inception
WEB POWERED SECURITY
Simple & Easy Installation
Integrated Security - Access Control
Dashboard

Inception REST API Release Notes

Contents

Introduction to the Inception REST API

Inception supports third-party integration through a REST API, introduced in firmware version 3.0. The API provides various features useful for integration including user management, area/door/output control, and live monitoring of item state and activity progress (through HTTP long polling). Newer revisions of the API may add new features and modify existing ones; however, efforts are made to retain backwards compatibility wherever feasible.

As of firmware version 3.1, the current API protocol version can be retrieved from the **GET /api/protocol-version** endpoint, no authentication is required to obtain a response. The response JSON object should contain the current protocol version of the API, a number that is incremented with each revision of the REST API. Developers should be able to check this version number (or the 404 response indicating a pre-3.1 firmware) in combination with the REST API release notes to determine which API features are available in the installed Inception firmware, and whether to disable certain integration features or require the user to update their Inception firmware before using the integration.

These release notes summarise the API protocol additions and changes that have been made in each release; for technical details of how to use the API, including details of the JSON type formats used to send and receive data, see the on-board HTML documentation on an Inception device running up-to-date firmware. The documentation can be found at http://[inception-hostname]/ApiDoc (where [inception-hostname] is the IP address or hostname of your Inception device).

For more information about integrating with Inception and the REST API, see the Inner Range website at https://innerrange.com/Products-Solutions/Inception#Integrations.

Current Release

Protocol Version 16 (Firmware 5.2.0)

Feature Updates

New UnitPublicState: SecurePending

Storage Units have had their default behaviour changed to only secure the unit once all its inputs have secured. If any of its Inputs are in Alarm when the Secure command is sent, the Storage Unit will instead move to a *Secure Pending (Active Inputs)* state. The unit will remain in this state until either unlocked, or all its inputs are sealed, in which case it will transition automatically to the *Secure* state.

To accommodate this change in behaviour, a new *SecurePending* flag has been added to the *UnitPublicState* enum, with an Integer value of 256. For example, if a Storage Unit has a *UnitPublicState* of 260, this equates to the Storage Unit being in the *Open* (4) and *SecurePending* (256) states.

Separate Door Permissions for Entry vs Exit

The *Access* door permission has been broken into two separate permissions, *Entry* and *Exit*, which can be assigned separately or together to Permission Groups. Existing Door *Access* permissions will be automatically converted to *Entry + Exit* permissions upon upgrading to this firmware version or later.

To accommodate this change, two new flags have been added to the *PermissionAbilities* enum: *DoorEntry (256)* and *DoorExit (512).* This *PermissionAbilities* object is returned when performing a **GET /control/door/summary**, which returns the list of Doors that the requesting User is allowed to control, and what actions (*PermissionAbilities*) the user is allowed to perform on each of those Doors.

To ensure backwards compatibility with existing integrations, the *Access (8)* flag will be automatically added to the returned *PermissionAbilities* if both *DoorEntry* and *DoorExit* are present.

Issues Resolved

DataChange Responses for Output Changes

Resolved an issue where data changes were not coming through when an Output’s configuration was changed. Performed using **POST /monitor-updates** and a *RequestType* of “DataChange”, the targeted entity type will be long polled for any relevant changes since the specified *timeSinceUpdate*. While this worked correctly with all other supported entity types, attempts to perform this on the Output would result in a blank 200 OK response after one minute. As of this protocol version, the response returned when monitoring for data changes on Outputs will be consistent with responses for other entity types.

Past Releases

Protocol Version 15 (Firmware 5.1.2)

Issues Resolved

Longpoll Review Events Issue

The ApiReviewObject's WhenTicks property has been deprecated and replaced with a new ReferenceTime field. WhenTicks was intended to be used with the Live Review Events long polling feature of the API, using the WhenTicks and the ID of the last received review event as the reference time and ID for requesting new events.

As WhenTicks was of the Long Integer type, it would occasionally be incorrectly rounded, resulting in issues where Inception could not properly locate the referenced review event and the long poll feature appearing to stall.

Integrations should instead use the ReferenceTime field added to ApiReviewObject for this API feature to work correctly. ReferenceTime also returns the tick count of when the event occurred, however as type String, removing the possibility for rounding errors.

The API documentation has been updated to mark WhenTicks as deprecated and ReferenceTime as the correct property to use, however WhenTicks will continue to retain its existing functionality in order to support existing integrations that use this feature.

Protocol Version 14 (Firmware 5.1.0)

Feature Updates

Door Denied Access Events in API User Info Monitoring

The ‘DoorDenied’ event can now be added into the eventTypes for monitoring user info updates, monitoring whenever a Door Access Denied event occurs.

Protocol Version 13 (Firmware 5.0.0)

New Features

User API Tokens

Authorisation can now be done through a User API Token which will make requests on behalf of the user it was generated for. Attainable by generating the key in the web interface through **[Configuration > Users > Manage Users > Credentials > User API Token]** and can be used by adding it to the authorisation header with type “APIToken”.

Storage Unit Support

Storage units are a new feature in Firmware version 5.0.0, and will include the typical endpoints for controllable items such as Get Storage Unit, Get Storage Unit State, and Get Storage Unit Summary. Storage units can also be controlled by the API to secure, unlock, mark and clear the storage unit through the **POST /api/v1/control/storage-unit/{{storage_unit_id}}/activity** endpoint.

Link SkyCommand

A new endpoint to link a user to SkyCommand has been added (**POST /api/v1/config/user/{id}/skycommand**) which takes an “Email” parameter which corresponds to the SkyCommand account being linked.

Feature Updates

Door Attached Reader Names

Whenever a door attached reader is returned by a request, it will now include the Name property of that DoorAttachedReader.

Protocol Version 12 (Firmware 4.2.0)

Feature Updates

Control Doors – Cancel Access Requests

The Control Door Activity now has a new “CancelAccessRequests” option for DoorControlType, which cancels all current user access requests for that door.

Protocol Version 11 (Firmware 4.1.1)

Feature Updates

Control Doors – New Door Control Types

The Control Door Activity now has a new “ToggleLock” option for DoorControlType, which locks the door if it is currently unlocked, and vice versa.

A new “MuteHeldResponse” DoorControlType has also been added to allow the DOTL/Door Held Open response to be muted for a door.

The DoorPublicStates flag value returned from **GET api/v1/control/door/{id}/state** also includes a new flag for HeldResponseMuted (0x0400), which is set to true when the Door’s Held Response has been muted.

Configure Users – Generate Unused PIN

A new endpoint has been added to allow generating a unique PIN for a user **(GET api/v1/config/user/generate-unused-pin)** to make it easier to select a unique PIN when creating or editing users. A min/max length can be optionally specified using query parameters, otherwise the system’s default PIN length is used.

Protocol Version 9 (Firmware v4.1)

Feature Updates

Activities – Remote Control option

Activity objects (e.g. Control Area, Control Door) can now indicate that the activity is originating from a remote source by setting the “**RemotelyControlled**” boolean property to true (mostly relevant for Area Arm activities). This property works in tandem with the “Warn Before Remote (Off-Site) Area Arm” property on Inception’s System Settings page in order to warn on-site users by causing the area to enter an Arm Warning phase before the remote arm activity is carried out.

Protocol Version 8 (Firmware v4.0)

New Features

User Photos

User photos can now be retrieved, added, modified and deleted with the REST API, by using the **GET/POST/DELETE api/v1/config/user/{id}/photo** endpoints.

Feature Updates

Activities – Control All option

The Control Area, Control Door, and Control Lift Floor activities can now control all visible items of their type in the system by setting the "**ControlAll**" boolean property in the Activity JSON object to "true". Example requests have been added to the Inception REST API Postman request collection to demonstrate the new functionality.

It is recommended that "Control All" activities are sent to the **POST api/v1/activity** endpoint as opposed to a particular item's URL like **POST api/v1/control/area/[id]/activity** since the request may fail with a 404 response if the specific item ID is inaccessible or deleted.

API Usage over SkyTunnel – Multiple Servers/Redirection

The REST API for an Inception controller can be accessed from anywhere in the world by using Inner Range's free SkyTunnel service, if the controller is configured to allow HTTP access over SkyTunnel.

Due to popularity, Inner Range's free SkyTunnel service has been upgraded to include multiple servers to support increased scale and geolocation optimization. This change affects integrations that communicate with the API over SkyTunnel. Requests sent to a SkyTunnel URL (for example: **https://www.skytunnel.com.au/inception/IN001234**) may respond with a 307 Redirect status code containing the instance-specific URL in the "Location" header. There is no guarantee that an Inception controller will always remain connected to the same SkyTunnel instance, so integrations must make sure that redirect responses are followed correctly.

Issues Resolved

Door Summary – Permissions Issue

Fixed an issue where the door summary endpoint (**GET api/v1/control/door/summary**) would be inaccessible to users in some situations even if they have full door control permissions.

Protocol Version 7 (Firmware v3.3.0)

Feature Updates

Activities – Reporting ID support

When performing activities on specific items in the system (**POST api/v1/activity**) you can now use the item's "ReportingID" to specify the item instead of the item's Guid ID. This is intended to make it easier to use the ReportingID as an alternate identifier for systems that have difficulty using Guid identifiers. This is applicable for the following activity types:

- Control Area
- Control Door
- Control Input
- Control Custom Input
- Control Output

- Control Lift Floor

Control Areas – Associated Input Reporting IDs

The AreaAssociatedInput type returned by the **GET api/v1/control/area/{id}/inputs** now contains the Input's ReportingID for convenience. This also applies to the Associated Inputs contained in the Area Summary.

Update Monitor – Monitor Entity States

The Monitor Entity States long poll response now includes the ReportingID of each item in the response's stateData array, to make it easier to identify item state updates in systems that prefer to use the Reporting ID over the Guid ID.

Protocol Version 6 (Firmware v3.2.3)

New Features

Configure Users – Get User by PIN

Users can now be looked up via their PIN via the **GET /api/v1/config/user/by-pin/{pin}** endpoint. The results of the query can be filtered to specific User fields with the "includedProperties" query param to reduce response size. Note that all requests made to this endpoint will be logged in Review for security purposes.

Feature Updates

Configure Users – Get User

The **GET /api/v1/config/user/{id}** endpoint now supports the optional "includedProperties" query param (already in **GET api/v1/config/user**) which can filter response data to just the requested properties in order to reduce response sizes.

Control Lift Floor – Timed Free Access

The Control Lift Floor activity object (IRApiControlFloor) now has a new TimeSecs property to allow you to specify a period of time when putting a Floor into Timed Free Access mode.

Protocol Version 5 (Firmware v3.2.1)

New Features

Authentication – Hashed Login

REST API clients can now alternatively login using a SHA-256 hashed password (generated by taking the base64-encoded SHA-256 hash of "[username]:[password]") for additional security by using the **POST /api/v1/authentication/hashed-login** endpoint. See the Authentication section of the on-board REST API docs for more information.

Feature Updates

Configure Users – Read/Write Encrypted Passwords and PINs

The **/api/v1/config/user** endpoints previously did not support retrieval of user PINs and passwords (the User DTO's SecurityPIN and WebLoginPassword fields return "********" instead of the actual data), but it is now possible to retrieve PINs and passwords in an encrypted form by reading the "EncryptedSecurityPin" and

"EncryptedWebLoginPassword" fields in the ClientUserDto object returned by the **GET api/v1/config/user/{id}** endpoint. These encrypted string fields can also be written to when creating or updating users via the REST API, for example, in the case where you want to use the REST API to retrieve users from one Inception system and create copies of them in another Inception system while retaining the data in encrypted fields.

See the Configure User page and the ClientUserDto data model page in the REST API docs for more information on these fields.

Control Areas – Temporarily Isolate Issues when Arming

The Control Area Activity has gained a new boolean property named "TemporarilyIsolateIssues" which will cause an Arm attempt to temporarily isolate any current issues (e.g. unsealed inputs, modules offline) until the arm cycle ends. Previously, arming an area via the REST API would instead perform an "auto-isolate" on affected system entities, which would be de-isolated once the fault is next sealed or resolved. This behaviour was not suitable for cases such as faulty inputs, as a subsequently re-triggered input after de-isolation could cause a false alarm in the area. It is therefore recommended that the "TemporarilyIsolateIssues" property is set to true when you want to arm an Area using the REST API when there are still issues present.

Protocol Version 4 (Firmware v3.2)

New Features

Activities - Virtual Badging of Credentials/PINs at Readers

The API now supports virtual credential badging at door readers, enabling external systems (e.g. facial recognition, fingerprint scanners) to virtually present credentials at door readers by sending activity objects containing credential data (IRApiBadgeCredentialAtReader and IRApiSendPINDataToReader for credentials and PINs respectively) to the **POST /api/v1/activity** endpoint. Virtual credential badge events are processed identically to physical badge events, e.g. virtually badging "Card 1234" belonging to John Smith at a reader attached to Door 1 will attempt to grant access to the door based on the permissions of John Smith as though he physically presented the card at the reader, and trigger a success or fail response depending on if he was allowed through or not. This preserves all of the access and automation logic associated with accessing the door/area.

Information about using these new activity types can be found on the "Activities" page of the on-board REST API documentation (http://[YourInceptionAddress]/ApiControllerDoc?controller=api/v1/activity where [YourInceptionAddress] is the hostname/IP of your Inception system), and also the examples in the Postman request collection.

Feature Updates

Item Summaries

New "**GET /summary**" endpoints have been added for Controllable Inputs, Outputs, Lift Floors and Users. These endpoints, like the existing Area and Door ones, return a summary of all the controllable items of that type plus any additional data relevant to those items, packed into a single JSON response to make it easier to query for item information without incurring the overhead involved with having to send many requests per item.

Additionally, all Summary item entries now contain 2 new fields, the LastStateChangeTime and Permissions fields. The LastStateChangeTime is a timestamp representing the last time the item's state was modified, which can be passed as a reference point when long polling for new item state changes; and the Permissions field is an array of integer flags containing all of the permissions that the REST API user currently has in relation to the object, such as Arm/Disarm permissions for an Area or Access permission for a Door.

Historical Review Events

The “**GET /api/v1/review**” endpoint has gained 2 new query parameters (referenceId and referenceTime) which are intended to be used to specify an existing review event as an alternative to specifying a time range with the start/end parameters.

Configure Credential Templates

Card Templates have been renamed to Credential Templates to reflect the new functionality available in firmware v3.2. The “Configure Credential Templates” endpoint root URL has also changed from “**GET /api/v1/config/card-template**” to “**GET /api/v1/config/credential-template**” to reflect this change. The old endpoint is now deprecated, but will still work for existing integrations; new integrations should use the new endpoint URL(s).

Configure Users

The Credential DTO type (used by the “Credentials” property in the User DTO type) has gained a new “State” property which allows API users to enable and disable Users’ credentials (via the **UPDATE** or **PATCH** methods) if they are lost, stolen, etc. See the [**Data Models > CredentialState**] page in the onboard Inception REST API documentation to see a list of all supported Credential State values.

Postman Request Collection

The Inception Postman request collection (downloadable from the Example API Requests section of the REST API documentation) has been updated to include new sample requests for new and updated features.

Protocol Version 3 (Firmware v3.1.1)

Feature Updates

Authentication

The session ID can now be passed as a URL query parameter (“session-id”) instead of in the HTTP request’s Cookie header, for example: “**GET /api/v1/control/door?session-id= 27cc29d4-3607-4f3c-a352-94be0b197933**”. This method of passing the ID works with all Inception API requests and is intended for use in environments where it is not possible to edit HTTP request headers.

Protocol Version 2 (Firmware v3.1)

New Features

Protocol Version endpoint

A new endpoint has been added (**GET /api/protocol-version**) to return the current protocol version of the Inception REST API (currently at protocol version 2). This protocol version number will be incremented with each API revision and is intended to help developers determine whether newer endpoints are available on the currently loaded Inception firmware version or not. This method can be called without authentication.

System Info endpoint

A new endpoint has been added (**GET /api/v1/system-info**) to enable querying for information about the Inception system, including the system’s name and serial number.

Historical Review Events

Historical Review Events are now able to be retrieved from the new **GET /api/v1/review** endpoint. The endpoint supports optional parameters for start/end time range, limit/offset number of results, search direction; and filtering based on event category, event type, and involved entities.

Control Doors

Doors' attached readers can now be retrieved from the new **GET /api/v1/control/door/attached-readers** endpoint.

Area and Door Summaries

New endpoints have been added at **GET /api/v1/control/door/summary** and **GET /api/v1/control/area/summary** which return a data summary of all the controllable Doors and Areas in the system respectively, and all of the supplementary information available for them (e.g. door attached readers, area associated inputs). These new endpoints are intended to reduce the number of requests needed to retrieve large amounts of item information by combining them into a single request, particularly in network environments where making many HTTP requests per item is slow and costly.

Feature Updates

User States

The **UserPublicStates** flag set has been extended to include the user's current Anti-Passback Violation state and Anti-Passback Forgiveness state as separate flags. User's state can be requested from the **GET /api/v1/control/user/{id}/state** endpoint, or through real-time state updates from the Update Monitor.

Control Areas

The JSON response returned by the **GET /api/v1/control/area/{id}/inputs** endpoint now also includes the associated input's Name (string) field for convenience.

Item States

For item types with state information available (through the **GET /api/v1/control/{item-type}/{id}/state** endpoint), the JSON response object now contains the Info1 and Info2 string fields that may contain additional contextual state info based on the item type, e.g. the next scheduled arm time for Areas.

Activities

The **GET api/v1/activity/{id}** endpoint now returns a more streamlined list of activity update messages (in the new ActivityProgressMessage type format).

Some Activity updates now contain additional info in the form of a JSON object in their Data property (e.g. Area arm failure info due to unsealed inputs) to aid with responding to specific activity failure types. The types of JsonProgressData that can be received from each activity type are:

- Area Arm activities:
  - Arm Failure due to unsealed inputs, lists the unsealed input IDs.
  - Arm Failure due to module health, lists the module IDs with problems.
  - Arm Failure due to alarm reporting path problems, lists the CommsPathTypes that have problems.
  - Arm Failure due to unacknowledged system messages (no extra information).
  - Arm Failure due to no permission.
- All control activities:
  - Failure due to no permission.

Update Monitor

Live Review Events can now be tracked through the update monitor, with optional filtering based on the events' category, event type, or which items were involved in the event.

Item Data Changes (creations, updates and deletions) can now be tracked through the update monitor, for example, tracking when changes are made to User details in order to fetch the updated data when they are modified.

Configure Users

The EntityNameAndID JSON objects returned from the **GET /api/v1/config/user** endpoint by default now contain the DateTimeUpdated field in addition to the ID and Name.

The ClientUserDto “Permissions” field in the response from the **GET /api/v1/config/user/{id}** endpoint no longer include the ControlAbilities sub-field, as all related item permissions are now given by default when assigning individual permissions to users. More advanced/specific permission assignments must now be done through Permission Groups if required.

The Get All Users endpoint (**GET /api/v1/config/user**) now accepts an optional "modifiedSince" filter parameter to allow querying for just the users that have been modified since a certain date/time. It has also gained a “count” parameter which specifies the maximum number of items to include in a response, and an “includedProperties” parameter which optionally specifies a list of properties to additionally include in the response data, for example, you might send a request like **GET /api/v1/config/user?includedProperties=Credentials,Permissions** if you wanted to retrieve the credentials and permissions of all users in a single request.

Postman Request Collection

The Inception Postman request collection (downloadable from the Example API Requests section of the REST API documentation) has been updated to include new sample requests for new and updated features.

Protocol Version 1 (Firmware v3.0, Initial Release)

The initial release of the Inception REST API allows 3rd party integration through functionality including area/door/output control, item information retrieval, live item state monitoring, and user management. See the on-board Inception REST API documentation at http://[YourInceptionAddress]/ApiDoc for the full list of API features and how to use them.