May 18, 2020

Automation Anywhere Version A2019

Legal Notices

Examples and graphics are provided only as reference information and might not match your site.

Content

Manage

This collection of topics will guide you in configuring the devices and users that access Automation Anywhere.

Learn more about these topics:

- Activity
  Monitor, pause, stop, and resume ongoing automation activities.
- Devices
  The device is an Automation Anywhere Enterprise client machine through which a user connects to the Enterprise Control Room to create or run bots. Manage devices that are registered to the Enterprise Control Room through the My devices page located under the DEVICES tab.
- Workload management
  Divide your automations into small, logical Work Items from the Workload page. Process the Work Items simultaneously to ensure that your automation goals are achieved with optimum resource utilization.
- Managing packages
  Users with the Manage package permission can upload and manage packages.Automation Anywhere provides you with the flexibility to decide which packages you want to make available to the Bot Creators for creating bots.
- Credentials and lockers
  Sensitive information, such as user credentials, account numbers, and social security numbers that are included in automation tasks, are encrypted and stored as credentials centrally in the Credential Vault.
- Bot Store
  You can access the Bot Store from the Enterprise Control Room. From the Bot Store, you can download bots or packages to your Enterprise Control Room repository.
- Administration
  Enterprise Control Room administrators manage settings related to the database, Credential Vault, , users, roles, action packages, licensing, and more.
- Enterprise Control Room log files
  Various types of information about the Enterprise Control Room are captured in different log files. You can analyze these log files when the Enterprise Control Room or a bot encounters an error and identify the root cause for that error.
- Enterprise Control Room APIs
  The Automation Anywhere Enterprise Control Room provides various public APIs which allow you to customize your business automation for third-party applications.

Related tasks
Schedule a bot
Set up locker and assign credentials
Create a role
Create user
Set device credentials
Edit profile
Installing licenses

Activity

Monitor, pause, stop, and resume ongoing automation activities.

In Progress Activity

Enables user to monitor a range of process activities. Apply search parameters to find a specific bot. Search parameters include:

Status
: Choose the status to search, options include:

  - Active
  - Paused
  - Unknown
  - Paused for input
  - Pending execution
  - Queued

Current action
: Type the action.

Current bot
: Type the bot name.

bot
: Type the bot name.

Activity type
: Select the type, options include:

  - Run bot
  - Import queue files
  - Run bot with queue

Queue
: Type search criteria.

Device
: Type search criteria.

Action buttons

Use mouse to rollover action icons (vertical ellipsis) to identify specific functions. The following actions are enabled:

Refresh
: Refresh the contents to view the updated status.

Pause checked items
: Pause the process.

Resume checked items
: Resume running the process.

Stop checked items
: Stop the process.

Move checked items to history
: Moves the selected activity to history.

Export checked items to CSV
Export the data to a CSV file based on:

- Filters
- Selection

Customize columns
Show or hide specific columns.

- Schedule
  Schedule bots to run on unattended Bot Runners from the Activity > Scheduled tab in the Enterprise Control Room.
- Using event triggers
  All the bots that have a preset or predefined event as a trigger are listed under the Event triggers tab. An Enterprise Control Room user with View event triggers permissions can view Event triggers to monitor bots.
- Historical activity
  You can search, edit, and deploy previously created bots.
- Audit Log
  View logs and details of specific activities.

Schedule

Schedule bots to run on unattended Bot Runners from the Activity > Scheduled tab in the Enterprise Control Room.

Overview

Edit, view, activate, deactivate, delete a schedule, export selected activity details to a CSV file, and search by activity name.

Perform the following actions on a column:

- Click a column to sort it in ascending or descending order.
- Use a drag-and-drop operation to move the column left or right.
- Move the mouse cursor to the end of the column and drag to re-size.

Activity Table

The elements of the Activity table include:

Type
The type of schedule.
Next occurrence
The next time the scheduled bot will run.
Activity name
The name of the activity.
Bot Name
The name of the bot.
Schedule
The description of when the activity will run.

Devices
: The devices on which the bot will run at the scheduled time.

Status
: The Status of the scheduled activity.

Modified by
: The name of the user who last modified the activity.

Last modified
: The date and time when the activity was last modified.

Schedule Actions

Perform the following tasks on an individual Schedule by moving your mouse over the Actions icon.

- Schedule a bot
  You can schedule a bot to run at a specific time.
- Edit a scheduled activity
  Make changes to a schedule so that the automation is not skipped.
- Delete a scheduled activity
  Delete a scheduled activity.

Schedule a bot

You can schedule a bot to run at a specific time.

Prerequisites

Create a bot.

Note: Automation will fail in the following cases:

- If any of the bot dependencies are missing.
- If you do not have folder privileges on the dependencies.
- If you do not have the Run+Schedule permission.

Procedure

1. Navigate to Activity > Scheduled
2. In the Scheduled activity page, click Run bot and then select schedule bot.
3. Enter the Name and optionally add a Description for the schedule.
4. Add schedule details from the Schedule tab.
   Choose an option to schedule the bot:
   - Run once: To run the bot on a given day at a specified time, enter the Start date, Start time, and Time zone.

     The default value of the Start date is the current day. The default Start time is a roundup to the next half-hour.

     Note: The value of the Start date is always later than or equal to the current date. If the start date is the current date, the scheduled time cannot be less than the current time.

- Run repeatedly: To schedule a bot to run at specific time on a given day, set the Start date, End date, Start time, and Time zone values.

  The default value of the Start date is the current day. The default Start time is a roundup to the next half-hour. For example, if the current time is 13:43 hours, the default time 14:00 hours is displayed. The default value of the End date field is blank. The default Time zone is PDT (UTC-7:00) Los Angeles, America.

5. Select the Bots folder within TaskBots.
   The available bots are displayed with the option to select them. Any Input values and the bot dependencies are shown.
6. Select a Bot Runner user from the Available bot runners list in the Device/Run As tab.
7. Click the right arrow.
   The device is added to the Selected devices, which displays the list of connected and disconnected devices to the Enterprise Control Room.
   Note: If you want to enable a device, it must be connected to the Control Room. Also, if a device does not appear in the list, ensure that an active Bot Runner session is running on the device.
8. Click Device pools
   Select the desired pool from the list of device pools. An option is available to override the configured default device.
9. Click Schedule bot.
   The Schedule bot option remains disabled until all the required items, such as bots, schedule details, and devices, are selected.

Related concepts
Bot permissions for a role
Related tasks
Edit a scheduled activity
Delete a scheduled activity

Edit a scheduled activity

Make changes to a schedule so that the automation is not skipped.

Edit the scheduled activity in order to:

- Change the schedule type, date, or time.
- Add or remove Bot Runners from the schedule.
- Change the retry settings.

Procedure

1. Hover over the Actions icon of an item in the Activity table.
2. Click Edit.
   The Edit scheduled bot page appears.
3. Make changes to the schedule details, bots, and devices, as required.
   Note: The system redeploys the bots and dependencies only if there are updates to the bots or its dependent files.
4. Click Schedule bot.

Related tasks
Schedule a bot
Delete a scheduled activity

Delete a scheduled activity

Delete a scheduled activity.

To delete a scheduled activity:

Procedure

1. Hover over the Actions icon of an item in the Activity table.
2. Click Delete.
   A delete confirmation message appears.
3. Click Yes, delete to delete the scheduled activity.

Related tasks
Schedule a bot
Edit a scheduled activity

Using event triggers

All the bots that have a preset or predefined event as a trigger are listed under the Event triggers tab. An Enterprise Control Room user with View event triggers permissions can view Event triggers to monitor bots.

Available user roles associated with event triggers

Role	Description
View event triggers	Enables a view-only access and cannot run or delete any event triggers.
Manage event triggers	Enables a manage access to event triggers. Can delete an existing event trigger or click Run with event triggers to add it.

Supported actions

- View the trigger status, bot path, user name, role, and modification details.
- Use the Search option for the above parameters.
- Copy, cut, or delete the selected event trigger.
- Enable or disable the selected event trigger.

- Add event triggers
  Enterprise Control Room admin users with Manage event triggers role can add event triggers in Enterprise A2019.

Related concepts
Adding a trigger to run a bot

Add event triggers

Enterprise Control Room admin users with Manage event triggers role can add event triggers in Enterprise A2019.

Only administrators can add event triggers in Enterprise A2019.

Procedure

1. Click Bots > My bots.
2. Click Run bot > Run with event triggers.
3. In the Add event triggers page, select a TaskBot and use the right arrow to add it.
4. Click Next.
5. Select the role or user that must be associated with this bot, and use the right arrow to add it.
6. Click Add event trigger.
   The selected bot is added to the Event triggers page under Activity.

Historical activity

You can search, edit, and deploy previously created bots.

Search parameters

Apply search parameters to find a specific bot. These parameters include:

Status
: Choose an activity status:

  - Completed
  - Failed
  - Stopped
  - Timed out
  - Unknown
  - Deploy failed
  - Pending execution

Item name
: Enter the name of the item listed.

Device name
: Enter the device name.

Bot name
: Enter the name of the bot.

User
: Enter the user name.

Actions buttons

You can perform the following actions:

Export checked items to CSV
: Export the data to a CSV file based on:

- Filters
- Selection

Refresh
: Refresh the contents to view the updated status.

Customize columns
: Show or hide specific columns.

- Completed historical activity

  View a list of all completed activities and corresponding information.

Completed historical activity

View a list of all completed activities and corresponding information.

Completed activities

All activities, successfully completed or not, are listed in the historical activity page. From this page, run an activity again and perform other tasks such as export the data in the table in CSV format, customize columns, or refresh the list in the table. Apply search parameters in the search bar.

Note: Specify search parameters for the same column using OR operator. Specify search parameters for different columns, the system searches using AND operator.

Perform the following actions on a column:

- Click a column to sort it in ascending or descending order.
- Use a drag-and-drop operation to move the column left or right.
- Move the mouse cursor to the end of the column and drag to re-size.

Information displayed

The Activity table displays information such as the following:

Status
: The status of the activity, including unknown, completed, failed, stopped, or time out.

Item name
: The name of the item.

Device name
: The name of the Bot Runner machine.

Bot name
: The name of the bot.

User
: The name of the user in whose account that particular activity or automation was running on the device.

Started on
: The date and time when the activity was started.

Ended on
: The date and time when the activity was completed.

Last Modified
The date and time when the activity was changed.
Modified By
The name of the user who recently changed the activity.

Actions

Click any individual item to perform the following actions:

View
View details of the completed activity.
Run bot
Run the selected bot. Click Run to open the Run bot now page with all the values of the bot populated.

Tip: Move your mouse over and click the Run icon to run the activity again.

Actions must be done only at a table level, and not on individual items.

Audit Log

View logs and details of specific activities.

Audit Log displays a read-only table of records of actions done by users. These log records are searchable and exportable. Audit logs include both Successful and Unsuccessful entries.

Actions

The following Audit log actions are enabled:

Note: Use the mouse to roll over the action button icons to identify specific functions.

Time filter
View activities for a specific time period. The default time filter setting is Last 24 hours. Users can select from preset time filters or configure a custom time filter.
Search
Search the records. Select additional search filter criteria from the drop-down menu.
Tip: To search the exact phrase, enclose the search phrase within double quotes (for example, "Juan-Finance-564").
Export checked items to CSV
Export the data to a CSV file based on:

- Filters
- Selection

Refresh
Refresh the contents to view the updated status.
Customize columns
Show or hide specific columns.
View
To view details of a table entry, mouse over the entry to expand and click Audit details.

Audit log table

View the following audit details in the table. Click a column to sort it in ascending and descending order. Sort up to three columns by pressing Shift when selecting additional columns. Use a drag-and-drop operation to move the column left or right.

Table item	Description
Status	Shows action status.
Time	Shows the date and time of the action performed.
Action Type	Shows the type of action performed. Some of the action types captured in Audit logs are: • Connect Credential Vault • Create / Edit / Delete Role / User • User / Client Login / Logout • Allocate License • Create / Activate / Deactivate Automation • Run / Schedule bot Stopped / Resumed / Paused / Ended • Unlock the bot
Item Name	Shows the entity on which action was performed.
Action Taken By	Shows the user that performed the action.
Device	Shows the device or machine name or IP address that was used to perform the action.
Source	Shows the component: Enterprise Control Room, Enterprise client or API, from where the action originated or was performed.
Request ID	Shows the unique identity number assigned to a specific set of user actions.

Audit log entries

Each audit log has four (4) entries for each bot that has completed execution. The entry shows the status of each stage of the bot life cycle.

Audit log entry	Explanation	Troubleshooting
Create automation	Bot was sent to the control room and was compiled successfully.	If you do not see this entry or see a failure against it. Check if the Enterprise Control Room is up and running.
Bot Sent To Device	Enterprise Control Room deployed the bot on the specified device successfully.	If you do not see this entry or see a failure against it: • Check if you have configured the Node manager on the corresponding device correctly. • Check for the app Automation Anywhere Bot Manager in Add Remove Programs or in Control Panel > Uninstall a program. Try to reinstall by uninstalling and downloading the app again.

Audit log entry	Explanation	Troubleshooting
		• Try to reinstall by uninstalling and downloading the app again from Device Manager. • Ensure the device auto log in details are set correctly.
Run bot Deployed	Bot has started execution on the specified device.	If you do not see this entry or see a failure against it: • Check if you have configured the Node manager on the corresponding device correctly. • Check for app Automation Anywhere Bot Manager in Add Remove Programs or in Control Panel > Uninstall a program. • Try to reinstall by uninstalling and downloading the app again from Device Manager. • Ensure device auto log in details are set correctly.
Run bot finished	Bot execution completed successfully.	If you do not see this entry or see a failure against it: • Check for the bot execution progression in Activity In progress. • Check the code at the line where activity log has been paused for errors. • If it has paused at a message box, minimize all windows and check if the message box is in the background.
Bot Runner Session Continued	Enterprise Control Room deploys the TaskBots in a sequence for the same RDP session.	
Bot Runner Session Released	When a task successfully completes the execution.	
Download file	Downloading to the Enterprise Control Room.	
Upload file	Uploading a file to the Enterprise Control Room.	
Bot Runner Session	Enterprise Control Room gets the RDP session of Bot Runner machine.	

Devices

The device is an Automation Anywhere Enterprise client machine through which a user connects to the Enterprise Control Room to create or run bots. Manage devices that are registered to the Enterprise Control Room through the My devices page located under the DEVICES tab.

Devices

As an Enterprise Control Room user with Bot Runner, Bot Creator, and Device pools privileges, use the DEVICES tab to do the following:

- View a list of devices registered and connected to the current instance of the Enterprise Control Room.
- Create and view a list of device pools available from the current instance of the Enterprise Control Room.
- Run bots immediately on selected Bot Runners.
- Schedule bots to run on selected Bot Runners.
- Run bots on selected device pools.

Note: Only an admin user has access to see all the devices in the Enterprise Control Room. A non-admin user does not have access to view Bot Creators.

My devices

The My devices page displays all configured devices and the current state for each device listed. A device can be in one of the following states:

Connected
    Device is logged in to the Enterprise Control Room.
Disconnected
    Device is not logged in to the Enterprise Control Room.
Offline
    Device has been unregistered or disabled by the Enterprise Control Room administrator.

The following actions are enabled:

Add local host as a device
    The Bot agent is a lightweight application that enables you to run bots on your device by connecting a local machine to the Enterprise Control Room. To run bots on a local machine, install the Bot agent and add the local device to the list of enabled host devices.
Run bot now with checked items
    Runs the bot on selected device.
Export checked items to CSV
    Export the data to a CSV file based on:

  - Filters
  - Selection

Refresh
    Refresh the contents to view the updated status.
Customize columns
    Show or hide specific columns.

My device pools

Device pools provide a logical grouping of similar Bot Runners to run bots with the work items from a queue. For example, group devices of a particular department or unit and create a device pool for it.

- Register device and install Bot agent
  The Bot agent is a lightweight application that enables you to run bots on your device by connecting a local machine to the Enterprise Control Room. To run bots on a local machine, install the Bot agent and add the local device to the list of enabled host devices.
- Manage my device pools
  As a Device Pool administrator you can create, edit, and view all devices pools that can be used for scheduling automations and workload management. The device pool gives the ability to restrict bot deployments to a specific set of devices and to take advantage of any available device in the pool.

Register device and install Bot agent

The Bot agent is a lightweight application that enables you to run bots on your device by connecting a local machine to the Enterprise Control Room. To run bots on a local machine, install the Bot agent and add the local device to the list of enabled host devices.

The Bot agent version available for download is the latest and compatible with the Enterprise Control Room version that is used.

Note: Use the mouse to roll over action icons to identify specific functions.

Procedure

1. Log in to the Enterprise Control Room through your Automation Anywhere Enterprise URL.
2. Navigate to MY DEVICES.
3. From the action icons, click Add local bot agent.
4. Click Connect to my computer.
5. Follow the steps outlined in the wizard.
   Authenticated proxy access:

   If your device's access to the internet is controlled through an authenticating proxy server, you are prompted to provide the proxy server authentication details. These credentials are required for the device to communicate with the Enterprise Control Room.

   To enable the authenticated proxy, register the device through a Chrome browser with the Automation Anywhere Chrome extension enabled.

6. Refresh the My Devices page and verify that the local device is added.

   Watch the following video on how to install the Bot agent in Enterprise A2019:

   Install the Bot agent

Next steps

Set device credentials. Optionally, Edit profile.

To learn more, see Training - Bot Runners and Control Room communicate without human intervention. This course introduces you to learn how to register devices in the Enterprise Control Room

To access this course, you must log in with a registered Automation Anywhere University or A-People account. .

- Manually switch the Bot agent
  Switch the Bot agent on a registered device to work with a different Enterprise Control Room.
- Set device credentials
  To enable a device for running bots, set the local device credentials.
- Connect Bot agent to an authenticating proxy
  If your bot cannot connect to the Enterprise Control Room due authentication proxy credentials, complete the steps in this task to add the authentication details.
- Set device credentials
  To enable a device for running bots, set the local device credentials.
- Edit profile
  Manage user profiles.

Related tasks
Create device pools
Related reference
Manage my device pools

Set device credentials

To enable a device for running bots, set the local device credentials.

Prerequisites

The Bot agent is a lightweight application that enables you to run bots on your device by connecting a local machine to the Enterprise Control Room. To run bots on a local machine, install the Bot agent and add the local device to the list of enabled host devices. Add the local device before editing the credentials. See Register device and install Bot agent.

For Automation Anywhere Enterprise Community Edition users, your profile contains only one set of credentials at a time. These credentials are applied to any device you select to run your bots. Ensure each device that you use accepts the credentials in your profile.

Automation Anywhere Enterprise Cloud users have the option to apply different credentials to registered devices.

Procedure

1. Click the Device icon and select Update credentials.
2. In the Device login credentials section, enter the Username and Password for the device.
   Device login credentials are required to run a bot from this device.
   Note: Enterprise A2019 does not validate the device login credentials until you run a bot.

   If your username is part of a domain, include the domain within the format `domain\username`. Typically, home users are not part of a domain, unless they are specifically configured.

3. Click Update.

Next steps

Create your first bot

Edit profile

Manage user profiles.

For users of Enterprise Control Room configured with a non-directory environment, change the password, first name, last name, and email address.

Procedure

1. Click the Device icon and select Update credentials.
2. In the Device login credentials section, enter the Username and Password for the device.
   Device login credentials are required to run a bot from this device.
   Note: Enterprise A2019 does not validate the device login credentials until you run a bot.

   If your username is part of a domain, include the domain within the format `domain\username`. Typically, home users are not part of a domain, unless they are specifically configured.

3. Click Update

Manage my device pools

As a Device Pool administrator you can create, edit, and view all devices pools that can be used for scheduling automations and workload management. The device pool gives the ability to restrict bot deployments to a specific set of devices and to take advantage of any available device in the pool.

As a Device Pool Owner or Consumer, you can view only those device pools of which you are the owner or consumer. Only users with the AAE_Queue Admin role can do device management tasks.

Note: You need to create device pools to view those in the list. To get started, click Create device pool.

For ease of access, you can search by device pool name.

- Click a column to sort it in ascending and descending order. You can sort up to three columns by holding the Shift key when you click two more columns. This gives you the option of sorting two additional columns. This way the sorting is done on the entire table and not just the data that is currently visible to you. The last sorting is stored in memory applied by a user per session.
- Use a drag-and-drop operation to move the column left or right.
- Move your mouse cursor at the end of the column and drag to resize.

- Create device pools
  Create a device pool with a unique name and add Unattended Bot Runners to the device pool.
- View device pool details
  As a Enterprise Control Room user with device pool management privileges or as a device pool owner, you can view device pool details to ensure the information provided is correct and if required customize as per your automation requirement.

- Edit device pools
  As an Enterprise Control Room user with device pool management privileges or as a device pool owner, you can edit device pool details to customize your automation requirements.
- Delete device pools
  You can delete a device pool comprising of unattended Bot Runners after your entity's automation goals are achieved and the device pools are no longer required.

Create device pools

Create a device pool with a unique name and add Unattended Bot Runners to the device pool.

Prerequisites

- The Create device pools feature permission or the AAE_Pool Admin role must be assigned to you.
- You can add only those Unattended Bot Runners that are not part of any other pool and are not associated with any role.
- If the device associated with the Unattended Bot Runner is added to the device pool, you can only use the Run bot with queue option to run bots on that device. You cannot create a device pool comprising of Attended Bot Runners.
- You can add Enterprise Control Room user roles as consumers. Only users with these roles can use the pool for any automation.

By default, the creator of the pool is the device pool owner.

Procedure

To create a device pool, do the following:

1. Go to Devices > My device pools page.
2. Click Create device pool on the top right of the My device pool page.
   Tip: If no device pools are available, click the create a device pool link in the My device pool page.
   The Create device pool page appears.
3. Enter a valid device pool name.
   For example, you can create a Finance Automation pool that can run all finance-related automations on Unattended Bot Runners from the finance department.
4. Select Unattended Bot Runners from the list.
   This list shows only the devices with Unattended Bot Runner licenses.
   Restriction: Unattended Bot Runners that are a part of other device pools are disabled for selection.
5. Add the Unattended Bot Runner(s) to the Selected devices list.
   Tip: Click the left arrow button to remove the Bot Runner from the Selected devices list.
6. Subsequently, grant permissions to view, edit, and delete the device pool to the other Enterprise Control Room users:
   a) Click Next to select the Device Pool Owners.
   b) Select user(s) from the Available users list.
   Tip: Search the list of users based on their Username, First name, or Last name.
   c) Click the right arrow button
   The user appears in the Selected users list.
   Note: The device pool creator is listed as the default owner of the pool.
   d) Click the left arrow button to remove the user from the Selected users list.
   Restriction: You cannot remove the device pool creator.

7. Click Next to select the Device Pool Consumers.
   Do this step so that the device pool consumers can view the device pool when they run the automation for the bot with a queue by following the next set of steps.
   a) Select a Role from the Available roles list.
   Tip: Search for a role name.
   b) Click the right arrow button.
   The user appears in the Selected roles list.
   Tip: Click the left arrow button to remove the user from the Selected roles list.
8. Click Create Device Pool.
   The device pools for which you have consumer privileges are listed in the My Device Pools page.

Next steps

View device pool details
Related concepts
Run bot with queue

View device pool details

As a Enterprise Control Room user with device pool management privileges or as a device pool owner, you can view device pool details to ensure the information provided is correct and if required customize as per your automation requirement.

Use the Device pool details page to view automations that are scheduled to run with or without workload.

Procedure

To view device pool details:

1. Go to Devices > My Device Pools
2. Locate the device pool to view, mouse over the View action icon and click.
   The Device pool details page appears. The Scheduled Automations tab is selected by default. It lists the automations that are created using Run bot now or Schedule bot (upcoming schedules) on that device pool.

   The second tab, Run with Queue Automations lists the automations that are scheduled to run for Workload using the option Run bot with queue.

   The page also has the device pool details such as the Name, Description, Status, and General details. It allows you to view additional details of the device pool such as Automations, Devices, Owners, and Consumers.

3. Select each tab to view its details.

   Automations

   Shows the automations that are using the device pool and the order that is chosen to run those. This is shown as the default view. To find an automation quickly, use the search option using Status, Automation name, Queue, or Activity type.

   You can perform the following actions on a table column:

   - Click a column to sort it in ascending and descending order. You can sort up to three columns by holding the Shift key when you click two more columns. This gives you the option of sorting two additional columns. This way the sorting is done on the entire table and not just the data that is currently visible to you. The last sorting is stored in memory applied by a user per session.

- Use a drag-and-drop operation to move the column left or right.
- Move your mouse cursor at the end of the column and drag to resize.

Devices

Shows list of unattended Bot Runner devices that are part of the device pool.

Owners

Shows list of device pool owners that are granted permission to view, edit, and delete the device pool.

Create device pools

Consumers

Shows the list of device pool consumers who are granted permission to view the device pool as an option to running automations.

Create device pools

General Details

Shows the last modified date and time, name of the user who modified device pool details, and the Object Type which is the component on which modification was done.

Next steps

Edit device pools, or Run bot with queue.
Related tasks
Create device pools
Delete device pools

Edit device pools

As an Enterprise Control Room user with device pool management privileges or as a device pool owner, you can edit device pool details to customize your automation requirements.

When you open the device pool in edit mode, you have to first define the priority or the order in which the automations will run in the Automations tab. The priority options are visible only when you edit a device pool and are not available when you create a device pool. You can also update the Bot Runner, Owner, and Consumer details.

Procedure

1. Go to Devices > My Device Pools.
2. Locate the device pool you want to edit, mouse over the actions menu (vertical ellipsis), and click Edit. You can also edit device pool details when in view mode.

   View device pool

   The Device pool details page appears with the Scheduled Automations tab selected by default.

   The first tab, Scheduled Automations lists the automations that are created using Run bot now or Schedule bot (upcoming schedules) on that device pool.

   Run bot now

Schedule a bot

The second tab, Run with Queue Automations lists the automations that are scheduled to run for Workload using the option Run bot with queue.

3. Set the Queue Execution mode to edit workload automations.

   Select the Run with queue automations tab, then select either Round robin or Priority as shown in table to define the order in which your automations run in the queues:

   - Round robin: Run your automations at equal time intervals in the Time slice field.

     Set a Time slice in seconds, minutes, or hours. Calculate or estimate the time for each automation and then provide this number.

     - The default time slice is 5 minutes.
     - The time slice cannot be set to zero.

     Automations are executed for only 5 minutes first, then the system checks for other automations in queue for execution. If there are other automations in the queue, that automation is paused and the next automation is executed. This method continues until all automations in the queue are executed.
   - Priority as shown in table: Run your automations based on a priority defined in the priority table. This method enables you to run automations in the order of priority.

     Set the individual priorities for each of the queues. Priority 1 is the highest priority and that queue is processed first and completely by the device pool. Then the device pool moves onto the processing queue with Priority 2. When the queue with Priority 2 is processed completely, the device pool proceeds to processing queue with Priority 3, and so on.

     Automations are processed until all automations are consumed from the specified automation queue.

     This option appears only on the Run with queue automations tab. It is not available when you use Schedule Automations.

     The following details are listed the priority table:

Table item	Description
Priority	Shows the priority number allotted to that queue. • You can edit the Priority column. You can set or reset the priority of implementing the automations. Ensure that you provide a unique priority value to two different work items because same values are not be allowed. • You can also view the priority list in ascending or descending order by clicking the ordering arrows in the Priority header.
Status	Shows the automation status: Active or Inactive.
Automation Name	Shows the automation that is selected to run on the device pool.
Started On	Shows the run date and time of the automation.
bot	Shows the bot name that will run using this device pool.
Queue	Shows the queue name that will be used to run the automation using this device pool.

Table item	Description
Activity Type	Shows the activity type used to run the automation using this device pool - Run bot with queue.

You can perform the following actions on a table column:

- Click a column to sort it in ascending and descending order. You can sort up to three columns by holding the Shift key when you click two more columns. This gives you the option of sorting two additional columns. This way the sorting is done on the entire table and not just the data that is currently visible to you. The last sorting is stored in memory applied by a user per session.
- Use a drag-and-drop operation to move the column left or right.
- Move your mouse cursor at the end of the column and drag to resize.

4. Update the list of unattended Bot Runner devices that will be included in the device pool.
5. Update the list of device pool owners who are granted permission to view, edit, and delete the device pool.
6. Update the list of device pool consumers who are granted permission to view the device pool.
7. Click Save changes.

Related tasks
Create device pools
Delete device pools
View device pool details

Delete device pools

You can delete a device pool comprising of unattended Bot Runners after your entity's automation goals are achieved and the device pools are no longer required.

You can choose to delete your device pools in either of two ways:

- Delete one device pool
- Delete multiple or all device pools

Procedure

1. Delete one device pool:
   a) Go to Devices > My Device Pools.
   b) Locate the device pool to delete, mouse over the Delete action icon and click.
   c) Confirm or cancel as required.
2. Delete multiple or all device pools
   a) Select the device pools to delete or select all device pools by selecting the Select All check box in the header.
   b) Click Delete given at the top of the device pools table.
   c) Confirm or cancel as required.
   If the device pool is being used for workload automation, you will not be allowed to delete it.
   Based on your selection, the devices are deleted.

Related tasks
View device pool details
Create device pools

Workload management

Divide your automations into small, logical Work Items from the Workload page. Process the Work Items simultaneously to ensure that your automation goals are achieved with optimum resource utilization.

Prerequisites

To manage your workload automation, ensure that you are allocated a combination of any or all of the following roles and permissions:

Feature type	Privileges
User roles	• AAE_Admin • AAE_Queue_Admin • AAE_Pool_Admin
Activity permissions	• View my in progress activity • View my scheduled bots • Schedule my bots to run
Device permissions	• View and manage my Bot Creator, Bot Runner, and device pools • Create device pools • Administer all device pools
Bots permissions	• View my bots • Run my bots
Workload permissions	• View and manage my queues • Create queues • Administer all queues

- Create workload queues
  A queue is one of the main building blocks of Workload Management (WLM). A queue holds data known as Work Items for further processing. The system distributes these Work Items to individual Unattended Bot Runners in a device pool for processing.
- Run bot with queue
  Collectively process all work items of a queue across all the Bot Runners present in one or more device pools.
- Manage workload queues
  For workload maintenance tasks such as view the details of queues to pause, stop, or resume its automation, edit the queues, manage the work items in the queue, and delete the queues.
- Manage Work Items
  Manage Work Items of a queue to fix the discrepancies before queue processing and reduce your automation-related errors.

Create workload queues

A queue is one of the main building blocks of Workload Management (WLM). A queue holds data known as Work Items for further processing. The system distributes these Work Items to individual Unattended Bot Runners in a device pool for processing.

For workload automation create device pools, add Bot Runners to the pool, create queues, add queue owners/participants/consumers, define the work item structure, insert work items, and finally run the automation with the queue.

Create and attach work item template to a Bot
: Attach a work item template to a TaskBot to use the TaskBot in workload automation.

Create device pools and add bot runners to the pool
: Create a device pool with a unique name and add Unattended Bot Runners to the device pool.

Create queues
: Create queues that hold specific sets of data your bot is expecting for automation. To create queues, an Enterprise Control Room administrator assigns the AAE_Queue Admin role with View and manage my queues, Create queues, Administer all queues, and View my in progress activity permissions.

Add queue owners
: Add queue owners who can create, edit, and view queues. The queue creator is the default queue owner and is able to add other Enterprise Control Room users as queue owners, if required.

Add participants to queue
: Add queue participants from different roles defined in the Enterprise Control Room. This is an optional step.

Add consumers of queues
: Add queue consumers from different roles defined in the Enterprise Control Room. This is an optional step.

Define Work Item structure
: Define the Work Item structure for processing in a queue. This enables you to manually upload the Work Items from the system in the absence of ready data in a file.

Add Work Items
: Add Work Items from an Excel or CSV file to the queue after you define the structure.

Related concepts
Run bot with queue
Related tasks
Attach work item template to TaskBot

Create queues

Create queues that hold specific sets of data your bot is expecting for automation. To create queues, an Enterprise Control Room administrator assigns the AAE_Queue Admin role with View and manage my queues, Create queues, Administer all queues, and View my in progress activity permissions.

Prerequisites

Create a queue by providing details such as the queue name, queue owners, participants, consumers, and work item structure. A summary of these details is available in the tab on the left side of the Create queue page. Open any tab to edit the details.

Procedure

1. Go to Workload > Queues.
2. Click Create queue.
   The Create queue page appears.
3. Configure the following General Settings:
   a) Queue Name: Enter a name for the queue that reflects its purpose.
   For example, Payroll Queue for work items that are designed to manage a payroll system.
   b) Optional: Description: Enter a description that reflects what the queue will achieve.
   For example, the Payroll Queue will process automation that are designed to manage the payroll system.
   c) Reactivation Threshold: Set the threshold to resume queue processing.

   A queue is processed until all work items are completed. When new work items are added to the queue, the Reactivation Threshold value specifies the minimum number of new work items required to resume queue processing.

   Work items are those items with a Ready to Run status.

   Default threshold is 1 (one).

   d) Optional: Time required for a person to complete one work item: Select the average time that a person would need to complete one work item in seconds, minutes, hours, or days.
4. Click Next to add the queue owners.
   Add queue owners
   Note: You can choose to Create draft of queue and add the remaining information later.

Related tasks
Edit queues
Delete Queues

Add queue owners

Add queue owners who can create, edit, and view queues. The queue creator is the default queue owner and is able to add other Enterprise Control Room users as queue owners, if required.

Prerequisites

Queue owners are allowed to edit the queue and add new work items to the queue.

Procedure

1. Select user(s) from the Available Users list in the Owners tab.
2. Click the left arrow key.
   The users are added as Queue Owners in the Selected users list.
3. Click Next to add the queue participants.
   Add queue participants

Add participants to queue

Add queue participants from different roles defined in the Enterprise Control Room. This is an optional step.

Prerequisites

Participant roles can add new work items and view the queue. However, they are not allowed to edit other queue properties.

Procedure

1. Select role(s) from the Available Roles list in the Participants tab.
2. Click the right arrow button.
   The roles are added as Participants in the Selected roles list.
3. Click Next to add the queue consumers.
   Add consumers of queues

Add consumers of queues

Add queue consumers from different roles defined in the Enterprise Control Room. This is an optional step.

Prerequisites

Queue consumers can view the queue and all the work items in the queue. In addition, they can use this queue for running bots on Unattended Bot Runners.

Procedure

1. Select role(s) from the Available Roles list in the Consumers tab.
2. Click the right arrow button.
   The roles are added as Consumers in the Selected roles list.
3. Click Next to define the Work Item structure.
   Define Work Item structure

Define Work Item structure

Define the Work Item structure for processing in a queue. This enables you to manually upload the Work Items from the system in the absence of ready data in a file.

Define a Work Item structure using any one of the following methods:

1. Using an Excel/CSV file.
2. Using an existing work item template.
3. Manually
   Remember: The work flow to process Work Items differs for a queue based on the method that you choose in the Define Work Item Structure tab.

Procedure

1. Select a method to add header columns for Work Item processing:
   - Use an Excel/CSV file: Add the header columns from an existing Excel or CSV file. You can point to the Excel spreadsheet or CSV file you are using in one or more task bots you will run in this queue.

a) Type a unique name for the Work Item structure in the Work item template field.

For example, if the queue contains employee information, you can specify the Work item template as Employee Data.

b) Select a column for inclusion in the Work Item structure from the list of column names. The columns are defined based on the header rows of the selected Excel or CSV file. A maximum of ten (10) columns are allowed for selection and viewing in the Enterprise Control Room.

For example, you can select column headers Employee Name, Employee ID, and Designation. You can then select the Data Type - Text, Number, or Date for that column. You can also choose to view these columns being processed in the Activity page.

Note:

- c) The system allows you to filter/sort Work Items on the columns for viewing the Work Item data in the Enterprise Control Room.
- d) When you upload work items from an xls or xlsx file with data type as text, the Excel file column populated with a date in any format (for example, 8/6/2019) is converted to its corresponding WLM date format (for example, Sat Jun 08 00:00:00) in the Enterprise Control Room Work Item. However, the same is not applicable to a csv file.

e) Select up to three columns for sorting in an ascending or descending order. When the system processes the Work Items from the queue, it uses the sort criteria specified to retrieve the Work Items in that order.

For example, to process payslips with first Employee Id followed by Employee Name from 1 to n and A to Z, specify Employee Id and Employee Name in an ascending order.

- Use work item template: Add header columns by searching for an Existing work item template or from the Available work item templates.

  This allows you to pass the values or attributes from the template to a TaskBot with the help of Work Item variables when you use the option Run bot with queue.
  Tip: Search for an existing Work Item template when there are a large number of templates available for selection.

  Use Work Item variables

- Manually: Define the Work Item structure manually. You do not have to select from an existing structure.

  a) Type a name for the Work Item structure in the Work item template field.

  For example, if the queue contains employee information, add the Work item template as 'Employee Data'

  b) Add column header names for the Work Item and select the data type for each column - Text, Number, or Date
  c) Select the display and sorting for the columns in the .

2. Click Next to add the Work Items.
   Add Work Items

Add Work Items

Add Work Items from an Excel or CSV file to the queue after you define the structure.

Prerequisites

Tip: You can also add Work Items later by editing the queue.

Edit queues

Procedure

1. Click Browse to select an Excel or CSV file.
   The file is added as a Work Item in the queue.
   Note: When you upload work items from an xls or xlsx file with data type as text, the Excel file column populated with a date in any format (for example, 8/6/2019) is converted to its corresponding WLM date format (for example, Sat Jun 08 00:00:00) in the Enterprise Control Room Work Item. However, the same is not applicable to a csv file.
2. Click Create Queue.
   The queue is successfully added at the top of the Queues list. You can choose to apply the column sorting to view as required.

Next steps

- Now that you have created a queue, it is now ready for deployment from a bot.

  Run bot with queue

- Manage Work Items of a queue to fix the discrepancies before queue processing and reduce your automation-related errors.

  Manage Work Items

Run bot with queue

Collectively process all work items of a queue across all the Bot Runners present in one or more device pools.

To run a bot with queue, ensure you are allocated a combination of any or all of the following roles and permissions:

Feature type	Privileges
User roles	• AAE_Admin • AAE_Queue_Admin • AAE_Pool_Admin
Activity permissions	• View my in progress activity • View my scheduled bots • Schedule my bots to run
Device permissions	• View and manage my Bot Creator, Bot Runner, and device pools • Create device pools

Feature type	Privileges
	• Administer all device pools
Bots permissions	• View my bots • Run my bots
Workload permissions	• View and manage my queues • Create queues • Administer all queues

Use Run a bot with queue from any of the following pages:

1. Activity > Scheduled
2. Bots > My Bots > Public page
3. Workload > Queues page

The procedure for running a bot with a queue is the same in all these pages. To Run bot with queue, note that:

- You can run bots only on Unattended Bot Runners. You cannot run bots on Attended Bot Runners from the Enterprise Control Room.
- The bots have to be Checked in to the Public folder in order for the bots to qualify to be run with queue.

To process the work items using the Bot Runners, choose the TaskBot to run, select a queue, add a Bot Runner, and assign it to a device pool.

- Add TaskBots and dependent files
  Add a TaskBot, review the input values, and dependent files for the automation in the Taskbots tab from the Run bot with queue page.
- Add queue, Bot Runner, and device pool
  Add a queue, Bot Runner, and device pool to the automation from the Run bot with queue page.

Related tasks
View automation of a queue

Add TaskBots and dependent files

Add a TaskBot, review the input values, and dependent files for the automation in the Taskbots tab from the Run bot with queue page.

Procedure

1. Go to Activity > Scheduled, Bots > My Bots, or Workload > Queues page.
2. Click Run bot with queue
   You are taken to the Activity > Run bot with queues Create page.
3. Enter a Name for the automation.
4. Enter Description.
   Tip: This could describe the purpose of running the TaskBot with a queue.

5. Go to the folder that contains the required TaskBot.
6. Select a TaskBot to process in the queue from the list.
   By default, the Bots folder is selected.
   Tip: Use Search to find a TaskBot quickly.
   The Input values and Dependencies options appear at the bottom of the page.
7. Optional: Select the Input values check box to add the values of variables to the bot during run time.
   This is enabled only if the selected TaskBot has Input values.
8. Optional: Review the list of dependent files, if available.
   This is enabled only if the selected TaskBot has Dependencies.
9. Click Next to add a queue, Bot Runner and device pool.
   Add queue, Bot Runner, and device pool
   If the selected TaskBot does not contain a work item template, an error message appears at the top of the page.

   Define Work Item structure

   The TaskBot also has an icon that indicates a missing work item template.

Related reference
Bot dependencies

Add queue, Bot Runner, and device pool

Add a queue, Bot Runner, and device pool to the automation from the Run bot with queue page.

Prerequisites

You can select only those queues that are not in use and for which you have consumer access privileges. The In use queues show as disabled in the Available queues list and you cannot use multiple queues to add Bot Runners.

Tip: Use Search to quickly find the required queue, Bot Runner, and device pool.

Procedure

1. Select a Queue from the Queues list.
2. Click Next.
3. In the Run as tab, select a Bot Runner from the Available bot runners list.
4. Click Next.
5. In the Device pool tab, select a Device Pool from the Available device pools list.
6. Click Add.
   The queue and device pool are added to the run bot with the queue list.
7. Optional: Click Remove to replace the queue or the device.
8. Click Run bot with queue.
   The queue status changes to `In use` on the Queues page.

Next steps

Manage workload queues

Manage workload queues

For workload maintenance tasks such as view the details of queues to pause, stop, or resume its automation, edit the queues, manage the work items in the queue, and delete the queues.

Workload maintenance tasks

For workload automation maintenance, do the following (in any order):

View queue details
: Use the View queues details page to view the details of a particular queue.

Edit queues
: Edit a queue using two methods - from the Queues list, or from the View queue page.

Delete Queues
: Delete selected or all queues.

Manage Work Items
: Manage Work Items of a queue to fix the discrepancies before queue processing and reduce your automation-related errors.

Related reference
Actions allowed on view queue page

View queue details

Use the View queues details page to view the details of a particular queue.

Prerequisites

Permissions required:

1. Queue Owner or Queue Participant or Consumer rights
2. View and manage my Queues feature permission

Procedure

1. Go to Workload > Queues
2. Hover over a queue to view
3. Click the Actions menu (vertical ellipsis).
4. Click the view details button.

   This launches the View queues page which shows details of the queue in two sections:

   a) Name, Description, My access status, and queue Status such as:
   - b) New when the work item is added to the queue recently.
   - c) On hold when the work item is deferred from processing by a Queue owner, participant, or consumer.
   - d) Failed when the work item processing failed on an unattended bot.
   - e) Completed when the work item is successfully processed by a Bot Runner or marked Completed by a queue owner, participant or consumer.
   - f) Data error when there is an error in loading data from the file.
   - g) Active when the work item is currently being processed or staged for processing.

- h) Ready to run when the work item is successfully processed for execution does not have any data errors and can be staged for processing.

i) Queue contents in different tabs such as:

a) Work Items: This is the default view. This allows you to view all work items in a tabulated form. You can use the filter go view specific work items. For example, all work items with status as Completed. You can View, Edit, or Delete the individual work items in each row. You can also change the status of the work items in bulk. For example, change the status of all the work items in Data error to On hold.

b) General: View the Reactivation Threshold and Time required to complete one work item.

c) Owners: View the user names of queue owners who can edit the queue and add new work items.

d) Participants: View the user names of queue participants who can add new work items and view the queue.

e) Consumers: View the user names of consumers who can view the queue and all the work items in the queue. In addition, they can use this queue when running bots.

f) Work Item Structure: View the work item structure that you defined when creating the queue.

Tip: Edit any of these details by either clicking the edit this queue link or the Edit button. Also, delete the queue by clicking the Delete button.

j) dfdf

Next steps

View automation of a queue

Related concepts

Manage Work Items

Related tasks

Create queues

Edit queues

Delete Queues

Related reference

Actions allowed on view queue page

Edit queues

Edit a queue using two methods - from the Queues list, or from the View queue page.

Prerequisites

Permissions required:

1. AAE_Queue Admin role
2. Queue Owner rights to edit queues that you created
3. Queue Participant rights to edit queues that are created by other queue owners

Procedure

1. Go to Workload > Queues
2. Edit a queue from the View queue page or from the Queues list
3. Hover over a queue to edit
4. Click the Actions menu (vertical ellipsis).
5. Click the View button

   The View queue page is launched.

6. Click either of the following to launch the Edit Queue page
    - edit this queue link
    - Edit button
7. Edit the queue details such as the queue name (applicable only if in draft), description, work items, threshold and time values, owners, participants, and consumers.
   Note: The Work Item structure cannot be edited after it is defined.
8. Upload a file for the work item that will be used for processing in this queue
   The Work Items tab is shown by default.
   Tip: You can search for a work item quickly based either on Status or Status details using the search option.
9. Click Browse
10. Select the file to upload
    Note: You can upload only an Excel or CSV file.
11. Click Save changes
    If you provide a duplicate name, an error is displayed.
12. Edit the name and save the changes made to the queue
    An edit successful message appears.

Next steps

Delete Queues
Related concepts
Manage Work Items
Related tasks
Create queues
Related reference
Actions allowed on view queue page

Delete Queues

Delete selected or all queues.

Prerequisites

Permissions required:

1. AAE_Queue Admin role
2. Queue Owner rights

Procedure

1. Go to Workload → Queues
2. Delete selected or all queues
    - Delete selected queue
        a) Hover over a queue to delete.
        b) Click the Actions menu (vertical ellipsis).
        c) Click the Delete button.
        d) A confirmation message to permanently delete the selected queue appears.
        e) Click Yes, delete to confirm or No, cancel to discard the action.
        f) A confirmation message appears after you delete the queue
    - Delete multiple selected or all queues

a) Select the check box of required queues or select the check-box given in the header to select all queues
b) Click the Delete button above the table header.
c) A confirmation message to permanently delete multiple queues appears.
d) Click Yes, delete to confirm or No, cancel to discard the action.
e) A confirmation message appears.
Note: A queue will not be deleted if it is being used for processing a work item. An error message appears for that particular queue.

Related tasks
View queue details
View automation of a queue
Edit queues

Actions allowed on view queue page

Use different actions such as sorting, searching, or filtering on the table view of the queues.

Searching and filtering

For ease of access, apply search parameters to Status and Queue Name columns.

- Specify the search parameters in the search bar for Queue Name. When you specify search parameters for the same column, the system searches using OR operator. When you specify search parameters for different columns, the system searches using AND operator.
- Choose the search parameters from a list in the search bar for Work Item Status.

Table items

The following describes the list of items that can be viewed in the table:

Table Item	Description
ID	Shows the system generated id for a work item. When a work item is added to a queue, system generates an id for that work item.
Status	Shows Work item status: Use the View queues details page to view the details of a particular queue.
Start Time and End Time	Shows the Work Items processing start/end time and date.
Modified by	Shows the name of the user who had modified the Work Item last.
Last Modified	Shows the time and date when the Work Item was modified last.

Note: Apart from the above system generated columns, the fields that you define in your work item are also displayed as columns.

Actions on table column

Use the following actions on a table column:

- Click a column to sort it in ascending and descending order. You can sort up to three columns by holding the Shift key when you click two more columns. This gives you the option of sorting two additional columns. This way the sorting is done on the entire table and not just the data that is currently visible to you. The last sorting is stored in memory applied by a user per session.
- Drag a column to the left or right
- Move your mouse cursor at the end of the column and drag to re-size

Actions on Work Items

Use the following tasks on specific Work Items:

Table Item	Description
Refresh	Allows you to refresh the table contents so that you can view the latest Work Item status
Delete	Allows you to delete one or multiple Work Items.
Mark complete	Allows you to mark one or more Work Items as Complete whose status is On hold, Data Error, or Ready to run.
New	Allows you to mark one or more Work Items as New whose status is On hold, or Data Error
On hold	Allows you to mark one or more Work Items as On hold whose status is New
Customize columns	Allows you to show or hide specific columns. By default, all columns are displayed including the ones defined in the Work Item.

Alternately, select Work Items and use the following actions. Note that these actions can be performed only at a table level and not on individual Work Items.

Table Item	Description
View	Allows you to view details of selected Work Item.
Edit	Allows you to edit details of selected Work Item. You can see this icon only if you are the Queue Owner or Participant or Consumer and the status of the Work Item is Unsuccessful, On hold, or Data error
Delete	Allows you to delete the selected Work Item. Note that if a Work Item is in Active state, you are not allowed to delete it.

View automation of a queue

Use the View activity in progress page to view the automation details of the selected queue, Pause, Resume, or Stop the in-progress automation.

Prerequisites

Permissions required:

1. AAE_Queue Admin role
2. Queue Consumer or Queue Participant rights
3. Manage everyone's In progress activity feature permission

Note that though the View automation page is accessible from the Workload module, the page is launched from Activity module.

Procedure

1. Go to Workload > Queues
2. Hover over a queue with status In use
3. Click the Actions menu (vertical ellipsis) and select View automation.
   This launches the Activity > Run bot with queue > View page.
   - View the automation details of the queue such as:
     - The Bot name, path, dependencies, and if it requires Administrative privileges to run in the Task Bot tab.
     - The Queue used to run the automation in the Queue tab.
     - Bot Runner details such as Username, Device, and user Status in the Run as tab.
     - Details of the device pool in use to run the automation in the Device pool tab.
     - Details such as number of work items that were active, failed, pending, or completed processing in the Run history tab.
   - Pause
     The system will pause distributing work items from this queue to available bot runners in the device pool. Note: Until you resume this automation, any work items with Ready to Run status from this queue are not sent for processing.
   - Resume

     The system will start distributing the work items from this queue.

   - Stop

     The system stops distributing the work items from the queue associated with this automation.

     Select No, cancel to return to the details page or Yes, stop to stop the work item processing and return to the Queues page.

   Note that you cannot Pause/Resume or Stop actions directly from the Activity > In progress page. For these actions, the Workload > Queues > View automation action is used.

Next steps

Edit queues
Related tasks

View queue details
Delete Queues

Manage Work Items

Manage Work Items of a queue to fix the discrepancies before queue processing and reduce your automation-related errors.

Permissions required

You need any of the following permissions to manage the work items:

1. AAE_Queue Admin role
2. View and manage my Queues feature permission
3. Queue Owner, Queue Consumer, and/or Queue Participant permissions

- View work items
  View work items with a status of Completed, Unsuccessful, On hold, Active, or Data Error in the View work item page.
- Edit work items
  Use the Queues page or the Work item page to edit the work items of a queue.
- Delete work items
  Delete work items one at a time or in bulk in the View work item page.

View work items

View work items with a status of Completed, Unsuccessful, On hold, Active, or Data Error in the View work item page.

From the Workload > Queues page, you can:

1. Perform the following actions on either one queue or multiple queues:
   - Delete: This will delete the work item permanently.
   - Mark complete: This will mark the work item as complete.
   - Re-process:This will mark the work item in New state.
   - On hold: This will mark the work item as On hold.
2. Select a queue to View, Edit, or Delete a queue.

Procedure

1. Hover over a work item.
2. Click the Actions menu (vertical ellipsis).
3. Click View.
   The View work item page appears. The page provides details of the work item in four sections: Work Item Details, Work Item, Automation, and Work Item Results.
4. In the Work Item Details section, view the Status, Status Details, Start time and End time, and Queue Name.
   Actions allowed on view queue page
   Note: The Start time and End time are shown when the work item is being processed.
5. In the Work Item section, view the following:
   a) Attributes of the selected work item.

b) Audit log comments (if any) that were added when editing the work item.

6. In the Automation section, view the name of the automation, bot name, and device pool under which this work item was processed.
7. In the Work Item Results section, view the output status of the work item processed in the Enterprise Control Room. This is retrieved from the $workItemResult$ variable which is included in the bot created for workload automation.
8. In the General Details section, view the Last modified date and time, Modified by, and Object type.

Next steps

Edit work items

Delete work items

Related reference
Work item status and actions

Edit work items

Use the Queues page or the Work item page to edit the work items of a queue.

Prerequisites

You can edit a work item only if it is in a New, On hold, Data error, or Failed state.

Procedure

1. To edit a work item, use any of the following methods based on where you are in the Workload page:
   - Queues page:

     a) Hover over a work item to edit it.
     b) Click the Actions menu (vertical ellipsis).
     c) Click Edit.
   - Work Item page > Edit

   The work item page appears in edit mode.
2. Change the work item status to Mark complete, Defer, or Re-process in the Work item attributes and automation details section.
   The system will set the status to Data Error during the data load if there is any issue with the data. For example, if a user enters a text value for a number field, or an invalid date string for an attribute of date type, the status will be displayed as Data Error.
   Work item status and actions
3. Click Save changes.

Next steps

Delete work items

Delete work items

Delete work items one at a time or in bulk in the View work item page.

Procedure

1. Go to Workload > Queues.
2. Select and open the queue in view or edit mode.
3. Hover over a work item to view it.
4. Click the Actions menu (vertical ellipsis).
5. Click Delete.
   The selected work item is deleted successfully.
   Note: You can also delete a work item one at a time or in bulk using the Delete option provided above the Work items table.

Related reference
Work item status and actions

Managing packages

Users with the Manage package permission can upload and manage packages.Automation Anywhere provides you with the flexibility to decide which packages you want to make available to the Bot Creators for creating bots.

Users must have the appropriate administrative permission to view or manage action packages.

View packages

A user with View packages permission can view the packages that are available to Bot Creators. Go to the Bots > Packages page to view All packages.

The All packages page lists all the packages in the Enterprise Control Room that are available for Bot Creators. Packages can have multiple versions.

Manage packages

A user with the Manage packages permission can add new packages to the Enterprise Control Room and manage which packages versions are available in the Enterprise Control Room.

Add packages from the Bots > Packages > Add package page. Adding package options:

Reject

Stops the upload process.

Accept, enable and set as default

Uploads and enables the selected package, and setting it to the default package for the Enterprise Control Room.

Accept and enable

Uploads and enables the package, but the package is not set as the default package. Bot Creators have to specifically select non-default packages to use them for creating bots.

Manage packages from the Bots > Packages > All packages > View package page. Managing package options:

Set as default

Select a package and set is as the default. As soon as a package is set to default, it is the package that all Bot Creators in the Enterprise Control Room use.

Disable
: Disable a package so that users cannot use it to create new bots. Bots that were created using a disabled package continue to work.

Delete
: Deleting a package removes the actions contained in the package from the Enterprise Control Room for all users.
: Important: A package cannot be deleted if it is being used by a bot.

Note:

- Existing bots might be affected when existing packages are updated or disabled.
- It is recommended that Bot Creators update bots to use the latest version of a package.

Updates to packages by Automation Anywhere are available in each release of Enterprise A2019. The latest updated package can be set to the default package by users and administrators with Manage package permission.

Watch the following video on how to update a package in Enterprise A2019:

Update a package

- Add packages to the Enterprise Control Room
  Users with Manage package permission can add packages to the Enterprise Control Room for use by all Bot Creators.
- Manage Enterprise Control Room packages
  Manage packages in the Enterprise Control Room by setting a package as default, disabling it, or deleting it.

Add packages to the Enterprise Control Room

Users with Manage package permission can add packages to the Enterprise Control Room for use by all Bot Creators.

Prerequisites

Valid user login credentials for the Enterprise Control Room with Manage package permission are required.

Procedure

1. From the Bots > Packages page, click the Add package icon.
2. Browse to the location of the package to add.
   Packages are Java Archive (JAR) files that contain actions used to create bots.
3. Select the package to add, and click Upload package.
4. On the Bots > Packages > Confirm package page, choose any of the following options:

   Reject
   : Stops the upload process.

   Accept, enable and set as default
   : Uploads and enables the selected package, and setting it to the default package for the Enterprise Control Room.

   Accept and enable
   : Uploads and enables the package, but the package is not set as the default package. Bot Creators have to specifically select non-default packages to use them for creating bots.

Related concepts
Build and test a demo package and bot
Related tasks
Manage Enterprise Control Room packages
Add packages to the Enterprise Control Room

Manage Enterprise Control Room packages

Manage packages in the Enterprise Control Room by setting a package as default, disabling it, or deleting it.

Prerequisites

To manage packages in the Enterprise Control Room, users must have Manage package permission.

Package management actions apply to all users; however, the user can select specific package versions within a bot.

Procedure

1. From the Bots > Packages page, click View package.
2. From the Bots > Packages > View package, choose any of the following options:

   Set as default

   Select a package and set is as the default. As soon as a package is set to default, it is the package that all Bot Creators in the Enterprise Control Room use.

   Disable

   Disable a package so that users cannot use it to create new bots. Bots that were created using a disabled package continue to work.

   Delete

   Deleting a package removes the actions contained in the package from the Enterprise Control Room for all users.
   Important: A package cannot be deleted if it is being used by a bot.

Related concepts
Managing packages
Related tasks
Add packages to the Enterprise Control Room

Credentials and lockers

Sensitive information, such as user credentials, account numbers, and social security numbers that are included in automation tasks, are encrypted and stored as credentials centrally in the Credential Vault.

Credential Vault

Sensitive information does not need to be stored in bots or on Bot Runner systems. The Credential Vault facilitates a logical separation of credentials from the bots.

Credential Vault variables are created from the Enterprise Control Room and made available to all the Bot Creators and Bot Runners registered with the Enterprise Control Room. When bots are moved from one environment to another, no changes are required in the bots. Bots can seamlessly pick up the credential values applicable for the new environment from the Enterprise Control Room. Additionally, the Enterprise Control Room automatically stores configuration-related sensitive data into the Credential Vault by default.

Benefits of creating credentials

Apart from providing a secure and centralized location for storing credentials:

- Minimizes credential fraud.
- Provides an environment to enable improved security.
- Enables businesses to adhere to the processes and credential management compliance standards.
- Offers increased automation opportunities with secure data applications.

My Credentials tab

With the AAE_Locker Admin permission, you can view all user credentials. Other actions available under this tab include:

Create credential
: Create a credential and add the required attributes.

Create locker with checked items
: Select credentials to add to the locker.

View
: Roll over the view icon to see the details of the selected credential.

Delete
: Delete the selected credential.

In the search pane, filter credentials by credential name. The following information is displayed:

Type
: Shows the type of credential as user-provided or standard.

Name
: Name of the credential.

Locker Name
: Name of the assigned locker for the credential.

My Access

> Credential owner
> : Credential has been created by you.
>
> Credential non-owner
> : Credential has been created by other user.

Request Status

> All values provided
> : Value has been given.
>
> Requests sent
> : Request has been sent to users to input credential value.

Credential Owner
: Name of the user who has created the credential.

Last Modified
: Date and time when the credential was last edited.

Modified By
: Name of the user who has modified/edited the credential.

My Lockers tab

Credentials are further divided in logical groups called lockers. Lockers are used to group related sensitive information, that is included in automation tasks in the form of credentials, and share it with other users. Users with the following permissions can work with lockers:

Manage my lockers
: This permission enables you to create and manage lockers.

Administer ALL lockers
: This permission enables you to view all the lockers and do limited actions on them. This permission is available for AAE_Locker Admin role only.

The roles and permissions related to locker management are:

Owners
: A locker owner can edit, view, and delete a locker, and can add or remove other owners.

Manager
: A locker manager has access to the same functions as a locker owner, but does not have permission to add owners, managers, or participants to the locker.

Locker Participants
: A locker participant has access to view a locker and participants, and can also add own credentials to a locker. A locker participant does not have access or visibility of credentials created by other users.

Locker Consumers
: A locker consumer has access to view a locker and input credential value. To assign a locker consumer, select one or more user-defined roles. The users belonging to these selected roles are the consumers of the locker.

Note: Users can see lockers only if they have created them or if they are a member of that locker.
From the search pane you can filter lockers based on locker name. The following describes the list of items that can be viewed in the table:

Name
: Name of the locker.

My consumer permission
: Consumer or not a consumer.

My additional permission
: locker participant, locker manager, locker owner.

Managers
: Users with locker manager permission.

Credentials
: Number of credentials assigned to a locker.

Owners
: Name of user who created the locker.

Last Modified
: Date and time when the locker was last edited/modified.

Modified By
: Name of the user who has modified/edited the locker.

The following describes the list of performable actions that can be done on individual entries in the table:

View
: Enables you to view locker.

Edit
: Enables you to edit a locker.

Delete
: Enables you to delete a locker.

Credential Requests tab

Users can search for and request a credential from the presented list of available credentials.

- Set up locker and assign credentials
  The locker is used to group related sensitive information and can be shared with other users.
- Create credential
  Create a credential and add the required attributes.
- Create locker
  Create a locker to group similar credentials to share with other users.
- Credential Vault email notifications
  When the email notification setting is enabled, it ensures that users are notified of any changes to credentials and lockers.

Related tasks
Create credential
Edit a credential
Create locker
Set up locker and assign credentials

Set up locker and assign credentials

The locker is used to group related sensitive information and can be shared with other users.

Do the following to set up the locker and assign credentials:

Procedure

1. Create a role
   Define a role and assign permissions to access various features for building bots.
2. Create credential
   Create a credential and add the required attributes.
3. Create locker
   Create a locker to group similar credentials and share with other users.

Related tasks
Edit a credential
Edit a locker

Create credential

Create a credential and add the required attributes.

Users are enabled to add up to 50 attributes to each credential.

Procedure

1. Enter the Credential name and optionally, the Description.
2. Enter the Attribute name and optionally, the Description.
3. Select Input:

   Standard
   : Enter the value. All users see the same credential value set by the credential owner.

   User-provided
   : The value field is grayed-out because the values are not preset during creation. Only users of the locker containing this credential can provide the value.
4. Select Security:

   Masked
   : All masked values will be shown as asterisks.
5. Click Create credential.
   Assign the credential when adding the credential details. If no locker was created, create a locker and then assign the credential.

Next steps

Assign credentials to a locker. See Create locker.

- Edit a credential
  Edit details of a credential, such as the credential definition and value.
- View a credential
  As an authorized user, you can view details such as the credential details, attribute name, description, credential type and value, and general details of any credential.

Edit a credential

Edit details of a credential, such as the credential definition and value.

If a credential type is user-provided, then locker consumers have permission to edit the credential and their credential value.

Procedure

1. Go to Bots > Credentials
2. Select the credential and click Edit credential.
   If your credential is assigned to a locker, then you can only edit the value of common attribute. And if the attribute is user-provided, then the locker consumers can edit the value.
3. In the Edit credentials page, make the required changes.

   If email notification setting is enabled and credentials are added to a locker, then all the locker consumers shall receive an email. Learn more

- A credential can be edited by a credential owner, or if the credential type is user-provided then locker consumers can edit the credential value.

- In case of user-provided credential, you can only edit General information such as adding or removing a locker.

- In case of standard credential, you can edit General information such as adding or removing a locker and Attribute detail such as the credential value.

4. After you complete editing the credential, click Save changes or click Cancel to undo the changes. The maximum limit of credential attributes that is allowed is 50. If you have upgraded to the current version and have migrated credentials that have more than 50 attributes, when editing that particular credential, the following message displays: `Credentials can only have a maximum of 50 attributes`. To continue, remove the additional attributes that cannot be saved and add those to a new credential.

View a credential

As an authorized user, you can view details such as the credential details, attribute name, description, credential type and value, and general details of any credential.

Prerequisites

To view a credential, follow the steps mentioned below:

Procedure

1. Go to Bots > Credentials.
2. In My Credentials tab, choose the credential. Go to action list and click View credential. View credential page is displayed with the following details:
   - Edit credential- Allows you to modify the your credential.
   - Credential details- Description and credential owner.
   - Attribute name, credential description, type, value

   - General details- last modified (date and time), modified by, object type, credential type

Create locker

Create a locker to group similar credentials to share with other users.

Prerequisites

Make certain you have created the necessary credentials to add to a locker. One locker can hold up to ten credentials. A credential can only belong to one locker. See Create credential.
Credentials are further divided in logical groups called lockers. To create a locker, follow these steps:

Procedure

1. Navigate to Bots > Credentials

2. Type the locker name and optionally, type the locker description.
3. Select the Credentials to add to a new locker.
   Available credentials appear. Select one or multiple credentials from the list and add them to the locker.
4. Click on the Create locker with checked items.
5. Supply locker name and description.
6. Add the Owners.
   The locker owner can edit, view, and delete a locker and also add or remove other owners.
7. Type the Managers.
   The locker manager has access to functionality, but cannot add owners, managers, or participants to the locker.
8. Add the Participants.
   A locker participant has access to view a locker and add their own credentials to a locker.
   Note: A locker participant does not have access to or visibility of credentials created by other users.
9. Add the Consumers.
   Select one or more roles. Users belonging to these selected roles have access to the locker. System-created roles are not shown in the consumer list.
   If the credential type is:

   Standard
   : Locker consumers can view the locker and all the credentials inside the locker. All consumers see the same credential value set by the credential owner.

   User-provided
   : Locker consumers can input their information in user-provided credentials with user-provided attributes.

10. Click Create locker.

- Edit a locker
  Edit permissions for lockers.

Edit a locker

Edit permissions for lockers.

Prerequisites

To edit a locker, follow the steps mentioned below:

Procedure

1. Go to Bots > Credentials
2. In My Lockers tab, select the locker to edit. Then on the action list, click edit locker.
   Only a locker owner or locker admin has permission to edit a locker. You can make changes to the following:

   - Credentials- Add or remove credentials that are assigned to a locker.

   - Owners- Add or remove locker owners.
   - Managers- Add or remove locker managers.
   - Participants- Add or remove locker participants.
   - Consumers- Add or remove locker consumers.

     If email notification setting is enabled and credentials are added to a locker, then all the locker consumers will receive an email.

3. Click Save changes after you finish editing the locker.

Credential Vault email notifications

When the email notification setting is enabled, it ensures that users are notified of any changes to credentials and lockers.

Overview

Email notifications are sent for the following scenarios:

Credential is added to a locker

When credential is added to a locker, a notification is sent to all consumers of the locker to their email address registered in the Enterprise Control Room. The email consists of a link to the credential that is added to the locker. The consumers are redirected to edit the credential page wherein they input the credential value.

Member is added or removed from a locker

An email notification is sent when a new member (co-owner or participant) is added to a locker or removed from the locker as a member of participant.

Change in permission for locker members

When a locker owner/admin grants or removes locker membership permissions from a locker, an email notification is sent to the locker members at their email address. This ensures that members are notified of their membership changes within the locker.

Locker consumer gets added or removed from a role assigned to a locker, or consumer role gets added or removed from a locker

When a role assigned to a locker is modified by addition or removal of users, an email notification is sent to the new or existing user at their email address so that the consumers are notified that credentials are pending for their input in the locker.

Also when a new role added to a locker or an existing role is revoked from the locker, an email notification is sent to the new or existing consumers at their email address so that the consumers are made aware of the changes.

Bot Store

You can access the Bot Store from the Enterprise Control Room. From the Bot Store, you can download bots or packages to your Enterprise Control Room repository.

- Download bots to the Enterprise Control Room repository

  Download bots or packages from the Bot Store to your Enterprise Control Room repository.

- Submit bots to the Bot Store

  Submit bots or packages from the Enterprise Control Room repository to the Bot Store.

- Access the Bot Store from the Enterprise Control Room

  As a Bot Store registered user, you can log in to the Bot Store from the Enterprise Control Room.

- Submit bots or packages to the Bot Store
  You can submit bots or packages from the Enterprise Control Room to the Bot Store. Bots are submitted with dependencies. Packages are submitted to the Bot Store with a bot that demonstrates the use of the package.
- Download bots or packages to the Enterprise Control Room
  You can download bots or packages from the Bot Store to the Enterprise Control Room repository.

Related concepts
Bot permissions for a role

Access the Bot Store from the Enterprise Control Room

As a Bot Store registered user, you can log in to the Bot Store from the Enterprise Control Room.

Prerequisites

You must have valid Bot Store credentials to access the Bot Store from the Enterprise Control Room. If you do not have valid Bot Store credentials, you must register with the Bot Store.

Procedure

1. Log in to the Enterprise Control Room.
2. Click the Bot Store tab.
   The Bot Store opens in a separate window (https://botstore.automationanywhere.com).
3. Log in using your Bot Store credentials.
   In the Enterprise Control Room, you can see My downloads under the Bot Store tab.

Related concepts
Bot Store
Related tasks
Submit bots or packages to the Bot Store
Download bots or packages to the Enterprise Control Room

Submit bots or packages to the Bot Store

You can submit bots or packages from the Enterprise Control Room to the Bot Store. Bots are submitted with dependencies. Packages are submitted to the Bot Store with a bot that demonstrates the use of the package.

Prerequisites

You must have a system-created AAE_Bot Store Publisher role in order to submit bots or packages to the Bot Store. See Roles.

Procedure

1. Log in to the Enterprise Control Room.
2. Log in to the Bot Store using the Bot Store credentials.
3. Click BOTS > My bots.
4. Click Public workspace and select the Bot Store folder.
5. In the My bots page, move your mouse over the Action toolbar and click Submit to Bot Store.

a) In the Submit to Bot Store page, review the dependencies that will be bundled with your bot (including a parent bot) and make changes as required. In addition, ensure that the bot and all dependent bots and files are in the same folder to submit to the Bot Store.
b) Click Next to review the Bundled packages to ensure all packages are included with your submission.
c) If the bot to submit and the bundled package look complete, click Submit.
Alternatively, click Back to go back.
When you submit your files, the system displays the following message: `<bot/package name> submission is in progress. Go to Bot Store > My Submissions to complete your submission form. You will receive a confirmation email when the submission is completed.`
d) Click Take me to the Bot Store to complete your submission form in the Bot Store.

When you resubmit files to the Bot Store, it overwrites the previously submitted files, which are in a `Draft` status.

Related concepts
Bot Store
Related tasks
Access the Bot Store from the Enterprise Control Room
Download bots or packages to the Enterprise Control Room

Download bots or packages to the Enterprise Control Room

You can download bots or packages from the Bot Store to the Enterprise Control Room repository.

Prerequisites

You must have a system-created AAE_Bot Developer role in order to download bots or packages to your Enterprise Control Room repository. See Roles.

Procedure

1. Log in to the Enterprise Control Room.
2. Log in to the Bot Store using the Bot Store credentials.
3. Click Bot Store > My downloads.
4. In My downloads page, move your mouse over the Action toolbar and click Add to My bots.
   The following message appears: `If you have any files already installed in private workspace, they will be overwritten.`
5. Select Yes, continue to overwrite an existing file and continue with the installation, or select No, cancel to cancel your download.
   In My downloads page, the installation status of the selected file changes to `Installed`.
6. Verify the downloaded bots in the Bot Store folder of the private workspace.
   Custom packages downloaded from the Bot Store are available from the Packages repository. Or, you can can access them from the Bot Editor > Action Palette.
7. Verify that the downloaded packages have been enabled in the Packages repository. If the packages are not enabled, then go to individual packages and enable them.

Related concepts
Bot Store
Related tasks
Access the Bot Store from the Enterprise Control Room

Submit bots or packages to the Bot Store

Administration

Enterprise Control Room administrators manage settings related to the database, Credential Vault, , users, roles, action packages, licensing, and more.

Learn more about:

- Users management
  As a Cloud user with administrator permissions, you can create, view, edit, delete, enable or disable a user. Creating users steps vary depending if the user is a non-Active Directory, Active Directory, or an Single Sign On user from an IdP server.
- Roles
  Administrators configure roles for permission to perform actions such as create a bot, edit, or delete it for various features and operations in Enterprise Control Room.
- Settings
  Use the Settings tab to configure the connection to the Credential Vault, enable email notifications, integrate the Enterprise Control Room with a Git repository, enable secure recording mode, and configure user authentication.
- Licenses
  The All Licenses page displays detailed information about current product and device licenses.

Related tasks
Create a role
Create user
Installing licenses
Create credential

Users management

As a Cloud user with administrator permissions, you can create, view, edit, delete, enable or disable a user. Creating users steps vary depending if the user is a non-Active Directory, Active Directory, or an Single Sign On user from an IdP server.

Column Actions

- Click a column header to sort by ascending or descending order.
- Drag and drop a column header to move the column left or right.
- Drag the end of a column corner to re-size.

Individual User Actions

Do the following tasks on an individual user:

View
  Opens View user page in read-only mode. It shows user details, assigned roles, and general details, such as Last Modified, Modified by, Object type, and User type.
  You can edit a user detail and enable or disable a user.

Edit

Opens the Edit user page in write mode. It enables updates to user details, device login credentials, assigned roles, and device licenses.
When you edit a user, an email is sent notifying the user if SMTP is enabled.

Enable⊠Disable

Activates or deactivates the user. When you enable or disable a user, an email is sent notifying the user if SMTP is enabled.

Delete

Deletes the user. This is useful when users leave an organization or moved to another role. This free both the device, to which the user was attached, and the allocated license.
When you delete a user, an email is sent notifying the user if SMTP is enabled.

Table-level Actions

Do the following tasks by hovering over the icons at the top-right of the User table. These actions can be performed only at a table-level and not on individual items.

Create role with checked items

Adds a role and assigns the selected users. See Create user.

Delete checked items

Deletes the selected users. You cannot delete a user who is currently logged in.

Export to CSV

Exports the selected users in the table in CSV format.

Refresh

Refreshes the table and reflects the latest data.

Customize columns

Select the columns to show or hide in the table.

- Create user
  Add a new user by assigning a role and device license.
- Reset user password
  The Enterprise Control Room administrator generates an email process for the user to reset their password.

Create user

Add a new user by assigning a role and device license.

To create a new user, follow these steps:

Procedure

1. Go to Administration > Users.
2. Click Create user.
   The icon is located at the top-right of the User table. The Create user page is displayed.
3. In the General Details section, supply the following user details:

   Enable User

   Select for the user to be able to login immediately.

   Username

   Type a unique user name.

Description
: Optional. Type a description for the user.

First name
: Optional. Type the first name for the user

Last name
: Optional. Type the last name for the user.
: Note: The number of characters allowed in First name and Last name is 50.

Password
: Type and confirm a password for the user. Ensure the password follows any necessary password policy.

Email
: Type and confirm the email address for the user. If SMTP is enabled, the user is sent an email to this address to confirm the account. All important Enterprise Control Room notifications will be sent to this email address.
: Note: You can use the "@" character to accommodate email user-names.

4. In the Select roles section, assign a role from the Available roles table.
   Mark each role to assign. Select multiple roles for the user as necessary. Click ⊠ to move the roles to the selected column. For more information on specific roles and permissions, please see Roles.
   For example, to create a new Automation Anywhere Enterprise administrator, select AAE_Admin.
5. Assign a device license to the user.
   The following are license types:

   None
   : The user will have access to the Enterprise Control Room only.

   Bot Creator - Development license
   : Enables user to create and run bots. Auto login enabled by default.

   Unattended Bot Runner - Run-time license
   : Users with this license can perform all automation tasks that Attended users can perform. Additionally, this license can also be used for deployment, centralized scheduling, and API based deployment.

   Attended Bot Runner - Run-time license
   : Users with privilege to run bots on their workstations. These users can also make use of local schedules and triggers for time or event based automation.

6. Click Create user or Create user and add another.
   The new user is displayed in the User table. If SMTP is enabled, an email will be sent to the new user inviting them to log in.

- Create an Active Directory user
  Add the Active Directory user by selecting AD domain, providing AD environment details, and assigning a role and device license.

Related concepts
Licenses - an overview

Create an Active Directory user

Add the Active Directory user by selecting AD domain, providing AD environment details, and assigning a role and device license.

To create a new user, follow these steps:

Procedure

1. Click Create user.
   The icon is located at the top-right of the User table. The Create user page is displayed.
2. In the General Details section, supply the following information:

    a) Enable User

    Select for the user to be able to login immediately.

    b) Active Directory domain
    Select the active directory name for the user. The list displays all domains that are available in the domain controller.
    Note: Enterprise Control Room Active Directory supports single forest multi-domain environment.
    c) Username
    Click CHECK NAME IN ACTIVE DIRECTORY. If the user name is present in the Active Directory database, the First name, Last name, Email, and Confirm email fields are auto-populated. If the data is not auto-populated from, type the details into the fields.

    If the username is not present in the Active Directory database, an error message is displayed. Contact the network administrator to resolve the issue.

    d) Password

    Type and confirm a password for the user. Ensure the password follows any necessary password policy.

    e) Email
    Type and confirm the email address for the user. If SMTP is enabled, the user is sent an email to this address to confirm the account. All important Enterprise Control Room notifications will be sent to this email address.
    Note: You can use the "@" character to accommodate email user-names.
    f) Password

    Type a password for the user. Ensure that you are assigning a password that follows the password policy of your organization.

    g) Confirm Password

    Type the password again. This should be same to what you typed in the Password field.

3. In the Select roles section, assign a role from the Available roles table.
   Mark each role to assign. Select multiple roles for the user as necessary. Click ⊠ to move the roles to the selected column. For more information on specific roles and permissions, please see Roles.
   For example, to create a new Automation Anywhere Enterprise administrator, select AAE_Admin.
4. Assign a device license to the user.
   The following are license types:

   None
       The user will have access to the Enterprise Control Room only.
   Bot Creator - Development license
       Enables user to create and run bots. Auto login enabled by default.
   Unattended Bot Runner - Run-time license
       Users with this license can perform all automation tasks that Attended users can perform. Additionally, this license can also be used for deployment, centralized scheduling, and API based deployment.

Attended Bot Runner - Run-time license
: Users with privilege to run bots on their workstations. These users can also make use of local schedules and triggers for time or event based automation.

5. Click Create user or Create user and add another.
   The new user is displayed in the User table. If SMTP is enabled, an email will be sent to the new user inviting them to log in.

Reset user password

The Enterprise Control Room administrator generates an email process for the user to reset their password.

The change password email process.

Procedure

1. Administrators navigate to Administration > Users.
2. Select the desired user from the list and click Edit user.
3. Click Send reset password email.
   The selected user receives an email with the necessary instructions to reset the password.
   Note: If there is not an email server configured, please follow these steps to reset a password for a user:
   a) Open the URL for the Enterprise Control Room in your browser.
   b) Enter the user name, click Forgot Password, and follow the prompts to reset or change the password.
   c) Enter the user name, click the Forgot Password button, and follow the prompts to reset or change the password.

Related tasks
Log in to Automation Anywhere Enterprise Control Room

Roles

Administrators configure roles for permission to perform actions such as create a bot, edit, or delete it for various features and operations in Enterprise Control Room.

Roles

Role-based access control (RBAC) grants access to users based on the assigned roles and the accessibility provided to the user. The benefits of creating roles include:

- Increased security through controlling users access according to their specified roles.
- Decreased need of customer support.
- Easy and accurate monitoring of the use and access of data by higher management, leading to better research management.

The Enterprise Control Room enforces role-based access control. There are two types of roles:

System-created
: By default, these roles are preconfigured.

User-created

Users create these roles and the roles can be customized. If a user-created role is created with all Enterprise Control Room permissions, it is not considered an Enterprise Control Room Admin role. Only the system-created Admin role has this privilege.

Default roles

AAE_Admin

This role allows access to all features, including creating other Admin users and access to all folders and files. The only role that can access Enterprise Control Room settings.

AAE_Locker Admin

This role allows to view all credentials and all lockers. A Locker Admin can change the owner of a credential that they do not own. For lockers they do not own, they can delete the locker, edit permissions, and remove credentials.

Note: This permission is not applicable to Enterprise Control Room Admin role.

AAE_Basic

This role provides permissions to upload and download TaskBots in the My Tasks folder. Limited access to other features.

AAE_Pool Admin

This role allows user to view and manage all device pools.

Note: Users with AAE_Pool Admin do not have permission to see any bots and supporting files.

AAE_Queue Admin

This role allows the user to view and manage all queues.

AAE_Bot Insight Admin

This role provides permission to view and manage data in Bot Insight. Limited access to Enterprise Control Room features. (If Bot Insight license is installed). This allows a user to access Bot Insight RESTful APIs to get access to the data logged by the Enterprise Control Room, and by a task during 'Production' run.

AAE_Bot Insight Consumer

This role provides permission to view data in Bot Insight. Limited access to Enterprise Control Room features. (If Bot Insight license is installed)

AAE_Bot Insight Expert

This role provides permission to manage data in Bot Insight. Limited access to Enterprise Control Room features. (If Bot Insight license is installed)

AAE_Bot Developer

This role allows users to download bots or packages from the Bot Store to the Enterprise Control Room private workspace.

AAE_Bot Store Publisher

This role allows users to submit bots or packages to the Bot Store.

AAE_IQ Bot Validator

This role allows user to access the IQ Bot Validator screen. Limited access to Enterprise Control Room features. (For a Bot Runner with an IQ Bot license).

AAE_IQ Bot Services

This role grants a user the permissions to access the IQ Bot console. Limited access to Enterprise Control Room features.

Bot Insight, and IQ Bot roles are displayed only if you have respective licenses.

Permissions for roles

Only an administrator or Enterprise Control Room user with roles permission can assign roles to users and provide access to them for various features and operations. Assign the following permissions to a role:

Dashboard

View dashboards. Available to all users.

Activity

All users can view their own activity.

- In progress
- Scheduled
- Event Triggers
- Historical

Bots

View bots, credentials, values, and packages

- My bots
- Credentials
- Global values
- Packages

My Devices

View and manage my Bot Runners, and device pools.

Workload

View and manage my queues.

Bot Store

- View Bot Store allows all Enterprise Control Room users to view the Bot Store.
- Add bots from Bot Store to My Bots allows users to add bots or packages from the Bot Store to the Enterprise Control Room private workspace.
- Submit bots to Bot Store allows users to submit bots or packages to the Bot Store.

Audit Log

View all audit log actions.

Administration

View and manage settings and is available only for the Enterprise Control Room and Community Control Room administrators and cannot be granted to any other roles.

Table-level Enterprise Control Room actions

Use the table-level actions to perform the following tasks:

Create user

Creates a user and assigns the selected roles.

Create role with checked items

Creates a role with selected features.

Export checked items to CSV

Exports a selected item to a CSV file.

Delete checked items

Deletes selected roles. A role cannot be deleted if there are users assigned to it.

Refresh table

Refreshes the table.

Customize columns

In the table, shows or hides the column.

- Create a role
  You can define a role and assign permissions to access various features of the Enterprise Control Room.
- Bot permissions for a role
  Assign bot permissions when creating a role.

Related concepts
Bot permissions for a role
Related tasks
Create a role

Create a role

You can define a role and assign permissions to access various features of the Enterprise Control Room.

Prerequisites

Only an admin or Enterprise Control Room user with the Manage role permission can assign roles to users and provide them with access to various features and operations.

Procedure

Follow these steps to create a role:

1. Go to Administration > Roles.
2. Click Create role.
3. Enter a Role name, and optionally Role description.
4. Select a new role permissions:
   Default permissions:
   a) View dashboards
   b) View my in progress activity
   c) Manage my credentials and lockers
   d) View and manage my Bot runners, Bot creators and device pool
   e) View and manage my queues

   Activity
   Allows users to manage in progress bot activities.
   Event Triggers
   Allows users to manage event trigger options.
   Bots
   Allows users to manage bots options.
   Package Manager
   Allows users to manage packages.
   Devices
   Allows users to manage device options.
   Workload
   Allows users to manage queues.
   Bot Store
   - View Bot Store
   - Add bots from the Bot Store to My Bots
   - Submit bots to the Bot Store

Audit Log
: Allows users to manage audit log actions.

Administration
: Allows users to access administrative permissions.

API
: Allows users to access API options.

IQ Bot
: Allows users to view IQ Bot options.

5. Click Next.
6. In the Users tab, assign your role to existing users.
   Users shown as disabled cannot be selected if they have been deactivated by an Admin user. Also, your own user is reflected as disabled in the users' list and cannot be removed.
   Tip: You can select multiple users for your role in the Users tab. This allows more than one user to be assigned the same role at a time, which reduces the effort unlike the Users landing page.
7. After you complete selecting users for your role, click Create role.

Related concepts
Bot permissions for a role
Related reference
Roles

Bot permissions for a role

Assign bot permissions when creating a role.

TaskBots and other supporting files

Select from the following permissions:

- Select all - This permission includes upload, download, execute, delete, Run plus Schedule actions.
- Run plus Schedule - This permission includes run and schedule permission for TaskBots.
  - This permission is termed as Run when the user has Run my bots feature permission. You can explicitly select Run permission on a specific folder to allow the user to run all bots that belong to this folder.
  - This permission is termed as Schedule when the user has Schedule my bots to run feature permission. You can explicitly select Run permission on a specific folder to allow the user to schedule all bots that belong to this folder.
  - Run plus Schedule when user has both feature permissions. This allows the user to run and schedule bots that belong to this specific folder on which the permission is selected.
- Upload: This permission allows users to upload TaskBot files/ folder to Enterprise Control Room from Enterprise client.
- Download: This permission allows users to download TaskBots from the Enterprise Control Room.
- Delete: This permission allows users to remove files and their dependencies from the Enterprise Control Room.
- View Dashboards: This permission allows users to view Bot Insight dashboards.
- Bot Store
  - View Bot Store: This permission allows users to view the Bot Store.
  - Add bots from the Bot Store to My bots: This permission allows users to download bots from the Bot Store to the Enterprise Control Room repository.
  - Submit bots to Bot Store: This permission allows users to submit bots to the Bot Store.

Related tasks
Create a role

Settings

Use the Settings tab to configure the connection to the Credential Vault, enable email notifications, integrate the Enterprise Control Room with a Git repository, enable secure recording mode, and configure user authentication.

Credentials

Manage the Connection mode to connect to the Credential Vault using a Master key.

Configure Credential Vault Connection mode.

Email

All users have to confirm email accounts by clicking the confirmation link that they receive, set the password, and security questions before user can log in to the Automation Anywhere Enterprise Control Room. By default, email notifications are disabled. Mouse over the Edit icon to make changes.

Edit email notifications

IQ Bot

View the website address where IQ Bot is currently installed, if applicable. Click Edit to update the IQ Bot URL.

Secure recording mode

Click Edit in the Bots tab to enable or disable secure recording mode.

Secure recording mode ensures that sensitive data is not stored in the bots. When secure recording mode is enabled, the bots do not capture values of certain properties or store application images. This setting applies to all the users of the Enterprise Control Room.
Note: Secure recording mode only applies to bots that are created or edited after the mode is enabled.

User authentication

Configure the to authenticate users through the database option or switch to a SAML identity provider (IDP).

Set up SAML authentication

- Set up SAML authentication
  Switch an authenticated environment Enterprise Control Room database to a SAML identity provider (IDP).
- Configure Credential Vault Connection mode
  Credential Vault is a centralized location for securely storing credential information used by bots.
- Edit email notifications
  Specify details about the email server you want to use and the events when the email notification must be sent.
- Integrating Enterprise Control Room with Git repositories
  You can integrate the Enterprise Control Room with remote Git repositories so that you can manage bot version controls, back up and restore bots and the dependent files, perform visual code comparison, and manage interoperability with code scanning tools.

- Installing Enterprise Control Room for Cloud-enabled deployment
  You can store and process native business and operational data in an On-Premises deployment and take advantage of management and operational services from the Cloud.

Related tasks
Set up SAML authentication
Edit email notifications

Set up SAML authentication

Switch an authenticated environment Enterprise Control Room database to a SAML identity provider (IDP).

Prerequisites

Sign in to the Enterprise Control Room as an Admin user. The SAML IDP side setup must be validated before configuring the Enterprise Control Room.

To set up the Enterprise Control Room as a service provider in the SAML IDP, follow these steps:

1. Set the ACS or service provider URL to <Enterprise Control Room URL>/v1/authentication/saml/assertion.
2. Create an Entity ID, that is, any name that identifies the Enterprise Control Room on the SAML IDP.
3. Map the following Enterprise Control Room attributes to the corresponding IDP attributes:
   - UserID
   - FirstName
   - LastName
   - EmailAddress
4. Get the service provider metadata, generated as an XML file, from the SAML IDP for the Enterprise Control Room.

   This is required for setup within the Enterprise Control Room.

   Note: You have to add the values from Steps 2 and 4 in the Enterprise Control Room to complete the setup.

Procedure

To switch the Enterprise Control Room to a SAML authenticated environment, follow these steps:

1. Navigate to Administration > Settings.
2. Access User Authentication > Edit.
3. Select the Use SAML option to enter the SAML information.
   Note: The Use Control Room database option is selected by default.
4. In the SAML metadata field, enter the data from the SAML IDP.
5. In the Unique Entity ID for Control Room (Service Provider) field, enter the Entity ID.
6. In the Encrypt SAML Assertions field, select one of the following options:
   - Do not encrypt: the SAML assertions are not encrypted.
   - Encrypt: the SAML assertions are encrypted.
7. Optional: Enter the Public key and Private key values.
   Note: Enter keys if you require encrypted SAML assertions.
8. Click Validate SAML Settings.
   You have to validate your SAML settings before you can save your changes.
   When you click this option, you will be redirected to a SAML service provider web page where you will be prompted to enter credentials and other data. After validation is complete, you will be redirected back to this configuration page.

9. Log in to the page and perform these steps:
   a) Navigate to the Metadata Manager and add the new service provider.
   b) Enter Enterprise Control Room metadata in the required field.
   c) Enter the Entity ID for the Enterprise Control Room service provider.
   d) Select the option to retrieve the user's information such as username, first name, last name, email.
   e) Save the new service provider.
10. Click Save changes.
    After you have successfully saved your settings, you will be logged out of the Automation Anywhere Enterprise Control Room.
11. Log back in to the system with your new credentials.

Configure Credential Vault Connection mode

Credential Vault is a centralized location for securely storing credential information used by bots.

Configure the Connection mode to connect to the Credential Vault using a Master key.

Note: Provide this key every time you start or restart the Enterprise Control Room.

To configure settings for Credential Vault, select Express or Manual mode.

Express mode
: Auto connect to the Credential Vault with the master key that is stored in the system during Enterprise Control Room configuration.

Manual mode
: Use this to manually connect to the Credential Vault using the master key that was available during Enterprise Control Room configuration.

When switching modes, provide the Master Key in the field and click Save for the changes to take effect.

Tip: Restart the server machine (on which the Enterprise Control Room is installed) or services to allow changes to take effect.

All updates to the Credential VaultConnection mode are captured in the Audit Log.

Related tasks
Set up locker and assign credentials

Edit email notifications

Specify details about the email server you want to use and the events when the email notification must be sent.

Procedure

To configure Email settings, perform these steps:

1. Click ADMINISTRATION > Settings.
2. Expand the Email tab and click Edit.
3. Select Send email notifications.
   By default, the email notifications are disabled.
4. Specify the details of the server that you want to use to send email notifications:

- From this email address: Enter the email address that you want to use to send email notifications.
- Email server host: Specify the email server that you want to use to send email notifications.
- Email server port: Specify the email port that you want to use to send email notifications.
- Select the My server uses a secure connection (SSL/TLS) option if the server uses a secured connection.
- Select the My server requires authentication option if the server requires credentials for access.

  Specify the Username and Password you want to use to access the server.

Important: The options to specify or modify the email server details are available only for the On-Premises deployment.

5. Select any or all of the following option to specify the events when an email notification must be sent:
   - User initiates Forgot Password process from Login screen
   - User information changes, to the user
   - A user is activated, deactivated or deleted, to the user
   - A Task Bot stops running because it is unsuccessful, to the user who started or scheduled it
   - A BLM package is exported or imported, to the user who performed BLM export or import
6. Click Save changes.

Integrating Enterprise Control Room with Git repositories

You can integrate the Enterprise Control Room with remote Git repositories so that you can manage bot version controls, back up and restore bots and the dependent files, perform visual code comparison, and manage interoperability with code scanning tools.

Git integration with the Enterprise Control Room ensures one-to-one mapping of the bots between the Enterprise Control Room and the remote Git file structure. Git commits enable you to enforce security, compliance, and code standards, and ensures that an organization's established best practices can be applied to their bot development processes.

The Enterprise Control Room maintains a Git repository that stores the file history of your public workspace. Each time a bot developer performs a bot check-in, the bot and the dependent files are checked-in to the public workspace. The Enterprise Control Room creates a commit that contains all the contents of the particular check-in within the Git repository. The comment entered in the check-in process becomes the Git commit message and the user details are recorded as the author of the Git commit.

The Enterprise Control Room Git integration ensures one-to-one mapping of the bots checked-in to the public workspace of the Enterprise Control Room and the remote Git file structure. Bot definitions are stored as JSON files in the Git repository.

Supported Git repositories

You can configure the Enterprise Control Room to replicate the built-in Git repository with the remote Git host of your choice, and synchronize information using Git push. Examples of remote Git hosts include, but is not limited to:

- GitHub
- GitLab
- BitBucket

The Enterprise Control Room supports HTTPS connectivity to your remote Git host and uses standard Git push protocols to send or receive data. Exposing these files to your remote Git host enables you to review the bot code and files using third-party comparisons and static code analysis tools to help enforce compliance and maintain security standards.

Supported Git versions

You can integrate the Enterprise Control Room with any remote Git repository that supports Git push using HTTPS, which uses a username and password authentication method and supports Large File Storage (LFS).

For information about enabling Git LFS on your remote Git host, see Git LFS.

- Configure a remote Git repository in Enterprise Control Room
  Configure a remote Git repository in the Enterprise Control Room so that all the bot information that is stored in the Microsoft SQL Server is synchronized with the remote Git host. When you check in a bot, the local Git repository pushes the bot and its dependent files to the remote Git repository.

Configure a remote Git repository in Enterprise Control Room

Configure a remote Git repository in the Enterprise Control Room so that all the bot information that is stored in the Microsoft SQL Server is synchronized with the remote Git host. When you check in a bot, the local Git repository pushes the bot and its dependent files to the remote Git repository.

Prerequisites

In order to configure a remote Git repository in the Enterprise Control Room, ensure that you have the `AAE_Admin` role assigned to you.

Procedure

1. In the Enterprise Control Room, navigate to Administration > Settings > Git Integration.
2. In the Git Integration window, select Set up Git Integration.
3. In the Credentials menu, perform these steps:
   a) Enter your Git repository path in the User Name field in the following format:

   ```
   https://<name>@bitbucket.org/<companyname>/product.git
   ```

   b) Enter your Git repository password and confirm your password.
   c) Click Submit.
4. Click Connect and Save.
   The Enterprise Control Room connects to the remote Git repository.

Installing Enterprise Control Room for Cloud-enabled deployment

You can store and process native business and operational data in an On-Premises deployment and take advantage of management and operational services from the Cloud.

Automation Anywhere deploys and configures an Enterprise A2019 Cloud instance for this Cloud-enabled deployment option. Customer then installs the On-Premises application within their infrastructure for storing and processing customer data.

Note: Linux is not supported for Cloud-enabled On-Premises installations.
For Cloud-enabled deployment, the initial welcome email that you receive from Automation Anywhere you will find:

- URL to the Cloud instance
- Username and password
- Provisioning token needed to establish trust connectivity with the Automation Anywhere Cloud

Important: Do not discard the content of this email. You will need the information in the email to setup on-premises application. The cloud instance URL, username and password will be needed if you have to regenerate the token required to establish cloud connectivity.

Procedure

1. Receive your Cloud instance login credentials, with administrator privileges, the Enterprise A2019 dedicated URL from Automation Anywhere Enterprise and provisioning token.
2. Install and access the On-Premises instance.
   Receive and install Enterprise A2019; the installation user is assigned administrator privileges.

   Enterprise A2019 On-Premises Enterprise Control Room installation

3. Log in to the On-Premises Enterprise Control Room.
   Log in to Automation Anywhere Enterprise Control Room
4. Navigate to Administration > Settings > Cloud-enabled.
   This is where you will link the two instances.
5. Provide the provisioning token and URL for the On-Premises Enterprise Control Room.
6. Click Save changes.
   The trusted relationship between the instances is created.
   Note: IQ Bot registration is not supported on Cloud-enabled On-Premises installations.
7. To test Regulated Cloud functionality, open a browser, enter the URL of the Cloud Control Room, and press Enter.
   You are redirected to the On-Premises Enterprise Control Room.

Related concepts
Post-installation user management
Related tasks
Register device and install Bot agent
Related reference
Installed Enterprise Control Room directories and files

Licenses

The All Licenses page displays detailed information about current product and device licenses.

Product licenses

The Automation Anywhere Enterprise Control Room is the web-based application at the center of the Digital Workforce providing enterprise-wide management and control. The Enterprise Control Room ensures reliable, scalable, and secure bot deployment and execution. From this central vantage point, operators can receive tasks from the Bot Creator and push to the Bot Runners for execution with simple mouse clicks. The Automation Anywhere Enterprise Control Room monitors and audits all scheduled and running bots, in real time.

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

Device licenses

Bot Creator

The Bot Creator license provides the capability to create, schedule, trigger, and edit bots.

Bot Runner

The Bot Runner license provides authorization to execute bots, independently and asynchronously.

Unattended Bot Runner - Run-time license

Users with this license can perform all automation tasks that Attended users can perform. Additionally, this license can also be used for Automation Anywhere Enterprise Control Room deployment, centralized scheduling, and API-based deployment.

Attended Bot Runner - Run-time license

Users with privilege to run bots on their workstations. These users can also make use of local schedules and triggers for time-based or event-based automation.

IQ Bot A2019

IQ Bot automates business processes that rely on semi-structured or unstructured data. IQ Bot licenses are purchased based on the number of pages of processing required.

Bot Insight

Bot Insight provides real-time, RPA native analytics for both business insights and operational intelligence. Bot Insight Analytics license is purchased on a per user basis.

Entitlement models

Two licensing models are available for Automation Anywhere Enterprise Version A2019:

File-based entitlements

When Version A2019 operates in a file-based entitlement mode:

- A license file is configured, generated, and installed for each Control Room.
- The Control Room administrator can then issue these licenses to specific user accounts.
- Each user consumes a license within a Control Room. If the same user is created in multiple Control Rooms, they will use up a license entitlement for each Automation Anywhere Enterprise Control Room.
- File-based entitlements only supports a floating user license model.

Cloud-based entitlements

Available and accessed from a cloud-based license server. Information exchanged between the Control Room and the license server meet GDPR compliance requirements. If you cannot allow access to an external service, such as the License Service, because of network or security constraints, contact Automation Anywhere support.

- The cloud-based GUID can be installed only if there are no users file licenses in use.
- Administrators can reallocate user licenses after installing the cloud-based GUID.

RBAC on License Management

Access to License Management is deny-all and allow by exception based on roles and domains as defined in RBAC. Only those users who have access to License Management permission can view the entitlement details from the Automation Anywhere Enterprise Control Room.

Baseline inventory controls: Bot Creators, Bot Runners, and Bots

The Automation Anywhere Enterprise Control Room manages all automation operations. Inventory controls are maintained through the application of RBAC to establish a single point of control for Base Line Configurations (NIST CM-2), access restrictions for configuration management (NIST CM-5 and 6). Automated baseline reporting can be configured.

- Licensing and entitlements
  Any new customer who orders Automation Anywhere Enterprise products are to receive license confirmation from Automation Anywhere.
- Installing licenses
  Upload a new license into the Automation Anywhere Enterprise Control Room.
- Configure new Enterprise Control Room licenses
  The Enterprise Control Room in your order now requires configuration to generate and download new licenses.
- Enterprise Control Room Fail-Safe status
  When the Enterprise Control Room is unable to connect to the license server, it moves into Fail-Safe status.
- Installing licenses
  Upload a new license into the Automation Anywhere Enterprise Control Room.
- Enterprise Control Room Fail-Safe status
  When the Enterprise Control Room is unable to connect to the license server, it moves into Fail-Safe status.

Installing licenses

Upload a new license into the Automation Anywhere Enterprise Control Room.

Prerequisites

Administrative privileges are required to make changes to the licenses.

Note:

- The cloud-based GUID can be installed only if there are no users file licenses in use.
- Administrators can reallocate user licenses after installing the cloud-based GUID.

Be logged into the Automation Anywhere Enterprise Control Room as the administrator.

Procedure

1. Navigate to Administration > Licenses.
2. Select Install license from server or Install license from file.

Option	Action
Install license from server	a) Release all file based license allocations from users.

Option	Action
	b) Supply the unique Control Room GUID. c) Click Install license from server.
Install license from file	a) Browse to and select the license. b) Click Install license.

Related reference
Users management
Roles
Settings

Enterprise Control Room Fail-Safe status

When the Enterprise Control Room is unable to connect to the license server, it moves into Fail-Safe status.

With respect to the Enterprise Control Room license server database, the Enterprise Control Room can be in one of three status states. These states indicate what user licensing actions can be done. With each state change, an entry is made in the audit log.

Active
: Normal operations. All API calls from the Enterprise Control Room are accepted by the license server.
Users can be assigned floating licenses as they log on. Floating licenses can be released as users log off.

Fail-Safe
: Only the heartbeat API call is allowed to the license server. All other calls from the Enterprise Control Room are stopped.
Operations, such as granting logging in users a license, or deleting a license assigned to a logged in user are restricted.

Fail-Safe-Expired
: The Enterprise Control Room stops all operations, all users are logged out of the Enterprise Control Room.

Fail-Safe mode actions

When the Enterprise Control Room loses connection with the licensing server and moves into Fail-Safe mode:

- The Enterprise Control Room administrator is sent an email notification, saying the Enterprise Control Room is in Fail-Safe mode. The administrator can take remedial action to re-establish the connection.
- Currently logged in users continue to have access and can do tasks.
- User licenses cannot be allocated to or de-allocated from users.

Fail-Safe-Expired mode actions

When the Fail-Safe time limit expires, the Enterprise Control Room moves into Fail-Safe-Expired state:

- All connected users are shut down.
- The Enterprise Control Room reports Shutdown status to the license server.
- The Enterprise Control Room shuts down.

Active mode actions

When the Enterprise Control Room is restarted and operational, and connectivity to the license server established, the Enterprise Control Room is in Active state:

- Users who had assigned licenses prior to the Fail-Safe, have their original licenses re-allocated.
- New users can request and be allocated licenses.

Enterprise Control Room log files

Various types of information about the Enterprise Control Room are captured in different log files. You can analyze these log files when the Enterprise Control Room or a bot encounters an error and identify the root cause for that error.

Overview

The log files capture information about the errors encountered by various components of the Enterprise Control Room such as the Bot Store, devices, and license. You can use the information in the log files to identify the root cause of an error and resolve that error.

Log file locations

The Enterprise Control Room log files are available at C:\ProgramData\AutomationAnywhere\Logs on the machine on which you have installed the Enterprise Control Room. The log files are named WebCR_<COMPONENT-NAME>, for example, WebCR_BotStore, WebCR_Devices, and WebCR_License.

Enterprise Control Room APIs

The Automation Anywhere Enterprise Control Room provides various public APIs which allow you to customize your business automation for third-party applications.

These reference topics provide information that you can use with different APIs.

Filters in an API request body
: Filtering provides basic conditional queries and page control for processing API requests. There are 3 basic features related to filtering: filtering conditions, sorting, and pagination parameters.

Permissions to roles mapping
: Create roles from the Enterprise A2019 Administration user interface or through the User Management API by assigning a set of permission that enable users to access related features.

These APIs enable the third-party applications to consume RPA, orchestrate bots, and manage the RPA data based on events.

- Audit API
  Requests audit data for a given input combination of date filter, sorting mechanism, and pagination.

- Authentication API overview
  Use the Authentication API to generate, refresh, and manage JSON Web Tokens (JWT) that are required for authorization in all Enterprise Control Room APIs.
- Bot Deploy API
  The Bot Deploy API supports the runAsUser feature for bot deployment.
- Bot Execution Orchestrator API
  As an Enterprise Control Room administrator or a user with View and Manage Scheduled Activity permission, you can monitor bot progress using a set of Enterprise Control Room APIs.
- Bot Insight API
  Get bot process data for analytic analysis. Only users with Bot Insight administration role can access this API.
- Bot Lifecycle Management API overview
  Use the Bot Lifecycle Management API to export and import bots with dependent files and action packages for comprehensive automation life-cycle management.
- Credential Vault API overview
  As an Enterprise Control Room user with Manage my credentials and lockers role permissions, you have the option to use the Credential Vault API to manage your attributes, credentials, keys, lockers, and Credential Vault mode in the Enterprise Control Room.
- Migration APIs
  The Migration APIs allow users with the appropriate migration permission to view or manage bot migration from 11.x to A2019, including starting the migration of bots and retrieving details about migrations.
- Repository Management API
  The Repository management API is a role based access API that returns information for folders and files that you have permission to view in your Enterprise Control Room.
- User management API overview
  The User Management APIs enable you to create, search, update, or delete roles and users in your .
- Workload Management API overview
  Use the Workload Management (WLM) API to programmatically manage and create workitem models, queues, workitems, and automations in your Enterprise Control Room.
- Filters in an API request body
  Filtering provides basic conditional queries and page control for processing API requests. There are 3 basic features related to filtering: filtering conditions, sorting, and pagination parameters.
- Permissions to roles mapping
  Create roles from the Enterprise A2019 Administration user interface or through the User Management API by assigning a set of permission that enable users to access related features.

Audit API

Requests audit data for a given input combination of date filter, sorting mechanism, and pagination.

Prerequisites

You can view the Audit API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.

JSON Web Token (JWT)
    All Enterprise Control Room APIs require a JSON Web Token (JWT) to access the APIs. Generate an authentication token using the Authentication API. A JWT is required to run all Enterprise Control Room APIs.

Roles and license

Users with the AAE_Admin role or users with the View everyone's audit log actions permission are able to view audit logs for the Enterprise Control Room.

- URL: `http://<your_control_room_url>/v1/messages/list`
- Method: POST

Note: Use the Swagger definition files installed with your Enterprise A2019 Edition to test the APIs, or use a REST client.

Procedure

1. Add an authentication token to the request header.
   Note: Use the Authentication API to generate a JSON Web Token.
2. Select POST as the method.
   Note: Apply filters to perform basic conditional queries and pagination control for processing web pages. There are three basic features related to filtering: filtering conditions, sorting columns, and pagination parameters. Refer to the Filters in an API request body.

   The following example requests unsuccessful login attempts for the month of December.

   Request body:

```
{
  "sort": [
    {
      "field": "createdOn",
      "direction": "desc"
    }
  ],
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "gt",
        "field": "createdOn",
        "value": "2019-12-01T00:00:00.001Z"
      },
      {
        "operator": "lt",
        "field": "createdOn",
        "value": "2019-12-31T23:59:59.999Z"
      },
      {
        "operator": "eq",
        "field": "status",
        "value": "Unsuccessful"
      },
      {
        "operator": "substring",
        "field": "activityType",
        "value": "LOGIN"
      },
      {
        "operator": "substring",
        "field": "userName",
        "value": "joe.typical@myemiil.com"
      }
    ]
  },
  "fields": [],
  "page": {
    "length": "1000",
    "offset": "0"
  }
}
```

3. Send the request.
    - In Swagger, click Execute.
    - In a REST Client, click SEND.

    The response for this example returned data for date filter, sorting, and pagination. When there is no filtering used in the request, a successful response returns all pages for the specified Enterprise Control Room.

    Response body:

```
{
  "page": {
    "offset": 0,
    "total": 731064850,
    "totalFilter": 9
  },
  "list": [
    {
      "id": "XlHj6G4BFXSpOOji5B7S",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemiil.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T04:21:19Z",
      "requestId": "04965c2e-82e0-4ce4-a88d-bebe1dc3a2a8",
      "createdBy": "0"
    },
    {
      "id": "g1Hj6G4BFXSpOOji2Rwx",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemiil.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T04:21:16Z",
      "requestId": "61672553-477d-4012-ab47-2a27f6553c4e",
      "createdBy": "0"
    },
    {
      "id": "31Hj6G4BFXSpOOjivRdV",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemiil.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T04:21:09Z",
      "requestId": "cad26f91-8f13-4509-8a30-48c0e7462339",
      "createdBy": "0"
    },
    {
      "id": "2jyk6G4BFXSpOOji5MAg",
      "eventDescription": "User provided incorrect password.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemiil.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T03:12:30Z",
      "requestId": "b20083fb-a6d5-43ac-af50-944e4aea6fd9",
      "createdBy": "0"
    },
    {
      "id": "Wjyk6G4BFXSpOOjiu6z9",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemiil.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T03:12:20Z",
      "requestId": "a936ac6a-4962-40fd-92b5-2f03c2df66c4",
      "createdBy": "0"
    },
    {
      "id": "ezyk6G4BFXSpOOjilaFv",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemail.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T03:12:10Z",
      "requestId": "6f520201-6a6a-4d24-8fbc-82ea5e5a6fea",
      "createdBy": "0"
    },
    {
      "id": "JDyk6G4BFXSpOOjihZ-C",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemiil.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T03:12:06Z",
      "requestId": "61bb3ef8-2a06-4fab-adaf-172a78ca99a5",
      "createdBy": "0"
    },
    {
      "id": "7jyk6G4BFXSpOOjieJnK",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemiil.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T03:12:03Z",
      "requestId": "04d5b586-cc5b-4d3b-a78b-aaf364c1ceb4",
      "createdBy": "0"
    },
    {
      "id": "ETyk6G4BFXSpOOjiaJjt",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "12.xxx.xx.x",
      "userName": "joe.typical@myemiil.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-09T03:11:58Z",
      "requestId": "ebeb01de-1f81-4a7c-8978-405806e146bd",
      "createdBy": "0"
    }
  ]
}
```

Response headers:

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-encoding: gzip
content-length: 739
content-type: application/json
```

```
date: Mon, 09 Dec 2019 18:41:36 GMT
expires: 0
pragma: no-cache
status: 200
vary: Accept-Encoding, User-Agent
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability.

```
curl -X POST "https://product.supremomono.com/v1/audit/messages/list" -H "accep
t: application/json" -H "X-Authorization: eyJhbGciOiJSUzUxMiJ9.eyJzdWIiOiI1MiIs
ImNsaWVudFR5cGUiOiJXRUIiLCJsaWNlbnNlcyI6WyJERVZFTE9QTUVOVCJdLCJhbmFseXRpY3NMaWN
lbnNlc1B1cmNoYXNlZCI6eyJBbmFseXRpY3NDbGllbnQiOnRydWUsIkFuYWx5dGljc0FQSSI6dHJ1ZX
0sInRlbmFudFV1aWQiOiJhODc5MjE1Ny1jYjRmLTI3ZmItOTQ5Yy0wMzVmNDU1MThjNjEiLCJpYXQiO
jE1NzU5MTY1MTksImV4cCI6MTU3NTkxNzcxOSwiaXNzIjoiQXV0b21hdGlvbkFueXdoZXJlIiwibmFu
b1RpbWUiOjE0NDA0MTcwMDUxOTY1NzV9.IrmSOOzqE4pq09iOcLazsFoXkjKuw9eRVo9e6eaZWVIrYG
AcnJJ3zeeJrmi8HXUMsnCbYnLn-q2Y6HQT2rrQCoifnsQ9qrvoSBl-WUM1LXXc2Jw31r18i4J3yM1lY
wSNJ5-EvfS0pVf-tiDpdfHBWm9gGfaJzStKRx_TcGHaCN5_iCj3ZZbKGDGUqfEv7v4bkk_xwJCWJ2Tn
yY8gacKKtS3fBZb354OFJLoz8LYlnBt-e9Y3yus9aM6qIsGSrg9vwsu3b7wN7b44b-rpNmfWiwqN5N4
_UWVLvTblyNh8DOAd5B4uimFkPho3p1vY0so14TpfC59ztpkQS8lnqZbBWw" -H "Content-Type:
application/json" -d "{ \"sort\": [ { \"field\": \"createdOn\", \"direction\":
\"desc\" } ], \"filter\": { \"operator\": \"and\", \"operands\": [ { \"operator
\": \"gt\", \"field\": \"createdOn\", \"value\": \"2019-12-01T00:00:00.001Z\" }
, { \"operator\": \"lt\", \"field\": \"createdOn\", \"value\": \"2019-12-31T23:
59:59.999Z\" }, { \"operator\": \"eq\", \"field\": \"status\", \"value\": \"Uns
uccessful\" }, { \"operator\": \"substring\", \"field\": \"activityType\", \"va
lue\": \"LOGIN\" }, { \"operator\": \"substring\", \"field\": \"userName\", \"v
alue\": \"nafis.keshwani\" } ] }, \"fields\": [], \"page\": { \"length\": \"100
0\", \"offset\": \"0\" }}"
```

Related concepts
Example for createdOndate and userName filters for Audit API

Example for createdOndate and userName filters for Audit API

Create a filter that finds audit log entries for a specified date range for user with a specific string in their userName.

Request body

Finding the audit log entries you need is a formidable task. Use filtering to help narrow your results. The following example request identifies unsuccessful logins for users with the string "john,doe" in their userName from December 1, 2019 to December 7, 2019.

Example:

```
{
  "sort": [
    {
      "field": "createdOn",
      "direction": "desc"
    }
  ],
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "gt",
        "field": "createdOn",
        "value": "2019-12-01T00:00:00.001Z"
      },
      {
        "operator": "lt",
        "field": "createdOn",
        "value": "2019-12-31T23:59:59.999Z"
      },
      {
        "operator": "eq",
        "field": "status",
        "value": "Unsuccessful"
      },
      {
        "operator": "substring",
        "field": "activityType",
        "value": "LOGIN"
      },
      {
        "operator": "substring",
        "field": "userName",
        "value": "john,doe"
      }
    ]
  },
  "page": {
    "length": "1000",
    "offset": "0"
  }
}
```

Response body

This request identified 3 audit log entries out of 731,148,339 entries from this Enterprise Control Room's log entries.

```
{
  "page": {
    "offset": 0,
    "total": 731148339,
    "totalFilter": 3
  },
  "list": [
    {
      "id": "kLjB8G4BFXSpOOjiomK1",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "50.xxx.xxx.xx",
      "userName": "john,doe@mycompany.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-10T17:00:52Z",
      "requestId": "3c0f8e47-5820-43e8-b2b3-83b2f1cb86c9",
      "createdBy": "0"
    },
    {
      "id": "SLjB8G4BFXSpOOjikl5i",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "50.xxx.xxx.xx",
      "userName": "john,doe@mycompany.com",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-10T17:00:48Z",
      "requestId": "eba3e5a7-0034-440a-a786-110a84fea7c9",
      "createdBy": "0"
    },
    {
      "id": "7bjB8G4BFXSpOOjicEGO",
      "eventDescription": "User does not exist in Control Room.",
      "activityType": "LOGIN",
      "environmentName": "",
      "hostName": "50.xxx.xxx.xx",
      "userName": "john,doe",
      "status": "Unsuccessful",
      "source": "Control Room",
      "objectName": "N/A",
      "detail": "",
      "createdOn": "2019-12-10T17:00:39Z",
      "requestId": "64184450-aad5-4024-bcf5-491fb5276d0c",
      "createdBy": "0"
    }
  ]
}
```

Related concepts
Filters in an API request body

Related tasks
Audit API

Authentication API overview

Use the Authentication API to generate, refresh, and manage JSON Web Tokens (JWT) that are required for authorization in all Enterprise Control Room APIs.

You can view the Authentication API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.
The JWT is a text string with 703 characters.

```
{
  "token": "eyJhbGciOiJSUzUxMiJ9.eyJzdWIiOiIxIiwiY2xpZW50VHlwZSI6IldFQiIsImxpY2
Vuc2VzIjpbXSwiYW5hbHl0aWNzTGljZW5zZXNQdXJjaGFzZWQiOnsiQW5hbHl0aWNzQ2xpZW50Ijp0c
nVlLCJBbmFseXRpY3NBUEkiOnRydWV9LCJpYXQiOjE1NzMxMDc4NzMsImV4cCI6MTU3MzEwOTA3Mywi
aXNzIjoiQXV0b21hdGlvbkFueXdoZXJlIiwibmFub1RpbWUiOjM2NTc1NjI0OTQ2MzE2MDAsImNzcmZ
Ub2tlbiI6ImNiZjgwZWNkZmU5YmUwYzViOGI2MDk3NmU0ZTI2MTNiIn0.rGYxbS5kKUTxtZhYtRSXpm
IHwbf3IwLBIKDEA7odG5uGVAjD55Tv05bYdARx_3-tl1CBg_cDGbwj5FvaBt9u5xKu5W5j3Nur6x3PF
62NeB3ZIdxiUPaFBU0Br84mPJMD4_EpwBfbeSVOMH6ngiLtJYhIOtJa0kp4pAAm3mvkuOUELtH8lf3p
Qf-2Ose2fUAaebDkqiH13SUF1TONAjUQv6Ef_uY0wgq9SjZwKHg9SKUhX3S8PXAJne_ih2QnN8nUE1S
XGlkC04eoIvyWpFkM963XEjptc2uvwtVn42MdA4Nd1opD5yijEl9VM92Fe1sPb6_T5-oV-U1Iw0JHiX
2-Ug",  . . .
}
```

auth

POST `http://<your_control_room_url>/v1/authentication`

Body parameters:

```
{
  "username": "string",
  "password": "string",
  "apiKey": "string"
}
```

Make a post request to generate a JWT.

- The `username` of the Automation Anywhere user.
- The `password` of the Automation Anywhere user.

- The `apiKey` is required to configure Single Sign On (SSO). It can also be used in place of a password for users that are assigned to the API key generation role.
- A JWT is required in the header of other Enterprise Control Room APIs.
- Authentication tokens have a default timeout of 20 minutes.

Not all parameters are required to generate an authentication token. Go to the examples listed here for detailed information.

- Authenticate with username and password
- Authenticate with username and apiKey

Note:

Simple and Protected Negotiation GSSAPI Mechanism (SPNEGO)

You can use SPNEGO, pronounced "spenay-go," when your Enterprise Control Room is configured properly with the following authentication features:

- Active Directory (AD) mode of authentication
- AD is Kerberos enabled

In an Enterprise Control Room with SPNEGO properly configured, users do not need to enter a username and password to generate a JWT.

SPNEGO Authentication API URL example:`https://<your_control_room_url/v1/authentication/SPNEGO`

GET `http://<your_control_room_url>/v1/authentication/token/{token}`

URL parameter:
The token you are validating.
Note: The token is passed as a parameter in the URL. There are no parameters for the request body.

Read Validate an authentication token for task details.

POST `http://<your_control_room_url>/v1/authentication/token`

Body parameter:
A refresh token allows you to get a new token without the need to collect and authenticate credentials every time a token expires.

```
{
  "token": "string"
}
```

Click Refresh an authentication token for a detailed example of this API.

POST `http://<your_control_room_url>/v1/authentication/logout`

Header parameter:
Immediately expires the token that you add to the header of the request.

```
POST 'http://<your_control_room_url>/v1/authentication/logout'
-H 'X-Authorization: <access_token>
```

Click Immediately logout (expire) an authentication token for a detailed example of this API.

POST `http://<your_control_room_url>/v1/authentication/app/login`
The `.../atuhentticataion/app/login` API is a service to service authentication API used by Automation Anywhere internally supported applications. This API is not supported for use by external users.

Authenticate with username and password

Make a POST request with a username and password to generate a JSON Web Token (JWT) to use for authentication in Enterprise Control Room APIs.

Prerequisites

- Valid username and password for your Enterprise Control Room
- REST client or access to Automation Anywhere Swagger for your Enterprise Control Room.

Note: Passwords are 8-15 characters and contain the characters: a-z, A-Z, 0-9, at sign (@), dash (-), underscore (_), exclamation (!), pound (#), dollar ($), percent (%), ampersand (&), and period (.).

- URL: `http://<your_control_room_url>/v1/authentication`
- Method: POST

Procedure

1. Enter the following parameters in the request body.
   Request body:

   ```
   {
   "username": "jdoe",
   "password": "mypassword@123"
   }
   ```

   Depending on how your Enterprise Control Room is configured, a domain could be require with the username.

   ```
   {
   "username": "your-domain\\jdoe",
   "password": "mypassword@123"
   }
   ```

2. Send the request.
   - In a REST Client, click SEND.

- In the Swagger interface, click Execute.

Response body:

Note: The JWT is a 703 character string.

```
{
  "token": "eyJhbGciOiJSUzUxMiJ9.eyJzdWIiOiIxIiwiY2xpZW50VHlwZSI6IldFQiIsI
mxpY2Vuc2VzIjpbXSwiYW5hbHl0aWNzTGljZW5zZXNQdXJjaGFzZWQiOnsiQW5hbHl0aWNzQ2x
pZW50Ijp0cnVlLCJBbmFseXRpY3NBUEkiOnRydWV9LCJpYXQiOjE1NzMxMDc4NzMsImV4cCI6M
TU3MzEwOTA3MywiaXNzIjoiQXV0b21hdGlvbkFueXdoZXJlIiwibmFub1RpbWUiOjM2NTc1NjI
0OTQ2MzE2MDAsImNzcmZUb2tlbiI6ImNiZjgwZWNkZmU5YmUwYzViOGI2MDk3NmU0ZTI2MTNiI
n0.rGYxbS5kKUTxtZhYtRSXpmIHwbf3IwLBIKDEA7odG5uGVAjD55Tv05bYdARx_3-tl1CBg_c
DGbwj5FvaBt9u5xKu5W5j3Nur6x3PF62NeB3ZIdxiUPaFBU0Br84mPJMD4_EpwBfbeSVOMH6ng
iLtJYhIOtJa0kp4pAAm3mvkuOUELtH8lf3pQf-2Ose2fUAaebDkqiH13SUF1TONAjUQv6Ef_uY
0wgq9SjZwKHg9SKUhX3S8PXAJne_ih2QnN8nUE1SXGlkC04eoIvyWpFkM963XEjptc2uvwtVn4
2MdA4Nd1opD5yijEl9VM92Fe1sPb6_T5-oV-U1Iw0JHiX2-Ug",
  "user": {
    "id": 9,
    "email": "a@a.com",
    "username": "jdoe",
    "domain": null,
    "firstName": "",
    "lastName": "",
    "version": 9,
    "principalId": 9,
    "deleted": false,
    "roles": [
      {
        "name": "API_Key_Generation",
        "id": 23,
        "version": 0
      },
      {
        "name": "AAE_Basic",
        "id": 2,
        "version": 0
      },
      {
        "name": "Docrole1",
        "id": 18,
        "version": 0
      },
      {
        "name": "AAE_Meta Bot Designer",
        "id": 13,
        "version": 0
      }
    ],
    "sysAssignedRoles": [],
    "groupNames": [],
    "permissions": [

        . . .

    ],
    "licenseFeatures": [
      "RUNTIME"
    ],
    "emailVerified": true,
    "passwordSet": true,
    "questionsSet": true,
    "enableAutoLogin": false,
    "disabled": false,
    "clientRegistered": false,
    "description": "",
    "createdBy": 1,
    "createdOn": "2019-10-10T13:39:56-05:00",
    "updatedBy": 1,
    "updatedOn": "2019-10-13T02:09:38-05:00",
    "publicKey": null,
    "appType": null,
    "routingName": null,
    "appUrl": null
    }
}
```

Related concepts
Authentication API overview

Authenticate with username and apiKey

Make a POST request with a username and API to generate a JSON Web Token (JWT) to use to authenticate in Enterprise Control Room APIs.

Prerequisites

- A user with the Generate API-Key role
  Note: The Generate API-Key feature requires the creation of a custom role, see Create and assign API key generation role.
- Generate API-Key
- Valid username and apiKey for your Enterprise Control Room
- REST client or access to Automation Anywhere Swagger files for your Enterprise Control Room.

- URL: `http://<your_control_room_url>/v1/authentication`
- Method: POST

Using a Generate API-Key enables users to create tokens without the need to gather user credentials.

Procedure

1. Enter the following parameters in the request body.
   Request body:
   Note: The API-Key is a 40 character string.

```
{
"username": "jdoe",
"apiKey":  "3/.Z)8:P`+yVJq . . . *fTk.i>|VOOl&:"
}
```

Depending on how your Enterprise Control Room is configured, a domain could be require with the username.

```
{
"username": "your-domain\\jdoe",
"apiKey":  "3/.Z)8:P`+yVJq . . . *fTk.i>|VOOl&:"
}
```

Note: The API-Key Duration API-Key Duration can be configured by an Admin user from the ADMINISTRATION > Settings > General tab.

2. Send the request.
    - In a REST Client, click SEND.
    - In the Swagger interface, click .

   Response body:

   Note:

   Note: The JWT is a 703 character string.

```
{
  "token": "eyJhbGciOiJSUzUxMiJ9.eyJzdWIiOiIxIiwiY2xpZW50VHlwZSI6IldFQiIsI
mxpY2Vuc2VzIjpbXSwiYW5hbHl0aWNzTGljZW5zZXNQdXJjaGFzZWQiOnsiQW5hbHl0aWNzQ2x
pZW50Ijp0cnVlLCJBbmFseXRpY3NBUEkiOnRydWV9LCJpYXQiOjE1NzMxMDc4NzMsImV4cCI6M
TU3MzEwOTA3MywiaXNzIjoiQXV0b21hdGlvbkFueXdoZXJlIiwibmFub1RpbWUiOjM2NTc1NjI
0OTQ2MzE2MDAsImNzcmZUb2tlbiI6ImNiZjgwZWNkZmU5YmUwYzViOGI2MDk3NmU0ZTI2MTNiI
n0.rGYxbS5kKUTxtZhYtRSXpmIHwbf3IwLBIKDEA7odG5uGVAjD55Tv05bYdARx_3-tl1CBg_c
DGbwj5FvaBt9u5xKu5W5j3Nur6x3PF62NeB3ZIdxiUPaFBU0Br84mPJMD4_EpwBfbeSVOMH6ng
iLtJYhIOtJa0kp4pAAm3mvkuOUELtH8lf3pQf-2Ose2fUAaebDkqiH13SUF1TONAjUQv6Ef_uY
0wgq9SjZwKHg9SKUhX3S8PXAJne_ih2QnN8nUE1SXGlkC04eoIvyWpFkM963XEjptc2uvwtVn4
2MdA4Nd1opD5yijEl9VM92Fe1sPb6_T5-oV-U1Iw0JHiX2-Ug",
  "user": {
    "id": 9,
    "email": "a@a.com",
    "username": "jdoe",
    "domain": null,
    "firstName": "",
    "lastName": "",
    "version": 9,
    "principalId": 9,
    "deleted": false,
    "roles": [
      {
        "name": "API_Key_Generation",
        "id": 23,
        "version": 0
      },
      {
        "name": "AAE_Basic",
        "id": 2,
        "version": 0
      },
      {
        "name": "Docrole1",
        "id": 18,
        "version": 0
      },
      {
        "name": "AAE_Meta Bot Designer",
        "id": 13,
        "version": 0
      }
    ],
    "sysAssignedRoles": [],
    "groupNames": [],
    "permissions": [

        . . .

    ],
    "licenseFeatures": [
      "RUNTIME"
    ],
    "emailVerified": true,
    "passwordSet": true,
    "questionsSet": true,
    "enableAutoLogin": false,
    "disabled": false,
    "clientRegistered": false,
    "description": "",
    "createdBy": 1,
    "createdOn": "2019-10-10T13:39:56-05:00",
    "updatedBy": 1,
    "updatedOn": "2019-10-13T02:09:38-05:00",
    "publicKey": null,
    "appType": null,
    "routingName": null,
    "appUrl": null
  }
}
```

Related concepts
Authentication API overview
Related tasks
Authenticate with username and password

Refresh an authentication token

Refresh valid authenticatoin tokens without the need to collect user credentials.

Prerequisites

- The token you are refreshing.
- REST client or access to Automation Anywhere Swagger files for your Enterprise Control Room.

- URL: `http://<your_control_room_url>/v1/authentication/token`
- Method: POST

Procedure

1. Enter the following parameters in the request body.
   Request body:

```
{
    "token": "eyJhbGciOiJSUzUxMiJ9.eyJzdWIiOiIxIiwiY2xpZW50VHlwZSI6IldFQiI
sImxpY2Vuc2VzIjpbXSwiYW5hbHl0aWNzTGljZW5zZXNQdXJjaGFzZWQiOnsiQW5hbHl0aWNzQ
2xpZW50Ijp0cnVlLCJBbmFseXRpY3NBUEkiOnRydWV9LCJpYXQiOjE1NzMxMDgwNjEsImV4cCI
6MTU3MzEwOTI2MSwiaXNzIjoiQXV0b21hdGlvbkFueXdoZXJlIiwibmFub1RpbWUiOjM2NTc3N
TA4OTY5NzUxMDAsImNzcmZUb2tlbiI6ImJiNjgzMGJhMDY5MWYwYjZiM2M3MDE4NGY0OGM0MWY
1In0.f3kPRspfm0sei9DGHd9NoyLK-iCO-vs--8b_pLG9XSUR0186uvXFopB75eVAaG-1l_AZh
R78UE6Voi7_UggzHkLRrEpQ-szR7cmFDpLxZ28xLnFJYhaIuMNdw9dWDVquBWTQSpYGNJd56D-
tFFHBodwVdNamqWHxaQebq1zMyUyQV6Q-gKdgubpT5gwuXnp-BwScjHOYM3Fpj_nt0nEbJC5uW
pJNtLQBpVzhsRwwlRKNOHQVbo6X7zkvKBoij8ewa5FWQwX7T-760BeqfssR6mmMUo0zRaneUKA
YAskz0B-X5PcyCkrVJju2XqItQ9XMGNP7h_MaUDotU_CJyguPZA"
}
```

2. Send the request.
   - In a REST Client, click SEND.
   - In the Swagger interface, click Execute.

   Response body: A new token with an updated experation time.

```
{
  "token": "eyJhbGciOiJSUzUxMiJ9.eyJzdWIiOiIxIiwiY2xpZW50VHlwZSI6IldFQiIsI
mxpY2Vuc2VzIjpbXSwiYW5hbHl0aWNzTGljZW5zZXNQdXJjaGFzZWQiOnsiQW5hbHl0aWNzQ2x
pZW50Ijp0cnVlLCJBbmFseXRpY3NBUEkiOnRydWV9LCJpYXQiOjE1NzMxMTMxMTgsImV4cCI6M
TU3MzExNDMxOCwiaXNzIjoiQXV0b21hdGlvbkFueXdoZXJlIiwibmFub1RpbWUiOjM2NjI4MDc
1NDY4NjYzMDAsImNzcmZUb2tlbiI6ImUxODBiZjYxMzQyMjkwZTRlM2Q4M2ZlNTU3YWRmMmE5I
n0.mtR1RNDe3EPzlaLQ7mwF0yIk8G00wLKGmKTFhM2689rItXHjLLgv0iYaM1LPUtRv9GafjhX
fshcIm9lucyf8k8t3A7SVJFoiFY2TUNgeouPgHl7XlpzpmenDRoT4Otu9R1_FTpMi40HH81ARG
7WoLDPdOyhxgL9ZvoVtRgkMiNTn1vUJWGhzd6wMYzf70rJO_TcMKgLh3X6fpPkY_xD2ykrKsds
MO2lZnzDjzuf3BCdzGjMj1q99WKBgVwyMafv4WApUX5peRZlsiVJdZrM2x890yovW2Yy_fY3wd
P_57XRp1oA5vnm9FxJN9lKyxVic3Qvx8BGtXmR-GQ3T8fndjw",
  "user": {
    "id": 1,
    "email": "a@a.com",
    "username": "admin",
    "domain": null,
    "firstName": "",
    "lastName": "",
    "version": 1,
    "principalId": 1,
    "deleted": false,
    "roles": [
      {
        "name": "AAE_Admin",
        "id": 1,
        "version": 0
      }
    ],
    . . .
    ],
    "licenseFeatures": [],
    "emailVerified": true,
    "passwordSet": true,
    "questionsSet": true,
    "enableAutoLogin": false,
    "disabled": false,
    "clientRegistered": false,
    "description": "",
    "createdBy": 0,
    "createdOn": "2019-09-25T16:03:05-05:00",
    "updatedBy": 0,
    "updatedOn": "2019-09-25T16:03:05-05:00",
    "publicKey": null,
    "appType": null,
    "routingName": null,
    "appUrl": null
  }
}
```

Related concepts
Authentication API overview

Validate an authentication token

Send a REST request to verify if a token is valid.

Prerequisites

- The token you are validating.
- REST client or access to Automation Anywhere Swagger files for your Enterprise Control Room.

- URL: `http://<your_control_room_url>/v1/authentication/token`
- Method: GET

Procedure

1. Enter the following parameters to the request URL.
   `http://<your_control_room_url>/v1/authentication/token?token=<token>`
2. Send the request.
   - In a REST Client, click SEND.
   - In the Swagger interface, click Execute.

   Response body:
   The token is valid.

   ```
   {
     "valid": true
   }
   ```

   The token is invalid.

```
{
  "valid": false
}
```

Related concepts
Authentication API overview

Create and assign API key generation role

API key generation is available in Enterprise Control Room v11.3.2 and later. By default this parameter is not enabled for any of the System-created roles. An administrator is able to create a custom role for API key generation and assign that custom role to users.

This task describes how an administrator can create and assign a custom role for API key generation.

Procedure

1. Log in as an administrator to the Enterprise Control Room.
2. Go to Administration > Roles.
3. Click Create role . . ..
4. Scroll to the API section.
5. Select Generate API-Key.
6. Type a unique name in the Role name field.
7. Click Create role.
8. Go to Administration > Users, and assign the custom role you just created to non-admin users.
9. Log in as the user you assigned the Generate API-Key role to.
10. Under the user name, click Generate API-Key, and copy the API-Key to your clipboard.

Next steps

Use the API-Key to log into an Enterprise Control Rooms using SSO, or use the API-Key to log in a user without the user's password.
Related tasks
Authenticate with username and apiKey

Immediately logout (expire) an authentication token

Immediately invalidate an access token so that it cannot be used for authentication.

Prerequisites

- The URL for the Enterprise Control Room in which the token was generated
- The token to expire

Procedure

1. Enter the following parameters to the request URL.

```
https://<your_control_room_url>/v1/authentication/logout
```

2. In a header for this request, enter the token that is to be expired.
   Note: There are no body parameters in this request.
3. Send the request.
   - In a REST Client, click SEND.
   - In the Swagger interface, click Execute.

   Response header:
   Note: A 204 response code indicates that the request was successful and that there is no additional content to be sent to the response body.

   ```
   Status Code: 204 No Content
   cache-control: no-cache, no-store, max-age=0, must-revalidate
   content-security-policy: default-src 'self'
   content-type: application/json
   date: Thu, 31 Oct 2019 08:37:35 GMT
   expires: 0
   pragma: no-cache
   x-content-type-options: nosniff
   x-frame-options: SAMEORIGIN
   x-xss-protection: 1; mode=block
   ```

Related tasks
Authenticate with username and apiKey

Bot Deploy API

The Bot Deploy API supports the runAsUser feature for bot deployment.

You can view the Bot Deploy API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.

- The Bot Deploy API works only in Enterprise Control Room public accounts, and not in Community Edition.
- The API supports callback URL for environments that have both the Enterprise Control Room and the callback server on the same network.

- Bot deploy task
  As a registered Bot Runner, deploy bots on registered devices that are part of a device pool. You can also pass variables to bots when they are deployed.

Related concepts
User management API overview
Related tasks
How to find a user's id
How to find a device pool id

How to find a bot id

Bot deploy task

As a registered Bot Runner, deploy bots on registered devices that are part of a device pool. You can also pass variables to bots when they are deployed.

Prerequisites

- runAsUserIds: the user id of a registered Bot Runner that has an active Unattended bot runner device license with the RUNTIME feature license.
  An unattended Bot Runner needs these permissions to deploy bots:

  ACTIVITY

  - View my scheduled bots
    - Schedule my bots to run
    - Edit my scheduled activity

  BOTS

  - View my bots
    - Run my bots

  Note: The user needs to have a device licenses for an Unattended bot runner.
- poolIds: the numeric id of a device pool that is linked with at least one registered device.
- The variable type and name used in the bot.

Deploy a bot and pass a variable value to the bot when it runs. .

- fileId: the numeric identifier for the bot to be deployed. How to find a bot id
- runAsUserIds: the numeric identifier for a user that is registered with your Enterprise Control Room as an Unattended bot runner. How to find a user's id
- poolIds: the numeric identifier of a device pool that has at least one active device. How to find a device pool id
- botInput: the variable used by the bot. You can pass variable values at the time the bot is deployed. In this request example, the bot uses the variable sDocHello as input and output. The string field is the value passed from the API to the bot

Procedure

1. Add an authentication token to the request header. Use the Authentication API to generate a JSON Web Token.
2. POST is the method used for this API.
3. Enter the request in valid JSON format.

```
{
  "fileId": 57911,
  "runAsUserIds": [
    2538
  ],
  "poolIds": [
    49
  ],
  "overrideDefaultDevice": false,
  "botInput": {
    "sDocHello": {
      "type": "STRING",
      "string": "Hello world, go be great."
    }
  }
}
```

4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND

   Response body:
   When successful, the Bot Deploy API returns a response with the deploymentId.

```
{
  "deploymentId": "14c2b6f8-c2a0-4a57-959d-ef413df0d179"
}
```

Next steps

Use the Bot Execution Orchestrator API to list the activity for a specific deploymentID.
Request URL:

```
<your_control_room_url>/v2/activity/list
```

Request body:

```
{
  "filter": {
    "operator": "eq",
    "field": "deploymentId",
    "value": "14c2b6f8-c2a0-4a57-959d-ef413df0d179"
  }
}
```

Response body:

```
{
    "page": {
        "offset": 0,
        "total": 13,
        "totalFilter": 1
    },
    "list": [
        {
            "id": "e49cea73-01fb-4a10-a7de-ce3e7b83a5ae_0e6c0971292ea185",
            "automationName": "deploy-test_20.05.01.17.08.35_jdoe_API",
            "fileName": "deploy-test",
            "filePath": "Automation Anywhere\\Bots\\deploytest\\deploy-test",
            "type": "TASK",
            "startDateTime": "2020-05-01T17:09:02Z",
            "endDateTime": "2020-05-01T17:09:05Z",
            "command": "",
            "status": "COMPLETED",
            "progress": 100,
            "automationId": "22260",
            "userId": "2538",
            "deviceId": "894",
            "currentLine": 1,
            "totalLines": 1,
            "fileId": "57911",
            "modifiedBy": "2540",
            "createdBy": "2538",
            "modifiedOn": "2020-05-01T17:08:47.657801Z",
            "createdOn": "2020-05-01T17:08:38.228573Z",
            "deploymentId": "14c2b6f8-c2a0-4a57-959d-ef413df0d179",
            "queueName": "",
            "queueId": "",
            "usingRdp": false,
            "message": "",
            "canManage": false,
            "deviceName": "MyCompany-JohnDoe",
            "userName": "jdoe"
        }
    ]
}
```

Related tasks
How to find a user's id
How to find a device pool id
How to find a bot id
Search for users
Create a new user
Create a new role

How to find a user's id

The numeric identifier, {id}, is used by APIs to uniquely identify users in the Enterprise Control Room.

Prerequisites

- The username of the user to search for.
- Minimally you need the administrative permission to View users.

In order to deploy bots using the run as user feature, you need the user id of an unattended Bot Runner. Use the User management API to find users, or follow these steps from your Enterprise Control Room UI.

This task shows how to find a user's unique numeric identifier from the Enterprise Control Room user interface.

Note: Administrators can create users and roles from the Administration > Roles > All roles > Edit role page. Users should be given only the permissions needed for the required task.

Procedure

1. Log on as a user that has View users permission.
2. Go to the Administration > Users page, and search for the user you need to find.
3. In the USERNAME column, click the name of the user to see the id in the URL.
4. Look in the Enterprise Control Room URL to see the users id.

   ```
   <your_control_room_url>/#/admin/users/2538/edit
   ```

   In this example, the user id is 2538.

Related tasks
Bot deploy task
How to find a device pool id
How to find a bot id

How to find a device pool id

How to find a device pool id from the Enterprise Control Room user interface.

Prerequisites

- The username of the user to search for.
- Minimally you need the administrative permission to view View users.

A device pool id is required when using the Bot deploy API to manage bot deployments.
Note: Administrators can create users and roles from the Administration > Roles > All roles > Edit role page. Users should be given only the permissions needed for the required task.

Procedure

1. Log on as a user that has permissions to View and manage ALL device(s) and Create device pools.
   Note: Because you are logged on to the Enterprise Control Room as a user that is assigned to a device pool, you are able to see device pools in My device pools.
2. Go to DEVICES > My device pools.
3. In the DEVICE POOL NAME column, click the device pool you want to use.
4. Look in the Enterprise Control Room URL to see the device pool id.

   ```
   <your_control_room_url>/#/devices/mydevicepools/49/edit
   ```

   In this example, the device pool id is 49.

Related tasks
Bot deploy task
How to find a user's id
How to find a bot id

How to find a bot id

The fileId used in the Bot Deploy API is the numeric identifier for the bot. Each bot has a unique numeric identifier.

Prerequisites

- Login credentials for a Bot Runner user that has permission to view the bots you want to deploy.
- The path to the folder that contains the bot you want to find.

This task shows you how to find the bot id by navigating the Enterprise Control Room user interface.

You can also use the Repository Management API to search a folder in the Enterprise Control Room repository for a list of bots and their ids.

Procedure

1. Log on as a Bot Runner user that has permissions to view the bots you want to find.
2. Go to BOTS > My Bots > Bots > <your folder>.
3. In the NAME column, click the name of the bot you want to deploy.
4. Look in the Enterprise Control Room URL to see the bot id.

```
<your_control_room_url>/#/bots/repository/public/taskbots/57911/view
```

In this example, the bot id is 57911.

Related tasks
Bot deploy task
How to find a user's id
How to find a device pool id

Bot Execution Orchestrator API

As an Enterprise Control Room administrator or a user with View and Manage Scheduled Activity permission, you can monitor bot progress using a set of Enterprise Control Room APIs.

You can view the Bot Execution Orchestrator API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.

High-level process for monitoring bots

Searchable fields for devices:

- hostName: The host name of the device configured as a Bot Runner. If a naming convention is used for host names, searching on a unique substring in the host name is an effective way to identifies Bot Runner devices.
- userId: The unique numeric identification for a specific user also identifies the Bot Runner device. Unique user naming conventions can be used to identify users and devices that are licensed and configured as Bot Runners.

Searchable fields for bots:

- name: The unique name of a bot. You can search on the exact name (eq) or a text string (substring) that is contained in the bots name.
- path: The relative path of a folder in the Enterprise Control Room. You can search on a full path or a substring contained in the path.

Bot Insight API

Get bot process data for analytic analysis. Only users with Bot Insight administration role can access this API.

You can view the BotInsight API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.

Automation Anywhere bots are built, run, and monitored in the Enterprise Control Room. Bot Insight accesses real-time business insights and digital workforce performance data to leverage content-level productivity data from the bots that are deployed.

Bot Lifecycle Management API overview

Use the Bot Lifecycle Management API to export and import bots with dependent files and action packages for comprehensive automation life-cycle management.

Bots can be exported from public folders and imported to a private folder in another Enterprise Control Room and checked into a public folder. Users must have the following permissions and licenses.

- Export bots
- Import bots
- Upload permission to the necessary folders
- Import permission to the necessary folders
- Bot Creator license

Dependent files and actions are automatically included.

Credential Vault API overview

As an Enterprise Control Room user with Manage my credentials and lockers role permissions, you have the option to use the Credential Vault API to manage your attributes, credentials, keys, lockers, and Credential Vault mode in the Enterprise Control Room.

You can view the Credential Vault API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.

By default, all users can create credentials. You are the Credential owner of any credentials that you created. As a Credential owner, you can update, delete, and transfer the ownership of your credentials.

Migration APIs

The Migration APIs allow users with the appropriate migration permission to view or manage bot migration from 11.x to A2019, including starting the migration of bots and retrieving details about migrations.

There are several prerequisites for migration to A2019 that must be completed before 11.x bots can be migrated to A2019. See Prerequisite tasks for migrating bots

- Start migration
  The Migration API enables users to convert and migrate bots (TaskBots and MetaBots) created using the Enterprise client version 11.x to A2019.
- Migration results list
  List the overall migration results for each migration you run. Filter by selected fields to get the specific results you need.

- Migration status results by id
  List bot migration results by a unique numeric identifier, {id}, and filter the results by selected fields.
- Migration action mapping results
  List action mapping results for bots by unique numeric identifiers for the migration {id} and the journal {journalid}, and filter the results by selected fields.

Start migration

The Migration API enables users to convert and migrate bots (TaskBots and MetaBots) created using the Enterprise client version 11.x to A2019.

Prerequisites

User with a runtime license and the following permissions can start a migration.

BOTS
    Run my bot
    Export bots
    Import bots
    Create bots
    Rename folders
DEVICES
    Register device
    View and Manage ALL device(s)
    Delete the devices(s)
AUDIT LOG
    View everyone's audit log actions
ADMINISTRATION
    Allow a bot-runner user to run migrations

- The folderIds that you want to migrate.
- The userIds to run as for the migration.

```
{
  "id": 0,
  "name": "string",
  "description": "string",
  "overwtiteBots": true,
  "botIds": [
    0
  ],
  "userIds": [
    0
  ],
  "folderIds": [
    0
  ],
  "includeChildFolders": true
}
```

Procedure

1. Add an authentication token to the request header. Use the Authentication API to generate a JSON Web Token.
2. POST is the method used for this API.
3. This request is structured to migrate a single bot, botIds 74, and run as user 18.

```
{
  "name": "Docs Test Migration",
  "description": "docs test",
  "overwtiteBots": true,
  "botIds": [
    74
  ],
  "userIds": [
    18
  ]
}
```

4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND

   The successful response includes a 200 success code and an empty body.

```
{ }
```

You can also migrate all the bots from a sub-folder.

Migrate all bots in a sub-folder
    Migrate all the bots contained in a sub-folder in your Enterprise Control Room repository.

Next steps

You can view the status of the migration using the Migration results list API.

Migrate all bots in a sub-folder

Migrate all the bots contained in a sub-folder in your Enterprise Control Room repository.

Prerequisites

Find the folder {id} you want to migrate

> List workspaces folders and files searches for files and folders in the private or public Enterprise Control Room repositories. Filter the results to identify the folder ids to be used in the migration request body.

userIds for one or more user with a RUNTIME device license

> Use userIds for registered users in the Enterprise Control Room as unattended bot runners with a RUNTIME device license and registered device. Search for users

Authentication token for a user with migration permission

> Request an authentication token using the login for a user that has the administrative permission to view and mange migrations. Authentication API overview

Procedure

1. Add an authentication token to the request header. Use the Authentication API to generate a JSON Web Token.
2. POST is the method used for this API.
3. This request starts a migration for all the bots contained in the sub-folder with the folderIds equal to 7. The user is an unattended Bot Runner with the userIds equal to 18.

```
{
  "name": "Follow a convention that is meaningful and easy to search.",
  "description": "Add a meaningful description.",
  "overwtiteBots": true,
  "userIds": [
    18
  ],
  "folderIds": [
    7
  ],
  "includeChildFolders": true
}
```

4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND

   The successful response includes a 200 success code and an empty body.

```
{ }
```

Next steps

You can view the status of the migration using the Migration results list API.

Migration results list

List the overall migration results for each migration you run. Filter by selected fields to get the specific results you need.

Prerequisites

- Administrator or user with View migration permission is required to view details about migrations.
- Access to the Automation Anywhere Migration API.

```
<your_control_room_url>/v3/migration/list
```

- Any values for parameters you want to use to filter and limit your search.

Supported filterable fields:

createdBy

The unique numeric identifier of the user who started the migration.

```
{
  "filter": {
    "field": "createdBy",
    "value": 6,
    "operator": "eq"
  }
}
```

numTotal

The total number of bots migrated in a specific migration, including bots successfully migrated, skipped, and failed.

```
{
  "filter": {
    "field": "numTotal",
    "value": 0,
    "operator": "gt"
  }
}
```

Filter on the numeric values of these similar fields.

numFailed

The number of bots that failed to be migrated in a specific migration.

numSkipped

Skipped bots include those bots that already exist, and the user has chosen not to overwrite existing bots (`"overwriteBots": false`).

numSuccess

The number of bots successfully migrated in a specific migration.

updatedBy

The numeric identifier of the user who started the migration.

```
{
  "filter": {
    "field": "updatedBy",
    "value": 6,
    "operator": "eq"
  }
}
```

updatedOn

The date and time when the migration was started.

```
{
  "filter": {
    "field": "updatedOn",
    "value": "2020-04-07T00:42:08.967Z",
    "operator": "eq"
  }
}
```

Procedure

1. Add an authentication token to the request header. Use the Authentication API to generate a JSON Web Token.
2. POST is the method used for this API.
3. Enter a request body. This request searches for migrations that contain the string doc in the name that was started between the specified dates.

```
{
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "gt",
        "field": "updatedOn",
        "value": "2020-04-08T00:00:00.001Z"
      },
      {
        "operator": "substring",
        "field": "name",
        "value": "doc"
      },
      {
        "operator": "lt",
        "field": "updatedOn",
        "value": "2020-04-13T00:00:00.001Z"
      }
    ]
  }
}
```

4. Send the request.
    - In Swagger, click Execute.
    - In a REST client, click SEND

    The response for this request returned 3 of 34 migrations.

    Response body:
    Note: There are some response fields that are not used for 11.x migration:
    - duration: is a legacy field that is no longer used for migration.
    - migrationType: is used for 10.x migrations only. It is not used for 11.x migration.

```
{
    "page": {
        "offset": 0,
        "total": 34,
        "totalFilter": 3
    },
    "list": [
        {
            "id": 21,
            "name": "Docs Test Migration",
            "startTime": "2020-04-09T21:09:25.590Z",
            "endTime": "2020-04-09T22:41:49.313Z",
            "createdBy": 17,
            "duration": "5543s",
            "numSuccess": 0,
            "numFailed": 0,
            "numSkipped": 0,
            "numTotal": 1,
            "status": "IN_PROGRESS",
            "updatedOn": "2020-04-09T21:09:25.590Z",
            "updatedBy": 17,
            "durationMillis": 5543723,
            "migrationType": "BOT"
        },
        {
            "id": 22,
            "name": "Docs Test Migration 02",
            "startTime": "2020-04-09T21:22:32.587Z",
            "endTime": "2020-04-09T22:41:49.313Z",
            "createdBy": 17,
            "duration": "4756s",
            "numSuccess": 0,
            "numFailed": 0,
            "numSkipped": 0,
            "numTotal": 1,
            "status": "IN_PROGRESS",
            "updatedOn": "2020-04-09T21:22:32.587Z",
            "updatedBy": 17,
            "durationMillis": 4756726,
            "migrationType": "BOT"
        },
        {
            "id": 24,
            "name": "Docs Test 03",
            "startTime": "2020-04-09T22:31:27.617Z",
            "endTime": "2020-04-09T22:41:49.317Z",
            "createdBy": 17,
            "duration": "621s",
            "numSuccess": 211,
            "numFailed": 0,
            "numSkipped": 0,
            "numTotal": 211,
            "status": "SUCCESSFUL",
            "updatedOn": "2020-04-09T22:31:27.617Z",
            "updatedBy": 17,
            "durationMillis": 621700,
            "migrationType": "BOT"
        }
    ]
}
```

Next steps

To view details about a specific migrattion, enter a specific migration id in the Migration status results by id API.

Migration status results by id

List bot migration results by a unique numeric identifier, {id}, and filter the results by selected fields.

Prerequisites

- Administrator View migration permission to be able to view details about a migration.
- Access to the Automation Anywhere Migration API.

```
<your_control_room_url>/v3/migration/{id}/results/list
```

- Any values for parameters you want to use to filter and limit your search.
- The numeric identifier, {id}, for the migration you want to view.

Supported filterable fields:

reason

```
{
  "filter": {
    "field": "reason",
    "operator": "substring",
    "value": "failed"
  }
}
```

sourceName

```
{
  "filter": {
    "field": "sourceName",
    "operator": "substring",
    "value": "Box01"
  }
}
```

sourcePath

```
{
  "filter": {
    "field": "sourcePath",
    "operator": "substring",
    "value": "My Metabots"
  }
}
```

status

```
{
  "filter": {
    "field": "status",
    "operator": "eq",
    "value": "FAILED"
  }
}
```

- FAILED
- SUCCESS
- SKIPPED

targetName

```
{
  "filter": {
    "field": "targetName",
    "operator": "substring",
    "value": "dep01"
  }
}
```

targetPath

```
{
  "filter": {
    "field": "targetPath",
    "operator": "substring",
    "value": "dep01"
  }
}
```

Procedure

1. Add an authentication token to the request header. Use the Authentication API to generate a JSON Web Token.
2. POST is the method used for this API.
3. Enter the numeric identifier for the migration you want to view the details about. In this example we are viewing the migration with the id of 32.

   ```
   <your_control_room_url>/v3/migration/32/results/list
   ```

4. Create a request to find the results you want to see. This filter searches for a string in the name of the migrated bot.

   ```
   {
     "filter": {
       "operator": "substring",
       "field": "targetName",
       "value": "logic-launch"
     }
   }
   ```

5. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND

   The response for this request returned 1 out of 3 responses for bot migration details.

   Response body:

```
{
    "page": {
        "offset": 0,
        "total": 3,
        "totalFilter": 1
    },
    "list": [
        {
            "sourceId": 24,
            "sourceName": "mbot-dep01.mbot",
            "sourcePath": "Automation Anywhere\\Bots\\My MetaBots\\mbot-de
p01.mbot",
            "sourceType": "application/vnd.aa.mbot",
            "targetId": 941,
            "status": "SUCCESS",
            "reason": "",
            "selectedByUser": true,
            "userId": 9,
            "id": 469,
            "targetName": "logic-launchweb01",
            "targetPath": "Automation Anywhere\\Bots\\My MetaBots\\mbot-de
p01\\logic-launchweb01",
            "targetType": "application/vnd.aa.taskbot"
        }
    ]
}
```

Next steps

Migration action mapping results

List action mapping results for bots by unique numeric identifiers for the migration {id} and the journal {journalid}, and filter the results by selected fields.

Prerequisites

- Administrator View migration permission to be able to view details about a migration.
- Access to the Automation Anywhere Migration API.

```
<your_control_room_url>/v3/migration/{id}/results/list
```

- Any values for parameters you want to use to filter and limit your search.
- The numeric identifier, {id}, for the migration you want to view.
- The numeric value for the {journalid} associated with the migration identifier.

```
/v3/migration/{id}/journal/{journalid}/actionmappings/list
```

Note: How to find a migration journalid

Supported filterable fields:

reason
Filter on a string within the reason field to return specific journal entries.

```
{
  "filter": {
    "field": "reason",
    "operator": "substring",
    "value": "not yet supported"
  }
}
```

remarks
Filter on a string within the remarks field to return specific journal entries.

```
{
  "filter": {
    "field": "remarks",
    "operator": "substring",
    "value": "not yet supported"
  }
}
```

sourceAction

Filter for specific actions used the bot being migrated.

```
{
  "filter": {
    "field": "sourceAction",
    "operator": "substring",
    "value": "OpenSpreadsheet"
  }
}
```

targetAction

Filter for specific actions used the bot target migrated bot.

```
{
  "filter": {
    "field": "targetAction",
    "operator": "substring",
    "value": "OpenSpreadsheet"
  }
}
```

Procedure

1. Add an authentication token to the request header. Use the Authentication API to generate a JSON Web Token.
2. Enter the migration id and journal id in the request URL>

```
/v3/migration/8/journal/8/actionmappings/list
```

3. POST is the method used for this API.
4. Request body.

```
{
  "filter": {
    "field": "reason",
    "operator": "substring",
    "value": "not yet supported"
  }
}
```

5. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND

   The response for this request returned 1 of 10 migration list objects results.

   Response body:

```
{
    "page": {
        "offset": 0,
        "total": 10,
        "totalFilter": 1
    },
    "list": [
        {
            "targetLineNumber": 1,
            "targetAction": "runTask",
            "isReviewRequired": true,
            "reason": " 1. The \"Run Task\" bot path defined with variable
s is not yet supported.",
            "remarks": "",
            "id": 6,
            "sourceLineNumber": 0,
            "sourceAction": ""
        }
    ]
}
```

- How to find a migration journalid
  Migrations can have more than one journalid. You can find the journalid in the response of a results list for a specific migration id.

How to find a migration journalid

Migrations can have more than one journalid. You can find the journalid in the response of a results list for a specific migration id.

In this request, we searched for results for the migration with the id 16.

```
<your_control_room_url>/v3/migration/16/results/list
```

The request returned this response. The id in each list object us the journalid. In this example, there are two journal identifiers, 30 and 31, for the migration with the id 16.

```
{
    "page": {
        "offset": 0,
        "total": 2,
        "totalFilter": 2
    },
    "list": [
        {
            "sourceId": 12,
            "sourceName": "Dependency of IGN-23437.mbot",
            "sourcePath": "Automation Anywhere\\Bots\\My MetaBots\\Dependency o
f IGN-23437.mbot",
            "sourceType": "application/vnd.aa.mbot",
            "targetId": 0,
            "status": "FAILED",
            "reason": "The logic IGN-23437 has some commands or actions which a
re not yet supported for migration.",
            "selectedByUser": true,
            "userId": 9,
            "id": 30,
            "targetName": "",
            "targetPath": "",
            "targetType": ""
        },
        {
            "sourceId": 12,
            "sourceName": "Dependency of IGN-23437.mbot",
            "sourcePath": "Automation Anywhere\\Bots\\My MetaBots\\Dependency o
f IGN-23437.mbot",
            "sourceType": "application/vnd.aa.mbot",
            "targetId": 0,
            "status": "FAILED",
            "reason": "Migration of MetaBot failed.",
            "selectedByUser": false,
            "userId": 9,
            "id": 31,
            "targetName": "",
            "targetPath": "",
            "targetType": ""
        }
    ]
}
```

This is how you would enter the migration id and journal id in an action mapping request.

```
<your_control_room_url>/v3/migration/16/journal/31/actionmappings/list
```

Repository Management API

The Repository management API is a role based access API that returns information for folders and files that you have permission to view in your Enterprise Control Room.

You can view the Repository Management API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.

The Repository management API searches an Enterprise Control Room repository using a folder ID associated with a user. The response returns the sub-folders contained in that folder. Folder access is based on the roles and licenses that are assigned to a user.

Bot developers have access to the folders where their bots are stored. An administrator, however, might have access to the folders in the root directories and its sub-folders.

Locate your folder IDs by logging in to your Enterprise Control Room and looking at the URL for the folder you want to search.

```
<your_control_room_url>/#/bots/repository/public/folders/7
```

In this example, 7 is the folder identification number for the sub-folder "My Tasks." The parent to this folder is named "Bots" with a numeric identifier of 923.

- List folders by id
  List and filter files under a specific parent folder by using the parent folder id.
- List workspaces folders and files
  Search in the Enterprise Control Room private or public repository for folders and files, and use filters to find exactly the files and folders you need.

List folders by id

List and filter files under a specific parent folder by using the parent folder id.

Prerequisites

- You need the numeric ID for the top level folder you want to search.
- An authentication token for a user registered in the Enterprise Control Room.
  Note: Users can only view the folders and sub-folders they have permissions to access.

In the examples used for this task, the top-level public folder identification number is 2. We are searching for sub-folders that contain the string "doc." The structure of this request limits the query to find sub-folders under the parent folder with the identifier of 2.

Procedure

1. Add an authentication token to the request header. Use the Authentication API to generate a JSON Web Token.
2. POST is the method used for this API.
3. The {folderid} is a numeric value. In the request URL, add a {folderid} for the folder you want to search. For this example we are using 2 for the {folderid}.

   ```
   https://<your_control_room_url/v2/repository/public/folders/2/list
   ```

   The following request searches for bots and folders that contain the string finance in the name parameter.

   Request body:

   ```
   {
     "filter": {
       "operator": "and",
       "operands": [
         {
           "operator": "substring",
           "field": "name",
           "value": "finance"
         },
         {
           "operator": "eq",
           "field": "folder",
           "value": "true"
         }
       ]
   ```

```
  }
}
```

4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND

   The response for this request returned 1 out of 329 folders. The request searched the name parameter for the string doc. The name parameter is the name of a sub-folder under the parent folder Bots.

   Response body:

```
{
  "page": {
    "offset": 0,
    "total": 329,
    "totalFilter": 1
  },
  "list": [
    {
      "id": "40378",
      "parentId": "2",
      "name": "TS-Docs",
      "path": "Automation Anywhere\\Bots\\TS-Docs",
      "description": "",
      "type": "application/vnd.aa.directory",
      "size": "0",
      "folder": true,
      "folderCount": "0",
      "productionVersion": "",
      "latestVersion": "",
      "locked": false,
      "lockedBy": "0",
      "createdBy": "2587",
      "lastModifiedBy": "2587",
      "lastModified": "2019-12-12T07:01:43.991Z",
      "workspaceId": "0",
      "botStatus": "DRAFT",
      "hasErrors": false,
      "workspaceType": "UNKNOWN",
      "metadata": false,
      "uri": "",
      "version": "0",
      "hasTriggers": false
    }
  ]
}
```

Next steps

The id "40378" from the response is the numeric identifier for the sub-folder "TS-Docs." Use the numeric identifiers of bots and folders in API requests such as the Start migration API.

The Enterprise Control Room user interface URL contains the ids for bots and folders. View a bot or folder to find its numeric identifier.

Bot {id} in the user interface

49502 is the unique numeric identifier for a bot.

```
<your_control_room_url>/#/bots/repository/public/taskbots/49502/view
```

Folder {id} in the user interface

40378 is the unique numeric identifier for a folder.

```
<your_control_room_url>/#/bots/repository/public/folders/40378
```

List workspaces folders and files

Search in the Enterprise Control Room private or public repository for folders and files, and use filters to find exactly the files and folders you need.

Prerequisites

- Authentication token for a registered user in the Enterprise Control Room
- The API URL: `<your_control_room_url>/v2/repository/workspaces/{id}/files/list`
- Determine whether you want to search the Enterprise Control Room public repository or your private repository.

URL strings `{id}`

- The `{id}` is a string that is passed to the request URL.
- There are 2 options for the .../workspaces/{id}/...
  - public: searches all the files and folders included in the public repository of the Enterprise Control Room.

- private: searches only the files and folders that are in a users private repository.

Procedure

1. Add an authentication token to the request header. Use the Authentication API to generate a JSON Web Token.
2. POST is the method used for this API.
3. The {id} in the URL is a string. In the URL, enter public or private for the {id}.
   `<your_control_room_url>/v2/repository/workspaces/public/files/list`
4. Enter the following filter in the request body. This filter searches for only bots with the string "finance" in the name. Reaching for the boolean value of false for the folder field excludes any folders from the search and finds only bots.

```
{
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "substring",
        "field": "name",
        "value": "finance"
      },
      {
        "operator": "eq",
        "field": "folder",
        "value": "false"
      }
    ]
  }
}
```

5. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND

   The response for this request returned 2 out of 1006 possible results. The request searched for any bots with "finance" in the name.

   We excluded folders that might have "finance" in the path or folder name by setting the search parameter for folders to false. This way the response contains on bots with the string "finance" in the bot name.

   Response body:

```
{
  "page": {
    "offset": 0,
    "total": 1006,
    "totalFilter": 2
  },
  "list": [
    {
      "id": "55709",
      "parentId": "55711",
      "name": "finance-01",
      "path": "Automation Anywhere\\Bots\\Finance\\finance-01",
      "description": "Minor updates",
      "type": "application/vnd.aa.taskbot",
      "size": "814",
      "folder": false,
      "folderCount": "0",
      "productionVersion": "",
      "latestVersion": "53161",
      "locked": false,
      "lockedBy": "0",
      "createdBy": "2538",
      "lastModifiedBy": "2538",
      "lastModified": "2020-04-08T16:57:36.753549Z",
      "workspaceId": "0",
      "botStatus": "PUBLIC",
      "hasErrors": false,
      "workspaceType": "UNKNOWN",
      "metadata": false,
      "uri": "",
      "version": "6",
      "hasTriggers": false
    },
    {
      "id": "56357",
      "parentId": "55711",
      "name": "Finance-02",
      "path": "Automation Anywhere\\Bots\\Finance\\Finance-02",
      "description": "Docs check in",
      "type": "application/vnd.aa.taskbot",
      "size": "809",
      "folder": false,
      "folderCount": "0",
      "productionVersion": "",
      "latestVersion": "53160",
      "locked": false,
      "lockedBy": "0",
      "createdBy": "2538",
      "lastModifiedBy": "2538",
      "lastModified": "2020-04-08T16:34:36.549250Z",
      "workspaceId": "0",
      "botStatus": "PUBLIC",
      "hasErrors": false,
      "workspaceType": "UNKNOWN",
      "metadata": false,
      "uri": "",
      "version": "3",
      "hasTriggers": false
    }
  ]
}
```

Next steps

Use the numeric values of the response parameters as input values for other API requests. Here is an example with the Start migration API.

- "id": "55709": a unique numeric identifier for a single bot can be used as the input for "botIds": [55709] in the Migration API.
- "parentId": "55711": a unique numeric identifier for a folder that contains bots can be used as the "folderIds": [55711] in the Migration API.

User management API overview

The User Management APIs enable you to create, search, update, or delete roles and users in your .

You can view the User Management API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.

User Management Roles

Users need the following permissions in order to create and manage users and roles.

View users

Users with these permissions are able to create and manage users. These are administrator permissions. It is recommended that non-administrator users be given limited permissions for creating and managing users. Learn how to create a role with limited permissions that can be assigned to users.

Create users

Create new users in the Enterprise Control Room.

Edit users

Edit all users in the Enterprise Control Room, including users created by other administrators.

Delete users

Delete any user in the Enterprise Control Room.

View roles

Users with this permission can view roles to which they have access.

Manage roles

Users can create, edit and delete roles to which they have access.

View licenses

Users with these permissions are able to view and manage device licenses. Device licenses are required to enable users to perform specific tasks. For example, Bot Creators require a DEVELOPMENT device license in order to create bots.

Manage user's device license

Users with this permission can assign device licenses to users.

Role APIs

Use Role APIs to create a role, search for roles, retrieve a specific role using an object ID, update a role, or delete a role.

Create a new role

Creates a new role with a new role name.

```
POST http://<your_control_room_url>/v1/usermanagement/roles
```

Retrieve a specific role

Retrieves a specific role based on a unique role ID.

```
GET http://<your_control_room_url>/v1/usermanagement/roles/{id}
```

Update role

Modifies an existing role name based on a unique role ID.

```
PUT http://<your_control_room_url>/v1/usermanagement/roles/{id}
```

Delete an existing role

Deletes an existing role based on a unique role ID.

```
DELETE http://<your_control_room_url>/v1/usermanagement/roles/{id}
```

User APIs

Use User APIs to create a user, search for users, retrieve a user, update a user, or delete a user.

Create a new user

Creates a user with a new user name.

```
POST http://<your_control_room_url>/v1/usermanagement/users
```

Search for users

Retrieves current users based on search criteria, such as filtering, sorting, and pagination.

```
POST http://<your_control_room_url>/v1/usermanagement/users/list
```

Retrieve a specific user

Retrieves user details based on a unique user ID.

```
GET http://<your_control_room_url>/v1/usermanagement/users/{uid}
```

Update an existing user

Modifies an existing user name based on a unique user ID.

```
PUT http://<your_control_room_url>/v1/usermanagement/users/{uid}
```

Delete an existing user

Deletes an existing user based on a unique user ID.

```
DELETE http://<your_control_room_url>/v1/usermanagement/users/{uid}
```

Related concepts
Authentication API overview
Related reference

Permissions to roles mapping

Create a new role

Use Create role API to create a new role with permissions in the Enterprise Control Room.

Prerequisites

Each permission requires the following parameters:

- id: The numeric value that uniquely identifies the permission.
- action: The action the permission enables.
- resourceType: The resource group to which the action belongs.

All 3 parameters are required to create the permission.
You need limited administrative permission to create roles.

View roles

> View user roles.
>
> Manage roles
>
> > Create and manage user roles

Typically a user is given the role permission in conjunction with user management permission. Permissions to roles mapping

- URL: `http://<your_control_room_url>/v1/usermanagement/roles`

  Replace the content in the angle brackets with your Enterprise Control Room URL.

- Method: POST
- Use the Swagger definition files installed with your Enterprise Control Room to test the APIs. View the available Swagger APIs at: `http://<your_control_room_url>/swagger/`
- You can also use a REST client to complete this task.

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. See Authentication API overview.
2. Select POST as the method.
   `POST http://<your_control_room_url>/v1/usermanagement/roles`
3. In the request body, enter a new name for the role.
4. Send the request.

   The following request creates a new role that allows a user to create and manage roles and user in her Enterprise Control Room.

   Request body:

   ```
   {
     "name": "Role to manage users and roles",
   ```

```
  "description": "These are limited administrator permission.",
  "permissions": [
    {
      "id": 1,
      "action": "usermanagement",
      "resourceType": "usermanagement"
    },
    {
      "id": 3,
      "action": "createuser",
      "resourceType": "usermanagement"
    },
    {
      "id": 4,
      "action": "updateuser",
      "resourceType": "usermanagement"
    },
    {
      "id": 2,
      "action": "deleteuser",
      "resourceType": "usermanagement"
    },
    {
      "id": 62,
      "action": "rolesview",
      "resourceType": "rolesmanagement"
    },
    {
      "id": 12,
      "action": "rolesmanagement",
      "resourceType": "rolesmanagement"
    }
  ]
}
```

5. Send the request.
   - In Swagger, click Execute.

- In a REST client, click SEND.

Response body:

```
{
  "id": 767,
  "createdBy": 3215,
  "createdOn": "2020-03-19T22:44:21Z",
  "updatedBy": 3215,
  "updatedOn": "2020-03-19T22:44:21Z",
  "tenantId": 1,
  "version": 0,
  "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
  "description": "These are limited administrator permission.",
  "name": "Role to manage users and roles",
  "accessRestriction": null,
  "permissions": [
    {
      "id": 1,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "usermanagement",
      "resourceId": null,
      "resourceType": "usermanagement"
    },
    {
      "id": 2,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "deleteuser",
      "resourceId": null,
      "resourceType": "usermanagement"
    },
    {
      "id": 4,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "updateuser",
      "resourceId": null,
      "resourceType": "usermanagement"
    },
    {
      "id": 12,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "rolesmanagement",
      "resourceId": null,
      "resourceType": "rolesmanagement"
    },
    {
      "id": 62,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "rolesview",
      "resourceId": null,
      "resourceType": "rolesmanagement"
    },
    {
      "id": 3,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "createuser",
      "resourceId": null,
      "resourceType": "usermanagement"
    }
  ],
  "countPrincipals": 0,
  "principals": []
}
```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token. This example has been formatted for readability.

```
curl -X POST "https://<your_control_room_url>/v1/usermanagement/roles"
-H "accept: application/json"
-H "X-Authorization: <token>"
-H "Content-Type: application/json" -d "{ "name": "Role to manage user and role
s",
"description": "These are limited administrator permission.",
"permissions": [ {
  "name": "Role to manage users and roles",
  "description": "These are limited administrator permission.",
  "permissions": [
    {
      "id": 1,
      "action": "usermanagement",
      "resourceType": "usermanagement"
    },
    {
      "id": 3,
      "action": "createuser",
      "resourceType": "usermanagement"
    },
    {
      "id": 4,
      "action": "updateuser",
      "resourceType": "usermanagement"
    },
    {
      "id": 2,
      "action": "deleteuser",
      "resourceType": "usermanagement"
    },
    {
      "id": 62,
      "action": "rolesview",
      "resourceType": "rolesmanagement"
    },
    {
      "id": 12,
      "action": "rolesmanagement",
      "resourceType": "rolesmanagement"
    }
  ]
}
```

```
]

}"
```

Retrieve a specific role

Use the Return Specific Role API to retrieve a specific role in the Enterprise Control Room.

Prerequisites

View Roles
: Users who have `view roles` permissions can retrieve details of a specific role. Permissions to roles mapping

- URL: `http://<your_control_room_url>/v1/usermanagement/roles/{ID}`
- Method: GET
- Use the Swagger definition files installed with your Enterprise Control Room to test the APIs. View the available Swagger APIs at: `http://<your_control_room_url>/swagger/`
- You can also use a REST client to complete this task.

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. Authentication API overview
2. Select GET as the method.
3. Add a role ID that you want to retrieve to the URL.
   `GET http://<your_control_room_url>/v1/usermanagement/roles/740`

   In this example we used the role ID for a customRole.

   - In a REST client, click SEND.
   - In the Swagger interface, click Execute.

   Response body:

```
{
  "id": 740,
  "createdBy": 2623,
  "createdOn": "2020-02-24T19:08:09Z",
  "updatedBy": 2623,
  "updatedOn": "2020-02-24T19:08:09Z",
  "tenantId": 1,
  "version": 0,
  "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
  "description": "",
  "name": "customRole",
  "accessRestriction": null,
  "permissions": [
    {
      "id": 59,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "managecredentials",
      "resourceId": null,
      "resourceType": "credentials"
    },
    {
      "id": 12,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "rolesmanagement",
      "resourceId": null,
      "resourceType": "rolesmanagement"
    },
    {
      "id": 62,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "rolesview",
      "resourceId": null,
      "resourceType": "rolesmanagement"
    },
    {
      "id": 58,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "myschedule",
      "resourceId": null,
      "resourceType": "taskscheduling"
    },
    {
      "id": 97,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:39Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:39Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "view",
      "resourceId": null,
      "resourceType": "dashboard"
    },
    {
      "id": 30,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "view",
      "resourceId": null,
      "resourceType": "devices"
    }
  ],
  "countPrincipals": 0,
  "principals": []
}
```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token.

```
curl -X GET "http://<your_control_room_url>/v1/usermanagement/roles/740"
-H "accept: application/json"
-H "X-Authorization: <authentication_token>"
-H "Content-Type: application/json" -d "{740}"
```

Update role

Use the Update Role API to update an existing role in the Enterprise Control Room.

Prerequisites

Edit Roles
: Users who have `edit roles` permissions can update a role.

- URL: `http://<your_control_room_url>/v1/usermanagement/roles/{ID}`

  Replace the content in the angle brackets with your Enterprise Control Room URL.

- Method: PUT
- Use the Swagger definition files installed with your Enterprise Control Room to test the APIs. View the available Swagger APIs at: `http://<your_control_room_url>/swagger/`
- You can also use a REST client to complete this task.

This example show how to add a single permission, View users, to the custom role with the ID of 740.

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. See, Authentication API overview.
2. Select PUT as the method.
3. In the request URL, add a role ID you want to update.
   `PUT http://<your_control_room_url>/v1/usermanagement/roles/740`

   Request body:

```
{
  "id": null,
  "name": "Add_one_permission",
  "accessRestriction": null,
  "permissions": [
    {
      "id": 1,
      "action": "usermanagement",
      "resourceType": "usermanagement"
    }
  ]
}
```

4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND.

   Response body:

```
{
  "id": 740,
  "createdBy": 2623,
  "createdOn": "2020-02-24T19:08:09Z",
  "updatedBy": 3215,
  "updatedOn": "2020-03-20T21:40:34Z",
  "tenantId": 1,
  "version": 4,
  "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
  "description": null,
  "name": "Add_one_permission",
  "accessRestriction": null,
  "permissions": [
    {
      "id": 1,
      "createdBy": 0,
      "createdOn": "2019-05-21T03:09:31Z",
      "updatedBy": 0,
      "updatedOn": "2019-05-21T03:09:31Z",
      "tenantId": 1,
      "version": 0,
      "tenantUuid": "e100fbce-008c-04ec-4063-7af0af91fb2f",
      "action": "usermanagement",
      "resourceId": null,
      "resourceType": "usermanagement"
    }
  ],
  "countPrincipals": 0,
  "principals": []
}
```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token. This example has been formatted for readability.

```
curl -X PUT "https://canary.supremomono.com/v1/usermanagement/roles/740"
-H "accept: application/json"
-H "X-Authorization: <web_token> "
-H "Content-Type: application/json" -d "{
  "id": null,
  "name": "Add_one_permission",
  "accessRestriction": null,
  "permissions": [
    {
      "id": 1,
      "action": "usermanagement",
      "resourceType": "usermanagement"
    }
  ]
}
}"
```

Delete an existing role

Use the Delete role API to delete an existing role in the Enterprise Control Room.

Prerequisites

Manage roles permission

> Users who have `manage roles` permissions can delete roles. However, only custom roles can be deleted, the system-created roles, the first 16 roles with ID 1 to 16, cannot be deleted.

- URL: `http://<your_control_room_url>/v1/usermanagement/roles/{ID}`

  Replace the content in the angle brackets with your Enterprise Control Room URL.
- Method: DELETE
- Use the Swagger definition files installed with your Enterprise Control Room to test the APIs. View the available Swagger APIs at: `http://<your_control_room_url>/swagger/`
- You can also use a REST client to complete this task.

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. See Authentication API overview.
2. Select DELETE as the method.
3. In the request header, add a role ID you want to delete.
   `DELETE http://<your_control_room_url>/v1/usermanagement/roles/770`
4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND.

   Response body:

   ```
   "OK"
   ```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token.

```
curl -X DELETE "http://<your_control_room_url>/v1/usermanagement/roles/22"
-H "accept: application/json"
```

```
-H "X-Authorization: <authentication_token>"
-H "Content-Type: application/json" -d "{770}"
```

Create a new user

Use the Create New User API to create a new user in the Enterprise Control Room.

Prerequisites

View and manage users and roles

Authenticate with a user that has the following ADMINISTRATION permissions:

View users

Users with these permissions are able to create and manage users. These are administrator permissions. It is recommended that non-administrator users be given limited permissions for creating and managing users. Learn how to create a role with limited permissions that can be assigned to users.

Create users

Create new users in the Enterprise Control Room.

Edit users

Edit all users in the Enterprise Control Room, including users created by other administrators.

Delete users

Delete any user in the Enterprise Control Room.

View roles

Users with this permission can view roles to which they have access.

Manage roles

Users can create, edit and delete roles to which they have access.

View licenses

Users with these permissions are able to view and manage device licenses. Device licenses are required to enable users to perform specific tasks. For example, Bot Creators require a DEVELOPMENT device license in order to create bots.

Manage user's device license

Users with this permission can assign device licenses to users.

Minimum required parameters

- Roles: Each user must have at least one role. The role id is required to create a role from the User Management API.

  Role based accessibility enables appropriate access to relevant data and actions.

  Note: For this example, we created a Bot Creator user. In the request body we assigned the following Permissions to roles mapping:
  - AAE_Basic (ID: 2)
  - AAE_Meta Bot Designer (ID: 13)
- username: String (255 max)
- email: Must conform to standard email format (username@domain.com)
- password: String: 8-15 characters in length. Allowable characters: a-z, A-Z, 0-9, @, -, _, !, #, $, %, &, and . (period)

Additional recommended parameters

- `"enableAutoLogin": true`
- `"username": "NumerOneUser"`
- `"firstName": "Doc"`
- `"lastName": "Writer"`
- `"email": "username@mydomain.com"`
- `"password": "changeme"`
- `"description": "Test user creation."`
- `"licenseFeatures": [ DEVELOPMENT, RUNTIME, IQBOTRUNTIME, ANALYTICSCLIENT, ANALYTICSAPI ]`

  Users can be created without an assigned device license. There are Permissions to roles mapping that enable privileges for specific users and roles.

- URL: `http://<your_control_room_url>/v1/usermanagement/users`

  Replace the content in the angle brackets with your Enterprise Control Room URL.

- Method: POST
- Use the Swagger installed with your Enterprise Control Room to test the APIs. View the available APIs at: `http://<your_control_room_url>/swagger/`
- You can also use a REST client to complete this task.

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. See Authentication API overview.
2. Select POST as the method.
   `POST http://<your_control_room_url>/v1/usermanagement/users`
3. In the request body, enter the mandatory parameters and recommend parameters.

   Request body :

```
{
  "roles": [
    {
      "id": 2
    },
    {
      "id": 13
    }
  ],
  "enableAutoLogin": true,
  "username": "BotCreatorUser",
  "firstName": "Robert",
  "lastName": "Developer",
  "email": "bob.dev@mydomain.com",
  "password": "changeme",
  "description": "Go create great bots.",
  "licenseFeatures": [
    "DEVELOPMENT"
  ]
}
```

4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND.

   The response returns user details.

   Response body:

```
{
  "id": 61,
  "email": "bob.dev@mydomain.com",
  "username": "botcreatoruser",
  "domain": null,
  "firstName": "Robert",
  "lastName": "Developer",
  "version": 0,
  "principalId": 61,
  "deleted": false,
  "roles": [
    {
      "name": "AAE_Basic",
      "id": 2,
      "version": 0
    },
    {
      "name": "AAE_Meta Bot Designer",
      "id": 13,
      "version": 0
    }
  ],
  "sysAssignedRoles": [],
  "groupNames": [],
  "permissions": [
    {
      "id": 97,
      "action": "viewbotstore",
      "resourceId": null,
      "resourceType": "botstore"
    },
    {
      "id": 33,
      "action": "upload",
      "resourceId": "7",
      "resourceType": "repositorymanager"
    },
    {
      "id": 61,
      "action": "createstandard",
      "resourceId": null,
      "resourceType": "credentialattribute"
    },
    {
      "id": 93,
      "action": "download",
      "resourceId": "9",
      "resourceType": "repositorymanager"
    },
    {
      "id": 134,
      "action": "viewuserbasic",
      "resourceId": null,
      "resourceType": "usermanagement"
    },
    {
      "id": 29,
      "action": "view",
      "resourceId": null,
      "resourceType": "repositorymanager"
    },
    {
      "id": 62,
      "action": "metabotdesigner",
      "resourceId": null,
      "resourceType": "metabot"
    },
    {
      "id": 34,
      "action": "download",
      "resourceId": "7",
      "resourceType": "repositorymanager"
    },
    {
      "id": 92,
      "action": "upload",
      "resourceId": "9",
      "resourceType": "repositorymanager"
    }
  ],
  "licenseFeatures": [
    "DEVELOPMENT"
  ],
  "emailVerified": true,
  "passwordSet": false,
  "questionsSet": false,
  "enableAutoLogin": true,
  "disabled": false,
  "clientRegistered": false,
  "description": "Go create great bots.",
  "createdBy": 19,
  "createdOn": "2020-02-09T23:25:20Z",
  "updatedBy": 19,
  "updatedOn": "2020-02-09T23:25:20Z",
  "publicKey": null,
  "appType": null,
  "routingName": null,
  "appUrl": null
}
```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token.

```
curl -X POST -H 'X-Authorization: <authentication_token>' -i '<your_control_roo
m_url>/v1/usermanagement/users' --data '{
  "roles": [
    {
      "id": 2
    },
    {
      "id": 13
    }
  ],
  "enableAutoLogin": true,
  "username": "BotCreatorUser",
  "firstName": "Robert",
  "lastName": "Developer",
  "email": "bob.dev@mydomain.com",
  "password": "changeme",
  "description": "Go create great bots.",
  "licenseFeatures": [
   "DEVELOPMENT"
  ]
}'
```

Next steps

- You can verify that the user was created by logging in to the Enterprise Control Room as the user that you created.
- System assigned roles, sysAssignedRoles, include a set of permissions that are required by the roles you assigned to the user and default roles that are assigned to all users.

Search for users

Use the Search for Users API to search for all users in the Enterprise Control Room.

Prerequisites

View Users

Users who have `view users` permissions can retrieve all users.

- URL: `http://<your_control_room_url>/v1/usermanagement/users/list`

  Replace the content in the angle brackets with your Enterprise Control Room URL.
- Method: POST
- Use the Swagger definition files installed with your Enterprise Control Room to test the APIs. View the available Swagger APIs at: `http://<your_control_room_url>/swagger/`
- You can also use a REST client to complete this task.

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. See Authentication API overview.
2. Select POST as the method.
   Apply filters to perform basic conditional queries and pagination control for processing web pages. There are three basic features related to filtering: filtering conditions, sorting columns, and pagination parameters. See Filters in an API request body.
   `POST http://<your_control_room_url>/v1/usermanagement/users/list`
3. Send the request.
   - In a REST client, click SEND.
   - In the Swagger interface, click Execute.

   Request body:

   The following request finds all users with a `username` that contains `doc` and who were created between December 1 and December 31, 2019.

```
{
  "fields": [],
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "substring",
        "field": "username",
        "value": "doc"
      },
      {
        "operator": "gt",
        "field": "createdOn",
        "value": "2020-03-01T00:00:00.989Z"
      },
      {
        "operator": "lt",
        "field": "createdOn",
        "value": "2020-03-06T23:00:00.123Z"
      }
    ]
  }
}
```

The response in this example returned data for the `doc` username.

Response body:

```
{
  "page": {
    "offset": 0,
    "total": 596,
    "totalFilter": 1
  },
  "list": [
    {
      "id": 3259,
      "username": "docs-admin",
      "domain": "",
      "firstName": "",
      "lastName": "",
      "version": 16,
      "principalId": 3259,
      "email": "terry.martin@automationanywhere.com",
      "emailVerified": true,
      "passwordSet": true,
      "questionsSet": true,
      "enableAutoLogin": false,
      "disabled": false,
      "clientRegistered": false,
      "description": "docs resource",
      "createdBy": 2623,
      "createdOn": "2020-03-05T21:37:00.087Z",
      "updatedBy": 3259,
      "updatedOn": "2020-03-05T22:05:39.160Z",
      "licenseFeatures": [],
      "roles": [
        {
          "id": 1,
          "name": "AAE_Admin"
        },
        {
          "id": 11,
          "name": "AAE_Pool Admin"
        },
        {
          "id": 10,
          "name": "AAE_Queue Admin"
        }
      ],
      "deleted": false
    }
  ]
}
```

Response body parameters:

Parameter Name	Description
id	System-generated ID number who created a user.

Parameter Name	Description
username	User name for a new user.
domain	Active directory domain name.
version	System-generated version number for a new user.
email	New user email address.
passwordSet	String: 8-15 characters; a-z, A-Z, 0-9, @, -, _, !, #, $, %, &, and . (period). Set a password for a new user only.
PrincipalId	System-generated ID number of an active directory principal user who created a new user.
Permission	A specific permission ID.
licenseFeature	Automation Anywhere license associated with this role.
Roles: id	System-generated role ID number associated with this user. Not every user has an associated role.
createdBy	System-generated ID number of an admin user who created a new user.
updatedBy	System-generated ID number of an admin user who updated the user.

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token.

```
curl -X POST "http://<your_control_room_url>/v1/usermanagment/users/list"
-H "accept: application/json"
-H "X-Authorization: <authentication_token>"
-H "Content-Type: application/json" -d "{
  "fields": [],
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "substring",
        "field": "username",
        "value": "doc"
      },
      {
        "operator": "gt",
        "field": "createdOn",
        "value": "2019-12-01T00:00:00.989Z"
      },
      {
        "operator": "lt",
        "field": "createdOn",
        "value": "2019-12-06T23:00:00.123Z"
      }
    ]
  }
}'
```

Retrieve a specific user

Use the Get Use Details API to retrieve a specific user in the Enterprise Control Room.

Prerequisites

View Users permission

Users who have `view users` permissions can retrieve details of a specific user.

- URL: `http://<your_control_room_url>/v1/usermanagement/users/{ID}`

  Replace the content in the angle brackets with your Enterprise Control Room URL.

- Method: GET
- Use the Swagger definition files installed with your Enterprise Control Room to test the APIs. View the available Swagger APIs at: `http://<your_control_room_url>/swagger/`
- You can also use a REST client to complete this task.

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. See Authentication API overview.
2. Select GET as the method.
3. In the request header, add a specific user ID you want to retrieve.
   `GET http://<your_control_room_url>/v1/usermanagement/users/2624`
4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND.

   Response body:

   ```
   {
     "id": 2624,
     "email": "a@a.com",
     "username": "docs-deploy",
   ```

```
    "domain": null,
    "firstName": "",
    "lastName": "",
    "version": 6,
    "principalId": 2624,
    "deleted": false,
    "roles": [
      {
        "name": "DocsDeploy",
        "id": 641,
        "version": 6
      }
    ],
    "sysAssignedRoles": [],
    "groupNames": [],
    "permissions": [
      {
        "id": 6,
        "action": "manageallschedules",
        "resourceId": null,
        "resourceType": "taskscheduling"
      },
      {
        "id": 11,
        "action": "deleteschedule",
        "resourceId": null,
        "resourceType": "taskscheduling"
      },
      {
        "id": 7,
        "action": "run",
        "resourceId": null,
        "resourceType": "repositorymanager"
      },
      {
        "id": 11323,
        "action": "view",
        "resourceId": null,
        "resourceType": "eventtriggers"
      },
      {
        "id": 59,
        "action": "managecredentials",
        "resourceId": null,
        "resourceType": "credentials"
      },
      {
        "id": 54,
        "action": "createfolders",
        "resourceId": null,
        "resourceType": "repositorymanager"
      },
      {
        "id": 53,
        "action": "manageallmyfolderschedules",
        "resourceId": null,
        "resourceType": "taskscheduling"
      },
      {
        "id": 28,
        "action": "view",
        "resourceId": null,
        "resourceType": "taskscheduling"
      },
      {
        "id": 55,
        "action": "renamefolders",
        "resourceId": null,
        "resourceType": "repositorymanager"
      },
      {
        "id": 30,
        "action": "view",
        "resourceId": null,
        "resourceType": "devices"
      },
      {
        "id": 58,
        "action": "myschedule",
        "resourceId": null,
        "resourceType": "taskscheduling"
      },
      {
        "id": 13,
        "action": "setproductionversion",
        "resourceId": null,
        "resourceType": "repositorymanager"
      },
      {
        "id": 10,
        "action": "addschedule",
        "resourceId": null,
        "resourceType": "taskscheduling"
      },
      {
        "id": 97,
        "action": "view",
        "resourceId": null,
        "resourceType": "dashboard"
      },
      {
        "id": 5,
        "action": "updateschedule",
        "resourceId": null,
        "resourceType": "taskscheduling"
      },
      {
        "id": 29,
        "action": "view",
        "resourceId": null,
        "resourceType": "repositorymanager"
      },
      {
        "id": 31,
        "action": "export",
        "resourceId": null,
        "resourceType": "repositorymanager"
      },
      {
        "id": 18370,
        "action": "participate",
        "resourceId": "141",
        "resourceType": "queue"
      },
      {
        "id": 32,
        "action": "import",
        "resourceId": null,
        "resourceType": "repositorymanager"
      }
    ],
    "licenseFeatures": [
      "RUNTIME"
    ],
    "emailVerified": true,
    "passwordSet": true,
    "questionsSet": true,
    "enableAutoLogin": false,
    "disabled": false,
    "clientRegistered": false,
    "description": "",
    "createdBy": 2036,
    "createdOn": "2019-11-09T22:11:04Z",
    "updatedBy": 3177,
    "updatedOn": "2020-02-15T20:15:27Z",
    "publicKey": null,
    "appType": null,
    "routingName": null,
    "appUrl": null
  }
```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token. This example has been formatted for readability.

```
curl -X GET -k -H 'X-Authorization: <authentication_token>'
-i 'https://canary.supremomono.com/v1/usermanagement/users/2624'
--data '{ }'
```

Update an existing user

Use the Update User Details API to update an existing user information in the Enterprise Control Room.

Prerequisites

Edit Users permission

Users who have `edit users` permission can update a specific user details.

- URL: `http://<your_control_room_url>/v1/usermanagement/users/{ID}`

  Replace the content in the angle brackets with your Enterprise Control Room URL.
- Method: PUT
- Use the Swagger definition files installed with your Enterprise Control Room to test the APIs. View the available Swagger APIs at: `http://<your_control_room_url>/swagger/`
- You can also use a REST client to complete this task.

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. See Authentication API overview.
2. Select PUT as the method.
3. In the request header, add an existing user ID you want to update. To find a user ID you want to update, execute the Search Users API.
   `PUT http://<your_control_room_url>/v1/usermanagement/users/3015`

   In the request body, add the mandatory parameters.

Parameter name	Mandatory parameters	Type	Description
Username	Yes	String (255 max)	New user name
Email	Yes	String; must include @ sign, such as a@a.com	New user email
Roles: name	Yes	String (255 max)	New name for the role

Request body

```
{
  "roles": [
    {
      "id": 637
    }
  ],
  "email": "a@a.com"
}
```

4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND.

   Response body:

```
{
  "id": 3015,
  "email": "a@a.com",
  "username": "docstest02",
  "domain": null,
  "firstName": null,
  "lastName": null,
  "version": 0,
  "principalId": 3015,
  "deleted": false,
  "roles": [
    {
      "name": "DocsAPIKey",
      "id": 637,
      "version": 0
    }
  ],
  "sysAssignedRoles": [],
  "groupNames": [],
  "permissions": [
    {
      "id": 58,
      "action": "myschedule",
      "resourceId": null,
      "resourceType": "taskscheduling"
    },
    {
      "id": 59,
      "action": "managecredentials",
      "resourceId": null,
      "resourceType": "credentials"
    },
    {
      "id": 91,
      "action": "generateapikey",
      "resourceId": null,
      "resourceType": "api"
    },
    {
      "id": 97,
      "action": "view",
      "resourceId": null,
      "resourceType": "dashboard"
    },
    {
      "id": 30,
      "action": "view",
      "resourceId": null,
      "resourceType": "devices"
    }
  ],
  "licenseFeatures": [],
  "emailVerified": true,
  "passwordSet": false,
  "questionsSet": false,
  "enableAutoLogin": false,
  "disabled": false,
  "clientRegistered": false,
  "description": null,
  "createdBy": 2623,
  "createdOn": "2020-01-31T17:36:23Z",
  "updatedBy": 3215,
  "updatedOn": "2020-03-22T22:59:04Z",
  "publicKey": null,
  "appType": null,
  "routingName": null,
  "appUrl": null
}
```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token.

```
curl -X PUT "http://<your_control_room_url>/v1/usermanagement/users/27"
-H "accept: application/json"
-H "X-Authorization: <authentication_token>"
-H "Content-Type: application/json" -d "'{
    "username": "docsusermd2",
    "domain": "",
    "firstName": "DocsUserMHD1",
    "lastName": "DocUserMHD2",
    "version": 2,
    "principalId": 27,
    "email": "aamd@aa.com",
    "description": "Created a user to create other roles and users",
    "createdOn": "2019-11-26T23:44:12.937Z",
    "updatedOn": "2019-11-26T23:51:39.163Z",
    "roles": [{
      "id": 27,
      "name": "RoleBotDocsMD6",
      "version": "0"
    }],
    "deleted": false
  }'
```

Delete an existing user

Use the Delete Existing User API to delete an existing user in the Enterprise Control Room.

Prerequisites

Edit Users

Users who have `edit users` permissions can delete an existing user.

JSON Web Token (JWT)

All Enterprise Control Room APIs require a JSON Web Token (JWT) to access the APIs. Generate an authentication token using the Authentication API. See Authentication API overview.

- URL: `http://<your_control_room_url>/v1/usermanagement/users/{ID}`

  Replace the content in the angle brackets with your Enterprise Control Room URL.

- Method: DELETE

Procedure

1. Add an authentication token to the request header.
   Use the Authentication API to generate a JSON Web Token. See Authentication API overview.
2. Select DELETE as the method.
3. In the request header, add an existing user ID you want to delete.
   `DELETE http://<your_control_room_url>/v1/usermanagement/users/3014`
4. Send the request.
   - In Swagger, click Execute.
   - In a REST client, click SEND.

   Response body:

```
{
  "id": 3014,
  "email": "a@a.com",
  "username": "docstest01",
  "domain": null,
  "firstName": null,
  "lastName": null,
  "version": 4,
  "principalId": 3014,
  "deleted": false,
  "roles": [],
  "sysAssignedRoles": [],
  "groupNames": [],
  "permissions": [],
  "licenseFeatures": [],
  "emailVerified": true,
  "passwordSet": false,
  "questionsSet": false,
  "enableAutoLogin": false,
  "disabled": false,
  "clientRegistered": false,
  "description": null,
  "createdBy": 2623,
  "createdOn": "2020-01-31T17:33:16Z",
  "updatedBy": 3215,
  "updatedOn": "2020-03-22T22:51:48Z",
  "publicKey": null,
  "appType": null,
  "routingName": null,
  "appUrl": null
}
```

Note: You can also run REST requests from a command terminal. The following is a curl request example. This example is formatted for readability. Replace the text inside the angle brackets, `<authentication_token>`, with your authentication token.

```
curl -X DELETE "http://<your_control_room_url>/v1/usermanagement/users/3014"
-H "accept: application/json"
-H "X-Authorization: <authentication_token>"
-H "Content-Type: application/json" -d "{ }"
```

Workload Management API overview

Use the Workload Management (WLM) API to programmatically manage and create workitem models, queues, workitems, and automations in your Enterprise Control Room.

You can view the Workload Management API in the Community Edition.
Note: APIs can be viewed in the Community Edition, but API functionality is limited. You need a licensed Enterprise A2019 Edition to access the full functionality of the APIs.
You need the following permissions to use this API.

- AAE_Pool Admin
- AAE_Queue Admin

Filters in an API request body

Filtering provides basic conditional queries and page control for processing API requests. There are 3 basic features related to filtering: filtering conditions, sorting, and pagination parameters.

Here is a representation of the JSON filtering structure used in the Automation Anywhere APIs.

```
{
  "filter": {
    "operator": "NONE",
    "operands": [
      null
    ],
    "field": "string",
    "value": "string"
  },
  "sort": [
    {
      "field": "string",
      "direction": "asc"
    }
  ],
  "page": {
    "offset": 0,
    "length": 0
  }
}
```

The most basic part of this JSON object is the filter array.

Understanding filters

Basic filter

Filters can be used search for a single condition or they can be wrapped in logical operands AND and OR. Filtering can be a simple conditional evaluation of a single field. The operator, field, and value used in a filter are specific to the API they are used in.
Note: Values in the angle brackets < > include a list of all potential values. There should be only one value for each parameter.

Single parameter filter

```
{
  "filter": {
    "operator": "<NONE, lt, le, eq, ne, ge, gt, substring, and, or, no
t>",
    "field": "string",
    "value": "string"
  }
}
```

Two parameter filter

```
{
  "filter": {
    "operator": "<and, or>",
    "operands": [
      {
        "operator": "<NONE, lt, le, eq, ne, ge, gt, substring, and, or
, not>",
        "field": "string",
        "value": "string"
      },
      {
        "operator": "<NONE, lt, le, eq, ne, ge, gt, substring, and, or
, not>",
        "field": "string",
        "value": "string"
      }
    ]
  }
}
```

Page

```
"page":{
    "offset":0,
    "length":0
}
```

Pagination rules parameters

- Offset:

  Type: integer

  The numeric value that indicates how many rows into a table that the filter starts evaluating.

- Length

  Type: integer

  The number of lines that are returned in a single page of results.

Sort

```
"sort": [
   {
     "field": "string",
     "direction": "<asc, desc>"
   }
```

- Field: The field that you want the results to be filtered by. This must be a supported filterable field. Filterable fields vary depending on the API.
- Direction

  Type: Enum [desc, asc]

  - asc = ascending (smallest to largest, 0 to 9, A to Z)
  - desc = descending (largest to smallest, 9 to 0, Z to A)

API filter examples

User management filter example

This example filter is based on the User Management API fields and parameters. This filter searches for the user's login name, username, and the user's real name, firstName.

Repository management filter example

This example filter is based on the Repository Management API fields and parameters. This filter example searches on the bot status and name.

Migration list results filter example

This example filter is based on the Migration API fields and parameters. This filter searches for migrations that contain a specific string in the name and was started, updatedOn, between two dates.

User management filter example

This example filter is based on the User Management API fields and parameters. This filter searches for the user's login name, username, and the user's real name, firstName.

This filter searches for the string "bot-creator" in the username field and the string "Adweta" in the firstName field.

```
{
  "sort": [
    {
      "field": "username",
      "direction": "asc"
    }
  ],
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "substring",
        "value": "bot-creator",
        "field": "username"
      },
      {
        "operator": "substring",
        "value": "Adweta",
        "field": "firstName"
      }
    ]
  }
}
```

sort

- field: the name of the field used to sort the response.
- direction: the sort order. It can be asc, ascending, or desc, descending.

filter

Filter consists for an operator, value, and field. Filters are operands when used in conjunction with a boolean operator, such as and.

- operands: filters are used as operands when combined in a filter by using a boolean operator. There are two available boolean operators:
  - or: one of the conditions must be met.
  - and: all of the conditions must be met.
- operator: there are 11 operators `NONE, lt, le, eq, ne, ge, gt, substring, and, or, not`. And and or are used to evaluate multiple filters together. The other operators are used to evaluate values within individual filters. Not all operators work with all fields.
- field: the name of the field used in the filter.
- value: the value of the field to be evaluated.

Repository management filter example

This example filter is based on the Repository Management API fields and parameters. This filter example searches on the bot status and name.

This filter searches for bots in the Enterprise Control Room repository with the string "finance" in the name of the file that have a status of "CHECKED_OUT."
Note: The Repository management API uses role based access. That means users can only see the files and folders to which they have access.

```
{
  "sort": [
    {
      "field": "directory",
      "direction": "desc"
    },
    {
      "field": "name",
      "direction": "asc"
    }
  ],
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "substring",
        "value": "finance",
        "field": "name"
      },
      {
        "operator": "eq",
        "value": "CHECKED_OUT",
        "field": "botStatus"
      }
    ]
  }
}
```

sort

- field: the name of the field used to sort the response.
- direction: the sort order. It can be asc, ascending, or desc, descending.

filter

Filter consists for an operator, value, and field. Filters are operands when used in conjunction with a boolean operator, such as and.

- operands: filters are used as operands when combined in a filter by using a boolean operator. There are two available boolean operators:
  - or: one of the conditions must be met.
  - and: all of the conditions must be met.
- operator: there are 11 operators `NONE, lt, le, eq, ne, ge, gt, substring, and, or, not`. And and or are used to evaluate multiple filters together. The other operators are used to evaluate values within individual filters. Not all operators work with all fields.
- field: the name of the field used in the filter.
- value: the value of the field to be evaluated.

Migration list results filter example

This example filter is based on the Migration API fields and parameters. This filter searches for migrations that contain a specific string in the name and was started, updatedOn, between two dates.

This filter searches for migrations that contain "doc" in the name and were started between April 8, 2020 at midnight and April 13, 2020 at midnight.

```
{
  "filter": {
    "operator": "and",
    "operands": [
      {
        "operator": "gt",
        "field": "updatedOn",
        "value": "2020-04-08T00:00:00.001Z"
      },
      {
        "operator": "substring",
        "field": "name",
        "value": "doc"
      },
      {
        "operator": "lt",
        "field": "updatedOn",
        "value": "2020-04-13T00:00:00.001Z"
      }
    ]
  }
}
```

filter

Filter consists for an operator, value, and field. Filters are operands when used in conjunction with a boolean operator, such as and.

- operands: filters are used as operands when combined in a filter by using a boolean operator. There are two available boolean operators:
  - or: one of the conditions must be met.
  - and: all of the conditions must be met.
- operator: there are 11 operators `NONE, lt, le, eq, ne, ge, gt, substring, and, or, not`. And and or are used to evaluate multiple filters together. The other operators are used to evaluate values within individual filters. Not all operators work with all fields.
- field: the name of the field used in the filter.
- value: the value of the field to be evaluated.

Permissions to roles mapping

Create roles from the Enterprise A2019 Administration user interface or through the User Management API by assigning a set of permission that enable users to access related features.

Roles are a logical container for permissions and have interdependencies with bots, users, and licenses. Users with the appropriate administrative permission can create custom roles and assign the roles to users. The following topics provide descriptions of the features and the necessary information to create roles with the User Management API.

- Dashboard permissions
  The dashboard permission provides view access to dashboards for all users.
- Activity permissions
  Activity permissions enable users to view, manage, and schedule bot activities.
- Event triggers permissions
  Enable users to run bots automatically depending on a specific event, such as a new window opening. You can limit users to only view triggers or to view and manage triggers.
- Bot permissions
  Bot permission include features for managing bots and the crendentials used by bots.
- MetaBot permission
  MetaBot are obsolete in Enterprise A2019; however, this permission is available to insure that all functionality previously supported by MetaBot is supported in Enterprise A2019.
- Package manager permissions
  The package manager permission enables users to view or view and manage action packages.
- Devices permissions
  The devices permissions enable users to register, view, and manage devices used to run bots.
- Workload permissions
  Workload permissions enable users to manage and create workitem models, queues, workitems, and automations in their Enterprise A2019 control room.
- Bot Store permissions
  Vies and manage your activity in the Bot Store marketplace for bots and Digital Workers.
- Audit log permissions
  View logs and details of specific activities.Enable users to view logs from the Enterprise A2019 control room.
- Administration permissions
  Administrators manage settings related to all aspects of the Enterprise A2019 control room, including users, roles, action packages, and licensing.
- API permissions
  Enable access to APIs and API related features.
- IQ Bot permissions
  IQ Bot provides cognitive (intelligent) automation that can learn further from humans to uncover and transform important, but less structured data to automate business processes quickly and efficiently, simultaneously reducing human error.

Related concepts
Enterprise Control Room APIs

Dashboard permissions

The dashboard permission provides view access to dashboards for all users.

View dashborads

View dashbord is a default permission for all users. All users have permission to view dashboards.

```
{
  "id": 96,
  "action": "view",
  "resourceType": "dashboard"
}
```

Related reference
Permissions to roles mapping

Activity permissions

Activity permissions enable users to view, manage, and schedule bot activities.

Activity

View my in progress activity

```
{
  "id": 58,
  "action": "myschedule",
  "resourceType": "taskscheduling"
}
```

All users can view their own activity.

Manage my in progress activity

```
{
  "id": 51,
  "action": "managemyschedule",
  "resourceType": "taskscheduling"
}
```

All users can pause, resume or cancel their own activity and move their finished activities to history.

View everyone's in progress activity from my folders

```
{
  "id": 36,
  "action": "everyoneschedule",
  "resourceType": "taskscheduling"
}
```

The user can monitor those ongoing automations where the user has either run or schedule access on the respective TaskBot.

Manage everyone's in progress activity

```
{
  "id": 52,
  "action": "manageeveryoneschedule",
  "resourceType": "taskscheduling"
}
```

The user can monitor and manage (pause, resume, cancel) those ongoing automations where the user has either run or schedule access on the respective TaskBot. Users can also move the finished automations to history.

View my scheduled bots

```
{
  "id": 28,
  "action": "view",
  "resourceType": "taskscheduling"
}
```

Users can see their scheduled bots regardless of which user scheduled the bot.

Schedule my bots to run

```
{
  "id": 10,
  "action": "addschedule",
  "resourceType": "taskscheduling"
}
```

This requires the ability to view and manage Bot runners.

Edit my scheduled activity

```
{
  "id": 5,
  "action": "updateschedule",
  "resourceType": "taskscheduling"
}
```

Users can edit their scheduled bots, even if the bots are scheduled by a different user.

Delete my scheduled activity

```
{
  "id": 11,
  "action": "deleteschedule",
  "resourceType": "taskscheduling"
}
```

Users can delete schedules for any of their bots regardless of which users scheduled the bot.

View and manage ALL scheduled activity from my folders

```
{
  "id": 53,
  "action": "manageallmyfolderschedules",
  "resourceType": "taskscheduling"
}
```

Users can view, edit, and delete all the schedules on the bot Folders that the user has access to. This includes the schedules that the user created or schedules created by other users.

View and manage ALL scheduled activity

```
{
  "id": 6,
  "action": "manageallschedules",
  "resourceType": "taskscheduling"
}
```

Users can view, edit, and delete all the schedules in the system. This includes the schedules that the user created or schedules created by other users.

Related reference
Permissions to roles mapping
Activity

Event triggers permissions

Enable users to run bots automatically depending on a specific event, such as a new window opening. You can limit users to only view triggers or to view and manage triggers.

Event Triggers

View event triggers

```
{
  "id": 907,
  "action": "view",
  "resourceType": "eventtriggers"
}
```

Users can only view event triggers.

Manage event triggers

```
{
  "id": 908,
  "action": "manage",
  "resourceType": "eventtriggers"
}
```

Users can manage event triggers.

Related reference
Permissions to roles mapping

Bot permissions

Bot permission include features for managing bots and the crendentials used by bots.

Bots

View my bots

```
{
  "id": 29,
  "action": "view",
  "resourceType": "repositorymanager"
}
```

Users can view the bots they create.

Run my bots

```
{
  "id": 7,
  "action": "run",
  "resourceType": "repositorymanager"
}
```

Users can run their own bots.

Export bots

```
{
  "id": 31,
  "action": "export",
  "resourceType": "repositorymanager"
}
```

A user can export bots and related bot dependencies. Exporting bots requires the user to have download permission for the bot and its dependencies.

Import bots

```
{
  "id": 32,
  "action": "import",
  "resourceType": "repositorymanager"
}
```

A user can import bots and their related dependencies. Importing a bot requires a user to have upload permissions for the bot and its dependencies.

Create folders

```
{
  "id": 54,
  "action": "createfolders",
  "resourceType": "repositorymanager"
}
```

Users can create subfolders inside folders to which they already have access.

Rename folders

```
{
  "id": 55,
  "action": "renamefolders",
  "resourceType": "repositorymanager"
}
```

Users can rename subfolders to which they already have access.
Note: Only empty folders can be renamed.

Manage my credentials and lockers

```
{
  "id": 59,
  "action": "managecredentials",
  "resourceType": "credentials"
}
```

By default, all users can view and manage their credentials. Other users can give permissions to interact with other lockers.

Manage my locker

```
{
  "id": 26,
  "action": "create",
  "resourceType": "locker"
}
```

Users can create and manage their own lockers.

Administer ALL lockers

This permission is currently not implemented in production.

Create standard attributes for a credential

```
{
  "id": 61,
  "action": "createstandard",
  "resourceType": "credentialattribute"
}
```

Users can create standard, shared attributes for a credential, in addition to user-provided attributes.

View and edit ALL credentials attributes value

```
{
  "id": 64,
  "action": "updateany",
  "resourceType": "credentialattributevalue"
}
```

The user can view and update attribute values of user-provided credentials that belong to other users through an API.

Bots auto-login credentials API

```
{
  "id": 46,
  "action": "botautologinapi",
  "resourceType": "credentialattributevalue"
}
```

The user can set the Auto-Login credentials of other users through an API.

Credentials and lockers

Related reference

Permissions to roles mapping

MetaBot permission

MetaBot are obsolete in Enterprise A2019; however, this permission is available to insure that all functionality previously supported by MetaBot is supported in Enterprise A2019.

MetaBot

Access to MetaBot Designer

Permission/feature ID: Not available
Bot Creator users can access MetaBot Designer to view, create, and update MetaBots.
Note: This feature is used for internal features and is not user assignable.

Related reference

Permissions to roles mapping

Package manager permissions

The package manager permission enables users to view or view and manage action packages.

Package Manager

View packages

```
{
  "id": 92,
  "action": "view",
  "resourceType": "packagemanager"
}
```

Users can view packages. Packages are groups of actions used by bots.

Manage Packages

```
{
  "id": 93,
  "action": "manage",
  "resourceType": "packagemanager"
}
```

Users can view and manage packages.

Related reference
Permissions to roles mapping

Devices permissions

The devices permissions enable users to register, view, and manage devices used to run bots.

Devices

Register device

```
{
  "id": 94,
  "action": "register",
  "resourceType": "devices"
}
```

Users can register a localhost device.

View and manage ALL device(s)

```
{
  "id": 94,
  "action": "register",
  "resourceType": "devices"
}
```

Allows users to register a localhost as a device.

Delete the device(s)

```
{
  "id": 240,
  "action": "delete",
  "resourceType": "devices"
}
```

Users can delete devices they have registered.

View and manage my Bot runners, Bot creators and device pools

Create device pools

```
{
  "id": 40,
  "action": "create",
  "resourceType": "pool"
}
```

Users can create and manage their own device pools.

Administer ALL device pools

Permission/feature ID: Not available

This permission is not yet enabled.

Related reference

Permissions to roles mapping

Workload permissions

Workload permissions enable users to manage and create workitem models, queues, workitems, and automations in their Enterprise A2019 control room.

Workload

View and manage my queues

```
{
  "id": 58,
  "action": "myschedule",
  "resourceType": "taskscheduling"
}
```

Users can create and manage their own queues.

Create queue

```
{
  "id": 41,
  "action": "create",
  "resourceType": "queue"
}
```

Allow the user to create and manage their own queues.

Administer ALL queues

```
{
  "id": 67,
  "action": "accessresourceany",
  "resourceType": "pool"
}
```

Allows the user to manage all queues in the system. Only available to users with the AAE_Pool Admin role.

SLA Calculator

```
{
  "id": 42,
  "action": "calculate",
  "resourceType": "sla"
}
```

Users can calculate workload service level agreements (SLA).

Related reference
Permissions to roles mapping

Bot Store permissions

Vies and manage your activity in the Bot Store marketplace for bots and Digital Workers.

Bot Store

View Bot Store

```
{
  "id": 1163,
  "action": "view",
  "resourceType": "botstore"
}
```

Users can upload a bot package or from the Bot Store to the Digital Worker Enterprise Control Room private workspace.

Add bots from the Bot Store to My Bots

```
{
  "id": 1164,
  "action": "addfrom",
  "resourceType": "botstore"
}
```

Users can add the bot package or Digital Worker from the Bot Store to their Enterprise Control Roomprivate workspace.

Submit bots to Bot Store

```
{
  "id": 1165,
  "action": "submit",
  "resourceType": "botstore"
}
```

The user can submit a bot package or Digital Worker to the Bot Store.

Related reference
Permissions to roles mapping

Audit log permissions

View logs and details of specific activities.Enable users to view logs from the Enterprise A2019 control room.

Audit log

View everyone's audit log actions

```
{
  "id": 14,
  "action": "recentactivities",
  "resourceType": "recentactivities"
}
```

Users can view all audit log activity for the Enterprise Control Room.

Related reference
Permissions to roles mapping

Administration permissions

Administrators manage settings related to all aspects of the Enterprise A2019 control room, including users, roles, action packages, and licensing.

It is recommended that you create individual roles wilth only specific administrative permissions. You can then assing the limited permission to users with just that specific permission requirement.

Administration

View users

```
{
  "id": 1,
  "action": "usermanagement",
  "resourceType": "usermanagement"
}
```

Users can only view all other users in the system.

Create users

```
{
  "id": 3,
  "action": "createuser",
  "resourceType": "usermanagement"
}
```

Users can create new users in the Enterprise Control Room.

Edit users

```
{
  "id": 4,
  "action": "updateuser",
  "resourceType": "usermanagement"
}
```

Users can edit all users in the system.

Delete users

```
{
  "id": 2,
  "action": "deleteuser",
  "resourceType": "usermanagement"
}
```

Users can delete any user in the Enterprise Control Room.

View roles

```
{
  "id": 62,
  "action": "rolesview",
  "resourceType": "rolesmanagement"
}
```

Users with this permission are able to view the different roles in an Enterprise A2019 control room.

Manage roles

```
{
  "id": 12,
  "action": "rolesmanagement",
  "resourceType": "rolesmanagement"
}
```

Users with this permission are able to manage as well as view the different roles in an Enterprise A2019 control room.

View migration

```
{
  "id": 1166,
  "action": "view",
  "resourceType": "migration"
}
```

Users can view new migrations.

Manage migration

```
{
  "id": 56,
  "action": "manage",
  "resourceType": "migration"
}
```

Users can view and run new migrations.

Update migration status

```
{
  "id": 1167,
  "action": "updatestatus",
  "resourceType": "migration"
}
```

Bot Runner Run-as user can update the bot conversion status in the Enterprise Control Room.

View Licenses

```
{
  "id": 48,
  "action": "licenseuserallocation",
  "resourceType": "licensemanagement"
}
```

Users can view the license details for the Enterprise Control Room.

Manage user's device licenses

```
{
  "id": 20,
  "action": "licensemanagement",
  "resourceType": "licensemanagement"
}
```

Users can assign device licenses to users.

Install licenses

```
{
  "id": 49,
  "action": "licenseinstall",
  "resourceType": "licensemanagement"
}
```

Users can install Automation Anywhere Enterprise licenses from the Enterprise Control Room.

Related reference
Permissions to roles mapping

API permissions

Enable access to APIs and API related features.

API

Bot Insight Data API

```
{
  "id": 47,
  "action": "botinsightapi",
  "resourceType": "api"
}
```

Allows access to Bot Insight RESTful APIs to the data logged by the Enterprise Control Room and by a task during production runs.

Generate API-Key

```
{
  "id": 91,
  "action": "generateapikey",
  "resourceType": "api"
}
```

Users can generate an apiKey that can be used in the Authentication API. See Authenticate with username and apiKey.

Related reference
Permissions to roles mapping

IQ Bot permissions

IQ Bot provides cognitive (intelligent) automation that can learn further from humans to uncover and transform important, but less structured data to automate business processes quickly and efficiently, simultaneously reducing human error.

IQ Bot

View IQ Bot

```
{
  "id": 69,
  "action": "viewiqbot",
  "resourceType": "iqbot"
}
```

Users can view the default dashboards in the IQ Bot portal.

View learning instances from the same role

```
{
  "id": 79,
  "action": "viewlearninginstancefromsamerole",
  "resourceType": "viewlearninginstance"
}
```

Users can view learning instances created by other users with the same role in the IQ Bot portal.

View ALL learning instances

```
{
  "id": 80,
  "action": "viewalllearninginstances",
  "resourceType": "viewlearninginstance"
}
```

Users can view all learning instances in the IQ Bot portal.

Launch validator

```
{
  "id": 73,
  "action": "launchvalidatior",
  "resourceType": "viewlearninginstance"
}
```

Access IQ Bot Validator to review and update documents with exceptions.

Create learning instances

```
{
  "id": 73,
  "action": "launchvalidatior",
  "resourceType": "viewlearninginstance"
}
```

Users can access IQ Bot Validator to review and update documents with exceptions.

Edit learning instances

```
{
  "id": 75,
  "action": "editlearninginstances",
  "resourceType": "viewlearninginstance"
}
```

Users can create learning instances in the IQ Bot portal.

Permission/feature ID: 75

Delete learning instances

```
{
  "id": 76,
  "action": "deletelearninginstances",
  "resourceType": "viewlearninginstance"
}
```

Users can edit their learning instances in the IQ Bot portal.

Send Learning instances to production

```
{
  "id": 77,
  "action": "sendlearninginstancestoprod",
  "resourceType": "viewlearninginstance"
}
```

Users can send their learning instances to production in the IQ Bot portal.

Train learning instance groups

```
{
  "id": 78,
  "action": "trainlearninginstancegroups",
  "resourceType": "viewlearninginstance"
}
```

Users can train their learning instance groups in the IQ Bot portal.

Permission/feature ID: 78

View domains

```
{
  "id": 71,
  "action": "veiwdomain",
  "resourceType": "viewiqbot"
}
```

Users can view all domains in the IQ Bot portal.

1

Create domains

```
{
  "id": 81,
  "action": "createdomains",
  "resourceType": "veiwdomain"
}
```

Users can create domains in the IQ Bot portal.

Import domains

```
{
  "id": 84,
  "action": "importdomains",
  "resourceType": "veiwdomain"
}
```

Permission/feature ID: 84
Users can import domains in the IQ Bot portal.

Export domains

```
{
  "id": 85,
  "action": "exportdomains",
  "resourceType": "veiwdomain"
}
```

Users can export domains in the IQ Bot portal.

View Administration

```
{
  "id": 72,
  "action": "veiwadministration",
  "resourceType": "viewiqbot"
}
```

Users can access the Administration tab in the IQ Bot portal.

View and manage settings

```
{
  "id": 86,
  "action": "veiwandmanagesettings",
  "resourceType": "veiwadministration"
}
```

Users can manage advanced configuration settings of the IQ Bot portal.

View and manage migration

```
{
  "id": 87,
  "action": "veiwandmanagemigration",
  "resourceType": "veiwadministration"
}
```

Users can access the migration utility to export and import learning instances in the IQ Bot portal.

Related reference
Permissions to roles mapping