
Privilege	Authorized actions
READ	<p>View.</p> <p>View / download agent events.</p>

Privilege	Authorized actions
MANAGE	<p>Have the READ privilege.</p> <p>Disconnect / remove agents.</p> <p>Use the agent to execute jobs within an environment.</p> <p>Share the agent with teams, individuals or environments.</p> <div> <p><b>Note</b></p> <p>To be able to share an agent with an environment, the user must have the READ privilege access on the environment.</p> </div>

Agents may be directly shared with environments, so that they can be used within these environments. This makes it possible for agent managers to properly control in which context their agents may be used.

On Agent Groups

Privilege	Authorized actions
READ	View the agent group.
EDIT	<p>Have the READ privilege.</p> <p>Add / edit / remove nodes in the group structure, import a group as a CSV file, delete the agent group.</p>
MANAGE	<p>Have the EDIT privilege.</p> <p>Share the agent group with teams, individuals or environments.</p> <div> <p><b>Note</b></p> <p>To be able to share an agent group with an environment, the user must have the READ privilege access on the environment.</p> </div>

Agent groups specify collections of agents, that can then be used within environments. The agent specification is based on attributes of the agent such as the login name or the machine name.

## 6.1.2 IRPAOfficer

IRPAOfficer is the role of the person responsible for the SAP Intelligent Robotic Process Automation application at the customer level.

IRPAOfficers have implicit MANAGE privileges on all objects in the system.

IRPAOfficers can also perform operations such as:

- Setting the alerts (in Agents / Alerts);
- Viewing the logs (in Monitoring / Logs) and downloading them (in Monitoring / Data);
- Checking the resource consumption (in Monitoring / Consumption);
- Performing configurations (in Configuration);
- Removing references to former employees within the application tables.

#### ⓘ Note

This role is extremely powerful. To ensure proper separation of concerns, **only very few users** must have this role.

## 6.1.3 IRPAAgentUser

IRPAAgentUser is the role given to any user who needs to register an agent.

There are 2 main use-cases:

- Attended mode: the person triggering the jobs must have this role.
- Unattended mode: the person registering the agent must have this role.

This role doesn't give access to the SAP Intelligent RPA web application. Users that simply need to register an agent, for instance in a call center, use this role. IRPAProjectMembers and IRPAOfficers already have this capability.

#### ⓘ Note

Agents registered by IRPAAgentUsers cannot be directly managed by these users, since they don't have access to the SAP Intelligent RPA cloud application. These agents must be managed by IRPAOfficers, or IRPASystemManagers.

## 6.1.4 IRPAParticipant

The IRPAParticipant role is necessary for business users that read their IRPA user tasks in SAP BTP My Inbox.

Our Custom UI needs this role to get the content and bindings of the user tasks.

This role doesn't give access to the SAP Intelligent RPA web application.

## 6.2 Obsolete roles

The following roles were introduced in previous versions of Intelligent RPA.

They should be replaced by the `IRPAProjectMember` role and the management of privileges that provide much more control on the usage of resources.

You should now avoid using these roles because they give access to all objects of a given kind (packages, environments, ...) independently of the privileges.

Role	Description
<code>IRPASupervisor</code>	Monitors all the jobs and the status of <b>all</b> agents.
<code>IRPASystemManager</code>	Imports desktop packages. Has MANAGE privileges on <b>all</b> packages. Registers agents / creates agent groups. Has MANAGE privileges on <b>all</b> agents and agent groups. Creates environments. Has EDIT privileges on <b>all</b> environments. Monitors all job executions. Views / downloads all logs.
<code>IRPAPersonalDataAccess</code>	Gives access to job details.

`IRPASupervisor` is the role of IT people who need to check if jobs run properly and that resources are properly used.

`IRPASystemManager` is the role of IT people in charge of setting up the projects for test and production. They manage the packages and their configurations, together with the agent groups and environments. This role can still be useful to manage agents registered by `IRPAAgentUsers`, but is very powerful and must be used with care.

### 📌 Note

This role is not as powerful as the `IRPAOfficer` role, but is still very powerful. To ensure proper separation of concerns, make sure it is only given to users that really need it.

`IRPAPersonalDataAccess` gives access to ALL job outputs. By default, job outputs are not visible because they might contain sensitive or personal information. Instead of using this role, it is preferable to provide the "View Jobs Data" privilege on selected environments to selected users

### 📌 Note

The `IRPAPersonalDataAccess` role doesn't provide access to the SAP Intelligent Robotic Process Automation application. It complements the other roles, by providing access permission to sensitive or personal information.

# 7 Data Storage Security

[Desktop Studio Data \[page 30\]](#)

[Cloud Studio Data \[page 30\]](#)

[Cloud Factory Data \[page 30\]](#)

[Desktop Agent Data \[page 31\]](#)

## 7.1 Desktop Studio Data

The Desktop Studio is a development environment ran in the context of a Windows session. As such, it has the possibility to read, create, update, or delete any file in the file system, provided access to its location is permitted.


## 7.2 Cloud Studio Data

The Cloud Studio contains the definitions of all the artifacts that constitute projects. Since the 2.0 release, it provides the ability to create automations, and to capture applications. Application screenshots and captures are stored encrypted in the Cloud Studio database, using a customer specific AES-256 key.

## 7.3 Cloud Factory Data

Apart from credentials, all data stored by the Factory (packages, configurations, environments, agent groups...) are stored in databases or services managed and operated by [SAP Business Technology Platform](#). Access to these data is governed by the user roles. No specific configuration is required beyond role assignment.

Most of the data in the databases such as object names and descriptions, attributes and their values (tags) are stored unencrypted. These elements should not contain any sensitive information. Only credential variables coming from the Desktop Studio, Cloud Studio variables ("regular" variables or [environment](#) variables), API keys, job outputs, logs and traces are stored encrypted.

Customer specific encryption keys are stored using the [SAP credential store](#)  provided by the SAP Business Technology Platform. See the [SAP Business Technology Platform Credential Store](#) online help to get more information.

## 7.4 Desktop Agent Data

### 7.4.1 Desktop Agent Work Directory

Desktop Agents need "read and write" access (permission, privilege) to the %localappdata%\SAP\Intelligent RPA\Projects directory, %localappdata% being the standard Windows environment variable pointing to a user and machine dependent folder. This folder contains different types of content:

- Project files are provided through interactions with the cloud application. They consist of the JavaScript files developed using the Desktop Studio. These files should never be modified; modification causes the integrity check to fail.
- Logs and traces generated by the agent are also stored in that directory.
- Business Analysis Metrics (BAM) cache files are located under log\localCache. These files should not be manually modified to avoid sending inaccurate information to the cloud.

#### Note

When executing Desktop Projects or Cloud Projects using the core SDK version 1.10 or older:

- Since the agent runs with the privileges of the Windows user, the user may view the contents of the project, logs, trace files and cached BAM files.
- Logs, trace files and cached BAM files may contain personal data. Access to these files should be limited and they should be deleted when no longer needed.

### 7.4.2 Storage of Authentication Tokens

When registering an agent, an authentication token identifying the agent is generated. This token depends on the Windows session user and the machine hardware. It is stored in the Windows Credential Manager and is thus only accessible to the logged-in user.

### 7.4.3 Access to the File System

To execute jobs, agents may need to access files located anywhere on the disk. This need depends on the actions performed by the agent.

#### ⚠ Caution

Projects should thus be reviewed to ensure that they are not vulnerable to path traversal attacks, where malicious user input could cause the agent to read or write to unexpected locations (even if the right to read or write to these locations will always be limited by the agent's user rights).

## 7.4.4 Mass Registration

Mass registration tokens must be stored in a place that is accessible from the different agents that need to register.

#### ⚠ Caution

The mass registrations must be located in a place where they can be read easily, but not deleted or replaced. Only administrators dealing with the deployment of Intelligent RPA agents must have the rights to modify or delete these files.

## Related Information

[Mass Agent Registration](#)

## 7.4.5 Local Agent Variables and Credentials

[Local Agent Variables](#) - A set of activities called "Local Agent Variables" allow to create, update, retrieve and delete local variables. Such variables are then stored within the Windows registry of the local computer (inside a dedicated registry folder).

[Local Agent Credentials](#) - A set of activities called "Local Agent Credentials" allow to create, update, and retrieve local credentials leveraging Windows Credential Manager. For more information about Windows Credential Manager, please visit the Microsoft website.

#### ⚠ Caution

Note that Local Agent Variables and Credentials will not be removed if the agent is uninstalled. Therefore, it is important not to store personal or sensitive information in the variables. It is also important to make sure that the credentials are removed as soon as they are no longer needed.



# 8 Data Protection and Privacy

## 8.1 Introduction

This section provides information about how SAP Intelligent Robotic Process Automation complies with data protection requirements.

Data protection is associated with numerous legal requirements and privacy concerns. In addition to compliance with general data privacy acts, it is necessary to consider compliance with industry-specific legislation in different countries. SAP provides specific features and functions to support compliance with regards to relevant legal requirements, including data protection. This section describes the specific features and functions that SAP provides to support compliance with the relevant legal requirements, including data privacy.

SAP does not give any advice on whether these features and functions are the best method to support company, industry, regional or country-specific requirements. Furthermore, this guide does not give any advice or recommendations in regard to additional features that would be required in a particular environment; decisions related to data protection must be made on a case-by-case basis, under consideration of the given system landscape and the applicable legal requirements.

### Note

In the majority of cases, compliance with applicable data protection and privacy laws will not be covered by a product feature alone. SAP software supports data protection compliance by providing security features and specific data protection-relevant functions. SAP does not provide legal advice in any form. Definitions and other terms used in this document are not taken from any given legal source.

## 8.2 Glossary

Term	Definition
Consent	The action of the data subject confirming that the usage of his or her personal data shall be allowed for a given purpose. A consent functionality allows the storage of a consent record in relation to a specific purpose and shows if a data subject has granted, withdrawn, or denied consent.
Deletion	The irreversible destruction of personal data.
Personal Data	Any information relating to a data subject.

Term	Definition
Purpose	The information that specifies the reason and the goal for the processing of a specific set of personal data. As a rule, the purpose references the relevant legal basis for the processing of personal data.
Retention period	The period of time between the end of the last business activity involving a specific object (for example, a business partner) and the deletion of the corresponding data, subject to applicable laws. The retention period is a combination of the residence period and the blocking period.

## 8.3 Legal Ground for Personal Data

SAP Intelligent Robotic Process Automation Factory uses the identifiers of its users – usually an email address – for authentication. These identifiers are stored with resources (projects, project artifacts, packages, environments...) to monitor who has created such resources, and who was the last user to modify them. SAP Intelligent Robotic Process Automation Factory also stores the logins of the Windows users as well as the machine and domain names related to the registered agents.

In both cases, usage of this information is allowed per the working contract between the users and the customer. On contract termination, references to a user should be removed from SAP Intelligent RPA Factory by an administrator, thanks to the [Data Protection and Privacy: Clearing User Data](#) feature in SAP Intelligent Robotic Process Automation Factory.

Any other personal data should only be processed in the context of SAP Intelligent Robotic Process Automation if the customer has a legal ground to do so. It is the responsibility of the customer to ensure that the data processing activities developed and executed using SAP Intelligent Robotic Process Automation comply with the applicable regulations.

## 8.4 Read Access Logging

As the only personal information stored in the SAP Intelligent RPA Factory application is the user's identifiers, there is no need for read access logging.

### ⚠ Caution

Job execution logs might contain sensitive or personal information, and on the agent machine, logs can be read by the Windows user running the agent. To limit the risk of disclosing (sensitive) personal information to unauthorized people, check that the project code never logs (sensitive) personal information. Do not activate logs auto-recording or technical trace recording in production environments, without having put in place the necessary measures to control the access to the generated logs.

### ⚠ Caution

Logs displayed in the monitoring section of the Cloud Factory might also contain personal information. Such information might be present in the job outputs and is not visible by default. To view it, the user must have the "View Jobs Data" privilege on the Environment (preferably), or the `IRPAPersonalDataAccess` role.

### ⚠ Caution

Intelligent RPA cannot track personal data entered in free form inputs, or in screenshots and captures. It is the customer's responsibility to check that personal information is not entered there or to track them appropriately. Test systems with fake users must be used instead of production systems during the application capture process, when captures show records containing information about individuals.

## 8.5 Change Log

For auditing purposes or for legal requirements, changes made to personal data should be logged, enabling the monitoring of who made changes and when.

The only personal data stored in the SAP Intelligent Robotic Process Automation Factory application are the e-mail addresses of users.

A change log is thus not applicable as e-mails can't be changed in the system.

## 8.6 Deletion of Personal Data

When handling personal data, consider the legislation in the different countries where your organization operates. After the data has passed the end of purpose, regulations may require you to delete the data.

Additional regulations may require you to keep the data longer after the end of purpose. During this period you must block access to the data by unauthorized persons until the end of the retention period, when the data is finally deleted. As the only personal data stored in SAP Intelligent Robotic Process Automation Factory are the e-mail addresses of users, a retention period is not applicable in this case. SAP Intelligent RPA administrators should delete personal data when the end of purpose has been reached with the [Clear User Data](#) feature in SAP Intelligent RPA Factory.

### Data Retention Durations

Information Type	Retention Duration
Backups	21 days
Job inputs and outputs	3 months (107 days including backups)

Information Type	Retention Duration
Job archives	24 months
<b>Note</b> Job archives do not include business data.	
All data following the end of the provision of services	51 days (including backups)
References to SAP Intelligent RPA following deletion	21 days (including backup retention)

### ⚠ Caution

When executing Desktop Projects or Cloud Projects using the core SDK version 1.10 or older:

- Log and trace files, as well as cached BAM files (located under %localappdata%\SAP\Intelligent RPA\Projects\<project>\log) may contain personal data (logins and machine names as a minimum). Access to these files should be limited and they should be deleted when no longer needed.
- These files only have a limited lifetime on the agent's machines, as long as the agent is running. If the agent is stopped, no mechanism can clean these log files. Make sure that log files are properly deleted if an agent is definitely stopped.

## 8.7 Automated Decision

Some regulations control the way automated decision may impact individuals.

For instance, article 22 of the General Data Protection Regulation states that:

1. The data subject shall have the right not to be subject to a decision based solely on automated processing, including profiling, which produces legal effects concerning him or her or similarly significantly affects him or her.
2. Paragraph 1 shall not apply if the decision:
  - a) is necessary for entering into, or performance of, a contract between the data subject and a data controller;
  - b) is authorised by Union or Member State law to which the controller is subject and which also lays down suitable measures to safeguard the data subject's rights and freedoms and legitimate interests; or
  - c) is based on the data subject's explicit consent.
3. In the cases referred to in points (a) and (c) of paragraph 2, the data controller shall implement suitable measures to safeguard the data subject's rights and freedoms and legitimate interests, at least the right to obtain human intervention on the part of the controller, to express his or her point of view and to contest the decision.
4. Decisions referred to in paragraph 2 shall not be based on special categories of personal data referred to in Article 9(1), unless point (a) or (g) of Article 9(2) applies and suitable measures to safeguard the data subject's rights and freedoms and legitimate interests are in place.

Article 15 states that: The data subject shall have the right to obtain from the controller confirmation as to whether or not personal data concerning him or her are being processed, and, where that is the case, access to the personal data and the following information:

[...]

8. the existence of automated decision-making, including profiling, referred to in Article 22(1) and (4) and, at least in those cases, meaningful information about the logic involved, as well as the significance and the envisaged consequences of such processing for the data subject.

When using Intelligent RPA, it is the responsibility of the customer to ensure that the usage of the automations complies with the applicable regulations in this respect.

## 9 Security-relevant Logging and Tracing

Security-relevant events raised in SAP Intelligent Robotic Process Automation Factory are logged in the audit log service provided by SAP Business Technology Platform.

Package creation or deletion events, package configuration events, environment creation, modification or deletion events are logged in this audit log.

For more information about the ways to access this information, please refer to the [audit log guide](#).

### Note

If the SAP Audit Log Service is unavailable, the audit messages will be stored in the SAP Intelligent RPA database until they can be sent to the audit log service. If the unavailability of the SAP Audit Log service lasts for more than 12 hours, customers will be warned.

# 10 Frequently Asked Questions

[Data Security \[page 39\]](#)

[Agent Permissions \[page 41\]](#)

[Network Security \[page 42\]](#)

[Customer Data Export \[page 43\]](#)

## 10.1 Data Security

### Which data are exchanged between the agent and the Factory at runtime?

From the agents to the Cloud Factory:

- A machine unique identifier (currently computed using a unique BIOS identifier and the "*Boot volume Device ID*").
- The agent user login with and without the domain name (the login is used in the agent group definitions).
- The machine name, as used in the agent groups.
- The agent version and various technical information associated with the agent (e.g. the studio mode).
- Information about the state of the agent (busy or not) and its execution mode (*attended* or *unattended*).
- Identifiers of various SAP Intelligent RPA objects such as packages, environments, resources, scenarios and automations.
- Business data as defined and required by the project:
  - Data used as inputs to scenarios or automations
  - Job outputs
  - "*Business Activity Monitoring*" notifications
- Encrypted logs and traces for Cloud Projects using the core SDK version 1.11 or newer.

From the Cloud Factory to the agents:

- The JSON Web Token used to authenticate the agent.
- The public key that corresponds to the customer's private key, generated in the Cloud Factory.

#### Note

This key is used to verify the integrity of the **bundles** containing the Javascript code.

- Identifiers of various SAP Intelligent RPA objects such as packages, environments, resources, scenarios and automations.

- The list of all usable packages. For each package:
  - A unique identifier, type and version.
  - A unique environment identifier.
 For V2 packages:
  - The list of its dependencies.
  - The signature of the bundle associated to the package.

#### 📌 Note

Each dependency bundle contains information (file name, mime type and size) about the packages and bundles to download.

- Business data:
  - The contents of the bundles or packages.
  - Input data used to execute a scenario or an automation (for unattended cases).
  - The results of the jobs triggered by a scenario ([addJob](#)) when they are needed to pursue its execution.
  - Environment variables values as requested in the course of a scenario or automation execution.

## **Some logs sent to the Cloud Factory monitor the status of agents, jobs, etc. Which data of the items stored in the Factory are accessible by SAP? Which logs, events, etc. can SAP trace and under which circumstances?**

These logs belong to the customer and are classified as confidential. As mentioned above, logs contain technical (start and stop events, package distribution, etc.) and business data.

All this information may be viewed by the customers from the SAP Intelligent RPA user interface (this sometimes requires the "[IRPAPersonalData](#)" role or privilege). SAP does not collect information that would not be visible there, and SAP employees do not have access to that information, unless they are platform administrators or support engineers, and thus are granted access to the databases. This access may be required for troubleshooting or maintenance tasks. Access to the production systems and databases is controlled, and access control lists are regularly reviewed.

Note that business data are encrypted by the application and cannot be accessed in clear text by administrators or support engineers when they perform maintenance or troubleshooting operations.

SAP also has access to technical logs that are being generated during the execution of the SAP Intelligent RPA software. These logs are used to troubleshoot the product and analyze bugs. They do not contain information that belongs to the customer.



## 10.2 Agent Permissions

### What is the agent service?

In unattended mode, if for some reason a machine running an agent is restarted, the session that is running the agent can be automatically reopened, and the agent restarted. This capability can be configured via the agent systray (*auto-start*).

To do so, when the agent is installed on a machine, a service named `SAP Intelligent Desktop Service` is also installed. This service is running under the local system identity but it is completely independent of the agent itself, that runs under the session user identity.

To open the session, the service needs the user's credentials. These credentials need to be provided when the *auto-start* option is selected (this is currently done by using the `CxCredStore` utility). They are stored in the Windows credential manager (and never sent to the cloud). When the machine is restarted, the service is restarted as well: it can access these credentials, reopen the user's session, and start the agent.

### **Is there a mechanism to restrict the agent's access to elements of the desktop operating system? Is there a mechanism within current design that would allow whitelisting or restrictions of tasks Desktop Agent can perform?**

The agent leverages the current security afforded to the logged-in session user. Therefore, the agent is limited to the same access as the user. If the user cannot access certain file shares because they lack permissions, the agent will be prohibited in the same manner.

To execute jobs, agents may need to access files located anywhere on the disk. This need depends on the actions performed by the agent. Projects should thus be reviewed to ensure that they are not vulnerable to path traversal attacks, where malicious user input could cause the agent to read or write to unexpected locations (even if the right to read or write to these locations will always be limited by the agent's user rights).

The agent's security profile can be strengthened thanks to standard best practices. For example, the user must not run in an administrator's session, must secure from path traversal attacks, delete local logs when no longer needed, etc.

**What is the mechanism to ensure that only a specific agent, which is running within a specific user's security context, will acquire and execute jobs? (i.e. The user system allows multiple users to log in. What would be the mechanism that ensures that a job created to open all excel files from home directory will be executed only within a specific user's context?)**

When a user registers an agent, an authentication token identifying the agent is generated. This token depends on the Windows session user and the machine hardware. It is stored in the Windows Credential Manager and is thus only accessible to the logged-in user. This way, it is possible to relate an agent and the corresponding user in the SAP Intelligent Robotic Process Automation Factory.

Packages are not deployed on all available agents, but only on those that have been defined in the deployment environment in the SAP Intelligent Robotic Process Automation Factory, either directly, or via agent groups. This is where you can control which agent or group of agents will be able to execute which automation.

## 10.3 Network Security

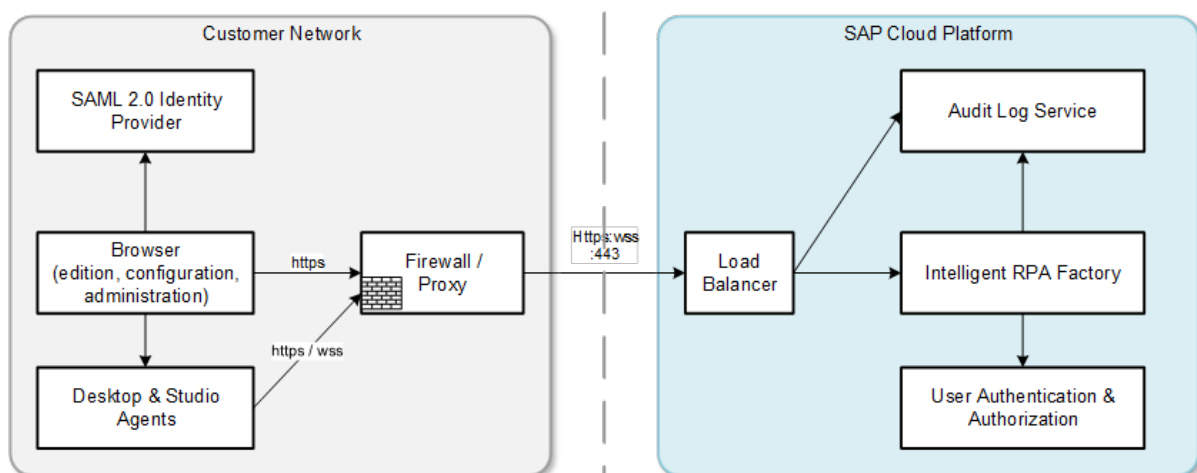
### Which protocols are used?

HTTPS is used for the communication between:

- The SAP Intelligent RPA web application and the SAP Intelligent RPA cloud backend.
- The SAP Intelligent Robotic Process Automation Desktop Agent and the SAP Intelligent Robotic Process Automation Factory Runtime at registration time.

WSS (secure web sockets, based on TLS 1.2) is used for the communication between the SAP Intelligent Robotic Process Automation Desktop Agent and the SAP Intelligent Robotic Process Automation Factory Runtime (project download, commands).

All communications between the Desktop Agent and the SAP Intelligent Robotic Process Automation Factory go through the 443 port. You do not need to open another outbound or inbound port.



## Is it possible to use a proxy? Is there any specific required configuration?

SAP Intelligent RPA can be used through a proxy. When running behind a proxy or a firewall, make sure that the WebSockets secure protocol (WSS) is properly supported and not blocked.

## Is it possible to allowlist the IP addresses that are used by SAP Intelligent Robotic Process Automation Factory?

It is possible to further limit the outbound communications by granting access only to specific IP addresses. These IP addresses depend on the data center and are listed in the SAP Business Technology Platform documentation. To find the IP addresses relevant for your regions, see: [Regions](#).

The IP addresses to allowlist are indicated in the LB IPs (ingress, for incoming requests) column, corresponding to your data center. \*.<data center>.hana.ondemand.com addresses must be mapped to these IP addresses, where <data center> must be replaced by the appropriate designation of your data center (eu10 and so on).

It is also necessary to allow the IP used to retrieve the SAP UI5 framework (used for the user interface of the cloud solution). This framework is retrieved from the `sapui5.hana.ondemand.com` domain. This domain is cached at different locations. Use `nslookup` or a similar utility (from your DMZ) to determine the appropriate IP address.

## 10.4 Customer Data Export

As mentioned in the **Terms and Conditions for Cloud Services**, customers can export their data at any time during the subscription term.

The article 4.5 also also mentions that a "Customer may export and retrieve its Customer Data in a standard format. Export and retrieval may be subject to technical limitations, in which case SAP and Customer will find a reasonable method to allow Customer access to Customer Data".

SAP Intelligent RPA provides means to export projects, packages and logs directly from the user interface.

If SAP Intelligent RPA customers need to export all their data, they can request a data export by raising a [Ticket](#) on component **CA-ML-IPA**.

### Note



The exported data cannot be imported back into SAP Intelligent RPA, either into the same or a different subaccount.

# Important Disclaimers and Legal Information

## Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.

About the icons:

- Links with the icon  : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon  : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

## Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

## Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

## Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

## Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.



© 2024 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see <https://www.sap.com/about/legal/trademark.html> for additional trademark information and notices.