



# BeyondTrust

## **Remote Support Autotask Integration**

## Table of Contents

---

<b>BeyondTrust Integration with Autotask</b> .....	<b>3</b>
<b>Prerequisites for the BeyondTrust Remote Support Integration with Autotask</b> .....	<b>4</b>
Applicable Versions .....	4
Network Considerations .....	4
Prerequisite Installation and Configuration .....	4
<b>Configure Autotask for Integration with BeyondTrust Remote Support</b> .....	<b>5</b>
Create an API User .....	5
Create BeyondTrust Application .....	5
Create the BeyondTrust Jump LiveLink .....	6
Create the BeyondTrust Session Key LiveLink .....	7
<b>Configure BeyondTrust Remote Support for Integration with Autotask</b> .....	<b>8</b>
Verify That the API Is Enabled .....	8
Create an API Service Account - BeyondTrust 16.1 and Earlier .....	8
Create an API Service Account - BeyondTrust 16.2 and Later .....	9
Add an Outbound Event URL .....	9
Add a Custom Link .....	10
<b>Configure the Autotask Plugin for Integration with BeyondTrust Remote Support</b> .....	<b>11</b>
BeyondTrust Appliance .....	11
Autotask Instance .....	12
Report Templates .....	13
<b>Use Cases for the Autotask Integration with BeyondTrust Remote Support</b> .....	<b>14</b>
Generate Session Key .....	14
Import BeyondTrust Session Data into Ticket .....	14
Jump to Configuration Item .....	15
Access Ticket from Representative Console .....	15
Auto-Time Entry Creation .....	15
Automatic Ticket Creation .....	15

# BeyondTrust Integration with Autotask

 **IMPORTANT!**

*You must purchase this integration separately from both your BeyondTrust software and your Autotask solution. For more information, contact BeyondTrust sales.*

Service desks and customer support organizations using Autotask can integrate with BeyondTrust to improve service levels, centralize support processes, and strengthen compliance. This document describes the installation and configuration of the BeyondTrust Remote Support integration with Autotask.

The Autotask integration with BeyondTrust Remote Support provides the following functionality:

- A BeyondTrust support session can be initiated from within Autotask.
- At the end of a session, the incident can be updated with session information, including a chat transcript, system information, session notes, and customer and representative surveys.

# Prerequisites for the BeyondTrust Remote Support Integration with Autotask

To complete this integration, please ensure that you have the necessary software installed and configured as indicated in this guide, accounting for any network considerations.

## Applicable Versions

- BeyondTrust Remote Support: 14.x and newer
- Autotask: Current release

## Network Considerations

The following network communication channels must be open for the integration to work properly:

Outbound From	Inbound To	TCP Port #	Purpose
BeyondTrust Middleware Engine Server	Autotask	443	API calls from the BeyondTrust Middleware Engine server.
BeyondTrust Middleware Engine Server	BeyondTrust Appliance	443	API calls from the BeyondTrust Middleware Engine server.
BeyondTrust Appliance	BeyondTrust Middleware Engine Server	8180 (default) 443 (optional)	The BeyondTrust Middleware Engine server receives outbound events from the appliance. However, if polling is used instead of outbound events, then this port does not have to be open.

## Prerequisite Installation and Configuration

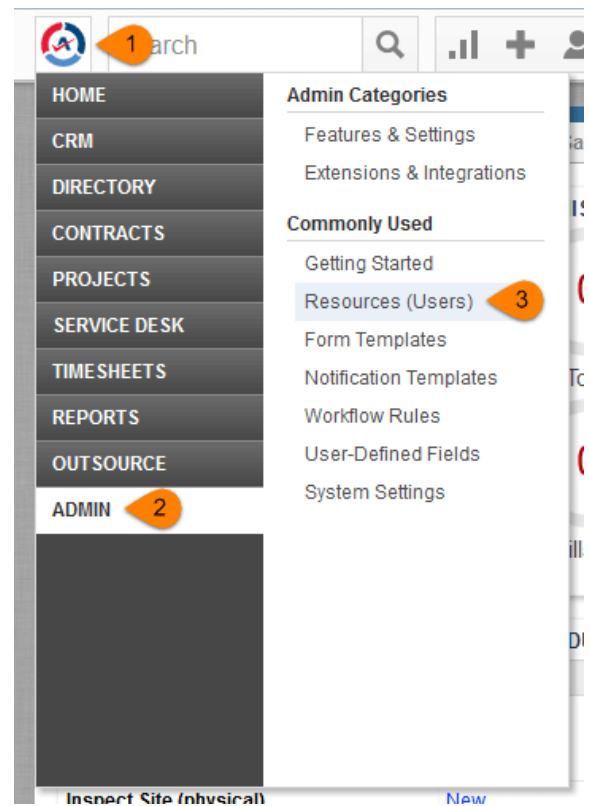
The Autotask integration is a BeyondTrust Middleware Engine plugin. To install the BeyondTrust Middleware Engine, follow the instructions in the [BeyondTrust Middleware Engine Configuration](#) document at [www.beyondtrust.com/docs/remote-support/how-to/integrations/middleware-engine](http://www.beyondtrust.com/docs/remote-support/how-to/integrations/middleware-engine).

# Configure Autotask for Integration with BeyondTrust Remote Support

The Autotask integration includes a few LiveLinks which must be created in Autotask for generating session keys and launching a Jump session to a Configuration Item.

## Create an API User

1. In Autotask, go to **Admin > Resources (Users)**.
2. Click the **New** button.
3. Create a user with the **System Administrator** security level.

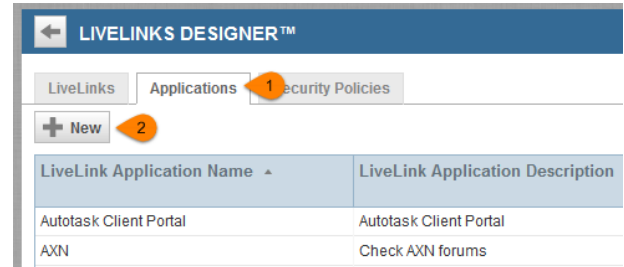


## Create BeyondTrust Application

1. In Autotask, go to **Admin > Extensions & Integrations > Autotask Add-ons > LiveLinks Designer**.

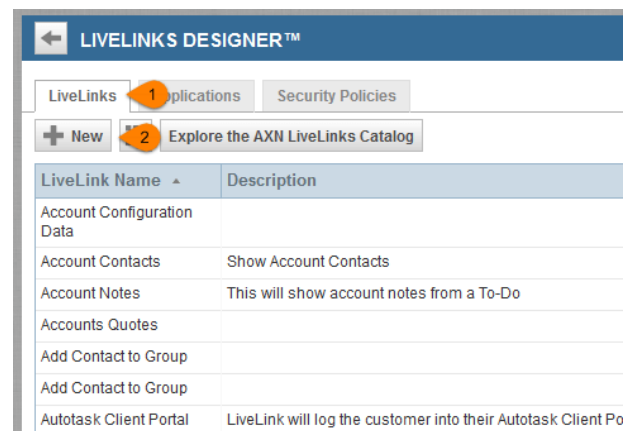


2. On the **Applications** tab, click the **New** button.
3. Enter **BeyondTrust** as the application name.
4. Click **Save and Close**.



## Create the BeyondTrust Jump LiveLink

1. In Autotask, go to **Admin > Extensions & Integrations > Autotask Add-ons > LiveLinks Designer**.
2. On the **LiveLinks** tab, click the **New** button.

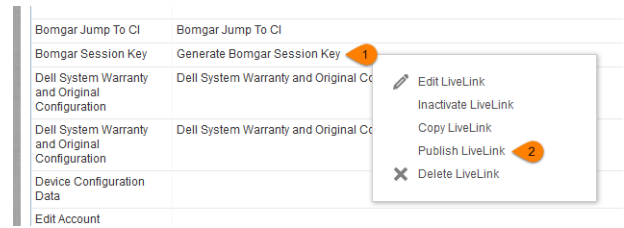


3. Create the LiveLink with the following settings:

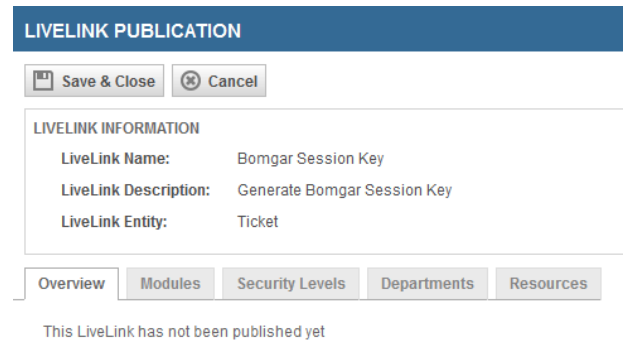
Field Name	Field Value
LiveLink Name	BeyondTrust Jump To CI
Label	BeyondTrust Jump To CI
Description	BeyondTrust Jump To CI
Active	[checked]
Category	Service Desk
LiveLink Application Name	BeyondTrust
Entity	Ticket
Base URL	https://<BeyondTrust-appliance-host>/api/client_script.ns
Querystring Values	?type=rep&operation=generate&action=start_pinned_client_session&search_string=<UDF-Device Name (ATG_InstalledProduct)>&session.custom.external_key=<TICKETID>

4. On the **Configuration** tab, ensure both **Ticket Details Window - Service Desk** and **Ticket Grid - Service Desk** are checked.

- Click **Save and Close**.
- Right-click the LiveLink in the list and select **Publish LiveLink**.



- Using the **Security Levels**, **Departments**, and/or **Resources** tabs, select the appropriate groups or users that should be able to view and use the LiveLink.
- Click **Save and Close** to publish the LiveLink.



## Create the BeyondTrust Session Key LiveLink

- Create another LiveLink with the following settings:

Field Name	Field Value
LiveLink Name	BeyondTrust Session Key
Label	BeyondTrust Session Key
Description	Generate BeyondTrust Session Key
Active	[checked]
Category	Service Desk
LiveLink Application Name	BeyondTrust
Entity	Ticket
Base URL	https://<BeyondTrust-appliance-host>/api/client_script.ns
Querystring Values	?type=rep&operation=generate&action=generate_session_key&session.custom.external_key=<TICKETID>

- On the **Configuration** tab, ensure both **Ticket Details Window - Service Desk** and **Ticket Grid - Service Desk** are checked.
- Click **Save and Close**.
- Follow the same steps as with the previous link to select the appropriate groups and users, and then publish the LiveLink.

# Configure BeyondTrust Remote Support for Integration with Autotask

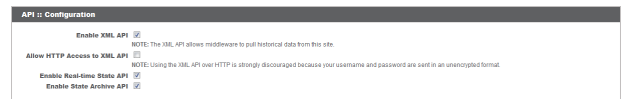
Several configuration changes are necessary on the BeyondTrust Appliance to integrate with Autotask. You must make these changes on each appliance for which you intend to create a plugin configuration, described in "[Configure the Autotask Plugin for Integration with BeyondTrust Remote Support](#)" on page 11.

All of the steps in this section take place in the BeyondTrust `/login` administrative interface. Access your BeyondTrust interface by going to the hostname of your BeyondTrust Appliance followed by `/login` (e.g., <https://support.example.com/login>).

## Verify That the API Is Enabled

This integration requires the BeyondTrust XML API to be enabled. This feature is used by the BeyondTrust Middleware Engine to communicate with the BeyondTrust APIs.

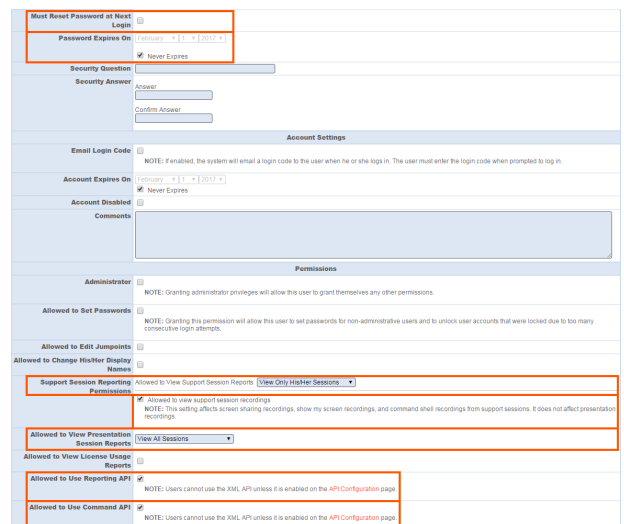
Go to `/login > Management > API Configuration` and verify that **Enable XML API** is checked.



## Create an API Service Account - BeyondTrust 16.1 and Earlier

The API user account is used from within the integration to make BeyondTrust Command API calls to BeyondTrust.

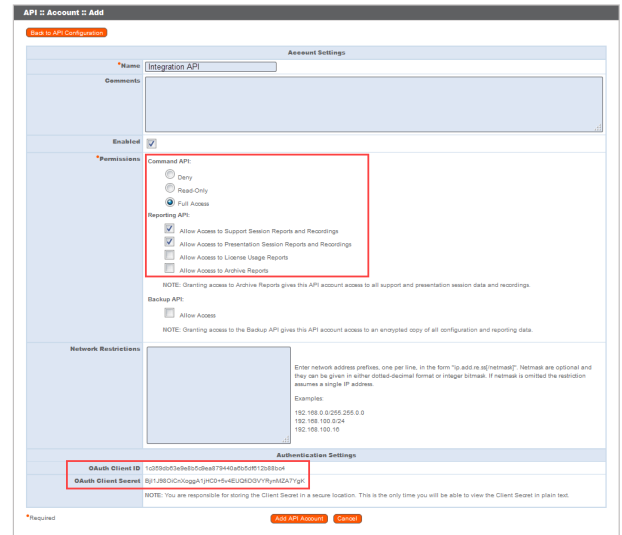
1. Go to `/login > Users & Security > Users`.
2. Click **Create New User** and name it **Integration** or something similar.
3. Leave **Must Reset Password at Next Login** unchecked.
4. Set **Password Expires On** to **Never Expires**.
5. Set **Allowed to View Support Session Reports** to **View All Sessions**.
6. Check **Allowed to view support session recordings**.
7. Set **Allowed to View Presentation Session Reports** to **View All Sessions**.
8. Check **Allowed to Use Reporting API** and **Allowed to Use Command API**.
9. Scroll to the bottom and save the account.





## Create an API Service Account - BeyondTrust 16.2 and Later

1. Go to **Management > API Configuration** and create a new API account.
2. Under **Permissions**, check **Full Access** to the **Command API**.
3. For the **Reporting API**, check **Allow Access to Support Session Reports and Recordings** and **Allow Access to Presentation Session Reports and Recordings**.
4. Be sure to copy the values for both the **OAuth Client ID** and **OAuth Client Secret** for use in a later step.



5. Click **Add API Account** to create the account.

## Add an Outbound Event URL

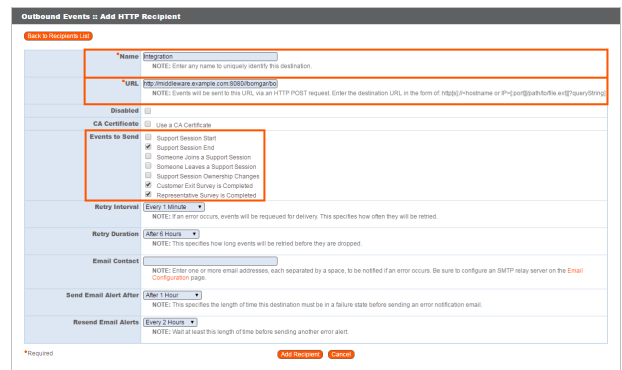
1. Go to **/login > Management > Outbound Events**.
2. Click **Add New HTTP Recipient** and name it **Integration** or something similar.
3. Enter the URL to use:

- If using an appliance ID of "default":  
`http://<middleware-host>:<port>/ERSPost`. The default port is 8180.
- If using an appliance ID other than "default":  
`http://<middleware-host>:<port>/ERSPost?appliance=<appliance-id>` where `<middleware-host>` is the hostname where the BeyondTrust Middleware Engine is installed. The default port is 8180. The `<appliance-id>` is an arbitrary name, but note the value used, as it is entered later in the plugin configuration. This name accepts only alphanumeric values, periods, and underscores.

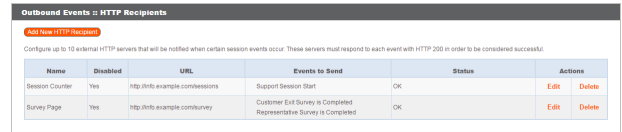
4. Scroll to **Events to Send** and check the following events:

- **Support Session End**
- **Customer Exit Survey is Completed**
- **Representative Survey is Completed**
- **Someone Joins a Support Session** (Optional)

5. Scroll to the bottom and click **Add Recipient**.



- Now, the list of outbound events should contain the event just added. The **Status** column displays a value of **OK** if communication is working. If communication is not working, the **Status** column displays an error which you can use to repair communication.



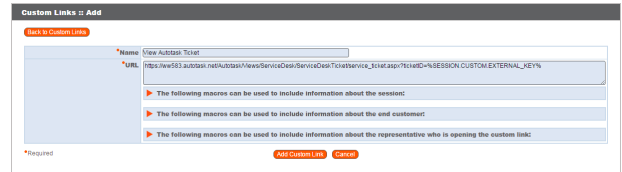
Name	Disabled	URL	Events to Send	Status	Actions
Session Counter	Yes	http://info.example.com/compsessions	Support Session Start	OK	Edit Delete
Survey Page	Yes	http://info.example.com/survey	Customer Exit Survey is Completed Representative Survey is Completed	OK	Edit Delete

## Add a Custom Link

BeyondTrust custom links can be configured to allow representatives to quickly access the Autotask item that is associated with the session.

### BeyondTrust 15.1 and later

- Browse to **Rep Console > Custom Links**.
- Under **Rep Console :: Custom Links**, click **Create New Custom Link**.
- Enter a name for the link, and then set the URL to `https://ww<#>.autotask.net/Autotask/Views/ServiceDesk/ServiceDeskTicket/service_ticket.aspx?ticketID=%SESSION.CUSTOM.EXTERNAL_KEY%` where `<#>` is the customer's Autotask instance number. If needed, you can use any of the available macros to customize the link according to your specifications.
- Click **Add Custom Link** to save the new link.



Custom Links :: Add

**Name** View Autotask Ticket

**URL** https://ww<#>.autotask.net/Autotask/Views/ServiceDesk/ServiceDeskTicket/service\_ticket.aspx?ticketID=%SESSION.CUSTOM.EXTERNAL\_KEY%

The following macros can be used to include information about the session:

The following macros can be used to include information about the end customer:

The following macros can be used to include information about the representative who is opening the custom link:

\*Required

**Add Custom Link** **Cancel**

### BeyondTrust 15.0 and earlier

- Browse to **Configuration > Options > Representative Console :: Custom Link**.
- Set the URL to `https://ww<#>.autotask.net/Autotask/Views/ServiceDesk/ServiceDeskTicket/service_ticket.aspx?ticketID=%SESSION.CUSTOM.EXTERNAL_KEY%` where `<#>` is the customer's Autotask instance number.
- Click **Save**.

# Configure the Autotask Plugin for Integration with BeyondTrust Remote Support

Now that you have configured Autotask and the BeyondTrust Appliance, deploy and configure the Autotask plugin. For additional information about working with the BeyondTrust Middleware Engine, please see the [BeyondTrust Middleware Engine Configuration](http://www.beyondtrust.com/docs/remote-support/how-to/integrations/middleware-engine) document at [www.beyondtrust.com/docs/remote-support/how-to/integrations/middleware-engine](http://www.beyondtrust.com/docs/remote-support/how-to/integrations/middleware-engine).

1. Copy the provided plugin zip file to the server hosting the BeyondTrust Middleware Engine.
2. Extract the plugin zip file to the **Plugins** folder in the directory where the BeyondTrust Middleware Engine is installed.
3. Restart the BeyondTrust Middleware Engine Windows service.
4. From the server, launch the middleware administration tool. The default URL is **http://127.0.0.1:53231**.
5. The **Autotask Plugin** shows in the list of plugins. Click the clipboard icon to add a new configuration.

## BeyondTrust Appliance

The first portion of the plugin configuration provides the necessary settings for communication between the plugin and the BeyondTrust Appliance. The configuration sections include:

1. **Plugin Configuration Name:** Any desired value. Because multiple configurations can be created for a single plugin, allowing different environments to be targeted, provide a descriptive name to indicate how this plugin is to be used.
2. **Appliance Id:** This can be left as **Default** or can be given a custom name. This value must match the value configured on the outbound event URL in the BeyondTrust Appliance. If outbound events are not being used, this value is still required, but any value may be used.
3. **BeyondTrust Appliance Host Name:** The hostname of the BeyondTrust Appliance. Do not include `https://` or other URL elements.
4. **BeyondTrust Integration API OAuth Client ID:** When using API accounts in BeyondTrust Remote Support 16.2.1 or newer, this field should contain the Client ID of the OAuth account.
5. **BeyondTrust Integration API OAuth Client Secret:** When using API Accounts available in BeyondTrust Remote Support 16.2.1 or newer, this field should contain the client Secret of the OAuth account.
6. **BeyondTrust Integration API User Name:** If using a BeyondTrust Remote Support version prior 16.2.1, this field should contain the username of the API service account created on the BeyondTrust Appliance.
7. **BeyondTrust Integration API Password:** If using a BeyondTrust Remote Support version prior 16.2.1, this field should contain the password of the above user.
8. **Locale Used for BeyondTrust API Calls:** This value directs the BeyondTrust Appliance to return session data in the specified language.
9. **Disabled:** Enable or disable this plugin configuration.



The screenshot shows the configuration form for the Autotask plugin. It includes fields for Plugin Configuration Name (QA Environment), Appliance Id (default), Bomgar Appliance Host Name (support.example.com), Bomgar Integration API OAuth Client ID (5d73a300-3c38fce77a8940549b00f1c8910), Bomgar Integration API OAuth Client Secret, Bomgar Integration API User Name, Bomgar Integration API Password, and Locale Used for Bomgar API Calls (English). There are also checkboxes for Disabled, Allow Invalid Certificates, Use Non-TLS Connections, and Outbound Event Types (Support Session End, Customer Exit Survey is Completed, Representative Survey is Completed, Someone Joins a Support Session). A Polling Interval field is set to 1 minute.

10. **Allow Invalid Certificates:** Leave unchecked unless there is a specific need to allow. If enabled, invalid SSL certificates are allowed in calls performed by the plugin. This would allow, for example, self-signed certificates. This is not recommended in production environments.
11. **Use Non-TLS Connections:** Leave unchecked unless it is the specific goal to use non-secure connections to the BeyondTrust Appliance. If checked, TLS communication is disabled altogether. If non-TLS connections are allowed, HTTP access must be enabled on the BeyondTrust **/login > Management > API Configuration** page. Using non-secure connections is discouraged.



*Note: When using OAuth authentication, TLS cannot be disabled.*

12. **Outbound Events Types:** Specify which events the plugin processes when received by the middleware engine. Keep in mind that any event types selected here must also be configured to be sent in BeyondTrust. The middleware engine receives any events configured to be sent in BeyondTrust but passes them off to the plugin only if the corresponding event type is selected in this section.
  - a. **Support Session End**
  - b. **Customer Exit Survey is Completed**
  - c. **Representative Survey is Completed**
13. **Polling Event Types:** If network constraints limit connectivity between the BeyondTrust Appliance and the middleware engine such that outbound events cannot be used, an alternative is to use polling. The middleware engine regularly polls the BeyondTrust Appliance for any sessions that have ended since the last session was processed. At this time, only the **Support Session End** event type is supported.



*Note: One caveat to polling behavior versus the use of outbound events is that if a session has ended but the customer exit survey has not yet been submitted within the same polling interval, the customer exit survey is not processed. This does not apply to representative surveys since the session is not considered to be complete if a representative survey is still pending.*

14. **Polling Interval:** Enter only if polling is used. This determines how often the middleware engine polls the BeyondTrust Appliance for sessions that have ended.
15. **Retry Attempt Limit:** Enter the number of retries that can be attempted if the plugin fails to process an event.
16. **Retry Outbound Event Types:** Specify which outbound events the plugin retries if it fails to process the event.
17. **Retry Polling Event Types:** Specify which polling events the plugin retries if it fails to process the event.

## Autotask Instance

The remainder of the plugin configuration provides the necessary settings for communication between the plugin and the Autotask instance. The configuration settings include:

1. **Autotask Services URL:** The services URL for the Autotask instance (e.g., `https://web-services<#>.autotask.net/at-services/1.5/atws.asmx`, where `<#>` is the same number found in the URL used to access the Autotask web interface (e.g., `https://ww<#>.autotask.net`).


2. **Autotask Username:** The username of the API account.
3. **Autotask Password:** The password of the above user.
4. **Autotask Note Type:** The type of note used for BeyondTrust Session information (default value is 1).
5. **Autotask Update Time Entry:** Choose if you want the integration to create time entries based on the session duration and primary representative.
6. **Autotask Publish Notes as Public:** If enabled, notes are published as public notes, meaning that they can be seen by end-users.
7. **Time Entry Default Data:** Only used if **Autotask Update Time Entry** is enabled. This setting contains a JSON string used to represent default data passed for certain fields when creating the time entry. Most can be left 'null', but a few are required.

The screenshot shows the configuration interface for the Autotask integration. It includes the following sections:

- Autotask Services URL:** A dropdown menu with the value "https://webservice2.autotask.net/autotask/1.5/obv.asmx".
- Autotask Username:** A text input field with the value "user@domain.com".
- Autotask Password:** A password input field with a masked value.
- Autotask Note Type:** A dropdown menu with the value "1".
- Autotask Publish Notes as Public:** A checkbox that is currently unchecked.
- Autotask Update Time Entry:** A checkbox that is currently checked.
- Time Entry Default Data:** A text area containing a JSON string:
 

```
{
  "roleName": "Sample Role",
  "summaryNotes": "Sample Session",
  "taskId": null,
  "internalSessionCodeID": null,
  "type": null,
  "date": null,
  "internalNotes": null,
  "internalNotes": null,
  "internalNotes": null,
  "internalNotes": null,
  "contractID": null,
  "contractID": null
}
```

- a. **Enable Automatic Ticket Creation on Session Start:** When a session starts that is not already tied to an existing ticket, a ticket is automatically created.


 **Note:** This setting requires an external key to be formatted in a JSON string when the representative first joins the session and also requires the plugin to accept the **Someone Joins a Support Session** outbound event.

- c. **Ticket Default Data:** This setting should be used only if automatic ticket creation is enabled. This setting contains a JSON string used to represent the default data passed for certain fields when creating a ticket. Any values provided in the external key override the corresponding default value provided here.
- d. **defaultDomain:** To identify a user in Autotask which matches the primary representative in the session, the BeyondTrust username is combined with this default domain to construct an email address. The email address should match a user in Autotask.
- e. **roleName:** This is a required field for time entry creation. The supplied role name must be a role in which any common BeyondTrust/Autotask users are members.
- f. **summaryNotes:** This is also a required field for time entry creation. The value supplied is used for all sessions.

After saving the configuration, click the test icon next to the new plugin configuration. No restart is needed.

## Report Templates

On the BeyondTrust Middleware Engine server, in the `<install dir>\Plugins\<integration>\Templates` folder, there are multiple files ending with `*.hbs`. These files are used by the plugin to format the textual session report and exit surveys that are added to the corresponding ticket each time a BeyondTrust session ends or each time a survey is submitted. The templates can be edited if desired.

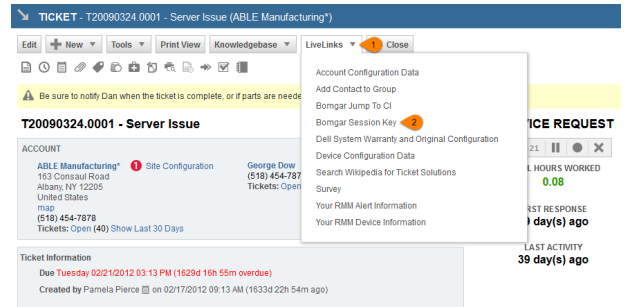
 **Note:** If changes need to be made to a template, it is a good idea to first back up the original in case the changes ever need to be reverted.

For additional information on Handlebars templates, see [handlebarsjs.com](http://handlebarsjs.com).

# Use Cases for the Autotask Integration with BeyondTrust Remote Support

## Generate Session Key

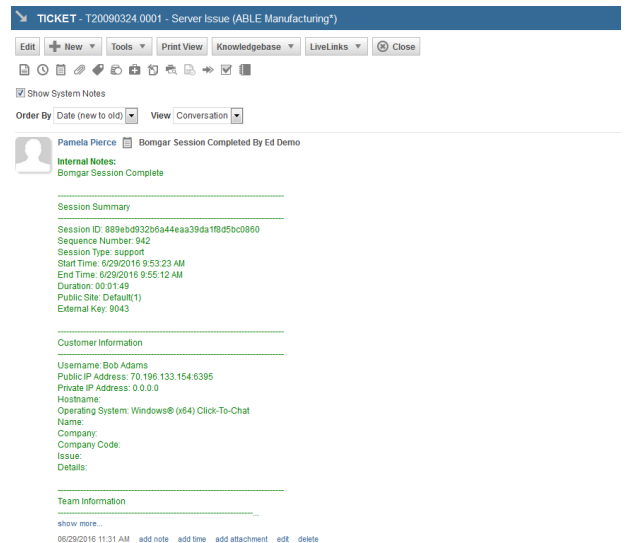
Support staff can generate a session key that can be given to the end user over the phone or via email to initiate a support session that is automatically associated with the selected ticket.



## Import BeyondTrust Session Data into Ticket

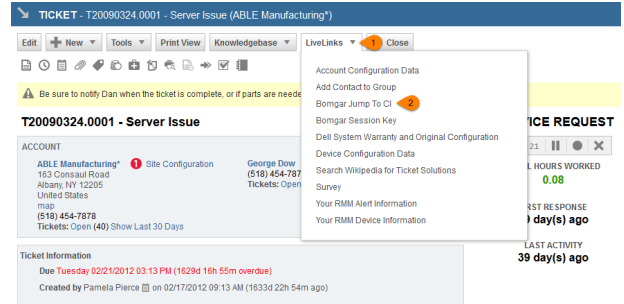
Once the session ends, the ticket is automatically updated with information gathered during the session, including:

- **Chat Transcript** (including files transferred, special actions, and other events)
- **System Information** (the General section plus other select details such as disk, memory, and network)
- **Session Notes**
- **Surveys** (customer and representative)



## Jump to Configuration Item

Support staff can leverage BeyondTrust Jump Technology to access a configuration item associated with a ticket directly from the Autotask ticket.

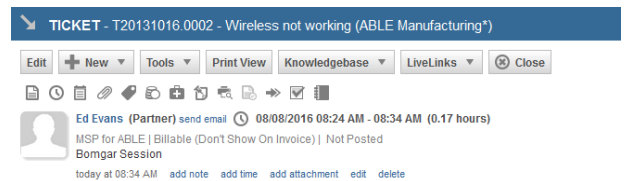


## Access Ticket from Representative Console

Using BeyondTrust's custom links ability, a representative can access the associated ticket directly from within the representative console. This saves time searching for the ticket in Autotask and provides the representative with any issue details, history, or other context to help quickly resolve the issue.

## Auto-Time Entry Creation

If desired, a time entry can be created and added to the associated ticket once a session ends. The time entry uses the same start and end time as the actual session for accurate time tracking, as well as associating it with the primary representative from the session.



## Automatic Ticket Creation

The automatic ticket creation feature does require a services engagement to implement for each customer. However, the fact that this functionality is even possible without a large custom development effort can be of great benefit to simplifying workflows and reducing the number of clicks necessary for a support representative.