



BeyondTrust

Remote Support CA Service Desk Integration

Table of Contents

BeyondTrust Remote Support Integration with CA Service Desk	3
Prerequisites for the BeyondTrust Remote Support Integration with CA Service Desk ..	4
Applicable Versions	4
Network Considerations	4
Prerequisite Installation and Configuration	4
Configure CA Service Desk for Integration with BeyondTrust Remote Support	5
Create New Tables	5
Implement Web Customizations	7
Add Web UI Elements	9
Configure BeyondTrust for the CA Service Desk Integration	13
Verify That the API Is Enabled	13
Create an API Service Account - BeyondTrust 16.1 and Earlier	13
Create an API Service Account - BeyondTrust 16.2 and Later	14
Add an Outbound Event URL	14
Configure the CA Service Desk Plugin for Integration with BeyondTrust Remote Support	16
Secure Remote Access Appliance	16
CA Service Desk Instance	17
Report Templates	18
Use Cases for the CA Service Desk Integration with BeyondTrust Remote Support	19
Generate Session Key	19
Import BeyondTrust Session Data into Ticket	19
Jump to Configuration Item	20
Click-to-Chat for Self Service Users	20
Auto-Ticket Creation	20

BeyondTrust Remote Support Integration with CA Service Desk

 **IMPORTANT!**

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

Service desks and customer support organizations using CA Service Desk 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 CA Service Desk.

The integration between CA Service Desk and BeyondTrust Remote Support enables you to initiate a request for a support session from within your support portal as an analyst or employee, providing secure remote support capabilities for your end user community. Additionally, all aspects of the remote support session can be captured directly within your CA Service Desk support ticket, offering information to the analyst and end user in order to properly diagnose, troubleshoot, and resolve user issues.

Prerequisites for the BeyondTrust Remote Support Integration with CA Service Desk

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
- CA Service Desk: 12.9 or 14.1 (plus any associated cumulative patches and test fixes)

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	CA Service Desk	443	API calls from the BeyondTrust Middleware Engine server.
BeyondTrust Middleware Engine Server	Secure Remote Access Appliance	443	API calls from the BeyondTrust Middleware Engine server.
Secure Remote Access 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 CA Service Desk 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. The BeyondTrust Middleware Engine Configuration document is also useful to understand how to work with the middleware engine.

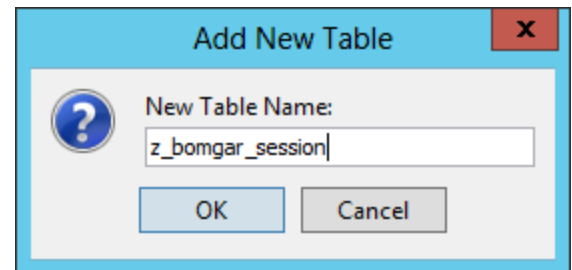
Configure CA Service Desk for Integration with BeyondTrust Remote Support

! IMPORTANT!

Before beginning the installation, please ensure you have backed up your CA Service Desk primary and secondary servers.

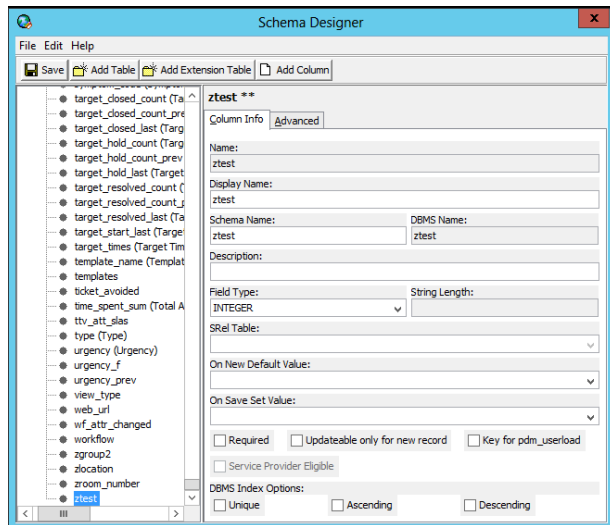
Create New Tables

1. Open the **Web Screen Painter** and log in as an administrator.
2. From within the Web Screen Painter, open **Tools > Schema Designer**.
3. Select **Add Table**, and create a table named **z_bomgar_session**.
4. Click **OK**.
5. The table info should now be displayed. Fill in the form with the following information:
 - a. **Name:** z_bomgar_session
 - b. **Display Name:** BeyondTrust Session
 - c. **Schema Name:** z_bomgar_session
 - d. **Description:** Fac_Attr_Entry z_bomgar_session
 - e. **Default Display Field:** bgr_session_id
 - f. **Foreign Key Field:** id



6. Click **Save**.
7. After creating and saving the table with the required basic information, click **Add Column** to add the following columns to the table:

Schema Name	Display Name	Field Type	SREL Table or Length	Required
analyst	analyst	SREL	SREL Table: cnt	No
bgr_session_id	bgr_session_id	String	100	Yes
cr_persid	cr_persid	SREL	SREL Table: cr	Yes
end_date_time	end_date_time	Date	n/a	Yes
recording_url	recording_url	String	1000	No
session_duration	session_duration	Duration	n/a	No
start_date_time	start_date_time	Date	n/a	No
support_session_detail	support_session_detail	String	30000	Yes



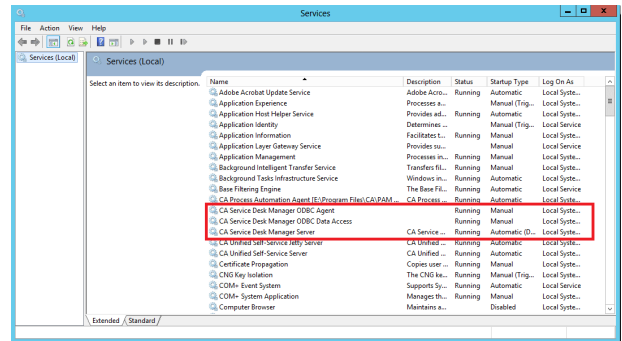
8. Once all columns have been created, select **File > Save and Publish**. When prompted to continue, click **Yes**.
9. Once again, select **Add Table**, and create a table named **zbgr_connection_type**.
10. Click **OK**.
11. The table info should now be displayed. Fill in the form with the following information:
 - a. **Name:** zbgr_connection_type
 - b. **Display Name:** BeyondTrust Connection Type
 - c. **Schema Name:** zbgr_connection_type
 - d. **Description:** Fac_Attr_Entry zbgr_connection_type
 - e. **Default Display Field:** connection_name
 - f. **Foreign Key Field:** id
12. Click **Save**.
13. After creating and saving the table with the required basic information, click **Add Column** to add the following columns to the table:

Schema Name	Display Name	Field Type	SREL Table or Length	Required
connection_name	Connection_name	String	100	Yes
default_connection_type	Default_connection_type	Integer	n/a	No
delete_flag	SRel_Attr_Entry zbgr_connection_type.delete_flag	SREL	SREL Table: Actbool	No
Description	Description	String	250	Yes
zjumpoint_required	Zjumpoint_required	SREL	SREL Table: bool (Boolean)	No
14. Once all columns have been created, select **File > Save and Publish**. When prompted to continue, click **Yes**.
15. Navigate to the **nr** table within Schema Designer.

16. Select **Add Column** and add the following fields to this table:

Schema Name	Display Name	Field Type	SREL Table or Length	Required
zbgr_conn_type	zbgr_conn_type	SREL	SREL Table: zbgr_connection_type	No
zbgr_jpoint	zbgr_jpoint	String	200	No

17. Once all columns have been created, select **File > Save and Publish**. When prompted to continue, click **Yes**.
18. Ensure all users are out of the system, and then shut down CA Service Desk services on all servers.
19. Open an administrative command prompt window and run the command **pdm_publish**.



Implement Web Customizations

1. If not already stopped, stop CA Service Desk services.
2. Browse to the directory where CA Service Desk is installed (e.g., **C:\Program Files (x86)\CA\Service Desk Manager**).
3. Copy the components package (**CA_Service_Management_Integration_Components.zip**) to the primary server and extract the contents to **\patches**.



Note: The **patches** folder is not created during CA Service Desk installation and may need to be manually created.

4. Locate **detail_in.html** in **.\patches\CA_Service_Management_Integration_Components\CA_ServiceDesk_FrontEnd_Code\site\mods\www\html\weblanalyst** and open the file in a text editor such as Notepad.
5. Open **detail_in.html** in **.\site\mods\www\html\weblanalyst** in a text editor.



Note: If you have not previously customized your Analyst Incident interface, you will not find a **detail_in.html** file located in **.\site\mods\www\html\weblanalysts**. In this case, you need to copy the **detail_in.html** file from **.\bopcfg\www\html\weblanalyst** to **.site\mods\www\html\weblanalyst**.

6. Within the first file, from the **patches** folder, there are six code snippets that must be copied to the corresponding location in the second file. These snippets are surrounded by lines which read `<!-- Integral Customization -- Start -->` and `<!-- Integral Customization -- End -->`. Copy the code between those lines and paste them in the same locations in the second file.



Note: Do NOT copy the "Start" and "End" lines.

7. Save the file when complete.

**IMPORTANT!**

*If you have multiple Analyst type form groups, make sure to apply the changes to each **detail_in.html** form located in each respective Analyst type form group. An Analyst type form group is any sub-folder underneath **.site\mods\www\html\web\analyst**.*

8. Copy the following files from **.patches\CA_Service_Management_Integration_Components\CA_ServiceDesk_FrontEnd_Code\site\mods\www\html\web\analyst** to **.site\mods\www\html\web\analyst**:
 - detail_z_bomgar_session.html
 - detail_zbgr_connection_type.html
 - list_z_bomgar_session.html
 - list_zbgr_connection_type.html
9. Repeat steps 4-7 above but for **cmdb_detail.html** in **.patches\CA_Service_Management_Integration_Components\CA_ServiceDesk_FrontEnd_Code\site\mods\www\html\web\analyst** as the source file and **.site\mods\www\html\web\employee** as the destination. There should be two snippets that must be copied and placed in the destination file.
10. Repeat steps 4-7 above but for **nr_cmdb_har_worx_tab.html** in **.patches\CA_Service_Management_Integration_Components\CA_ServiceDesk_FrontEnd_Code\site\mods\www\html\web\analyst** as the source file and **.site\mods\www\html\web\employee** as the destination. There should be one snippet that must be copied and placed in the destination file.
11. Repeat steps 4-7 above but for **detail_in.html** in **.patches\CA_Service_Management_Integration_Components\CA_ServiceDesk_FrontEnd_Code\site\mods\www\html\web\employee** as the source file and **.site\mods\www\html\web\employee** as the destination. There should be three snippets that must be copied and placed in the destination file.
12. Copy **start_session.js** from **.patches\CA_Service_Management_Integration_Components\CA_ServiceDesk_FrontEnd_Code\site\mods\www\wwwroot\scripts**.
13. Copy **Bomgar.png** from **.patches\CA_Service_Management_Integration_Components\CA_ServiceDesk_FrontEnd_Code\site\mods\www\wwwroot\img** to **.site\mods\www\wwwroot\img**.
14. Open **.ns.env** in a text editor and add the following line: `@NX_bomgar_HOST=<BeyondTrust_Host>` where `<BeyondTrust_Host>` is the hostname of your BeyondTrust site (e.g., support.example.com).
15. Start the CA Service Desk Manager services.
16. Open an administrative command prompt window, change directories to **.patches\CA_Service_Management_Integration_Components\CA_ServiceDesk_FrontEnd_Code**, and run the command to load data into the **zbgr_connection_type** table:
`Pdm_load -i f zbgr_connection_type.txt.`
17. Verify that all files copied in this section have been copied to each primary and secondary server. Log into each primary and secondary CA Service Desk server and verify that the web changes in this section have been replicated in the following files in these locations:
 - **.site\mods\www\html\web\analyst**
 - detail_in.html
 - detail_z_bomgar_session.html
 - detail_zbgr_connection_type.html

- list_z_bomgar_session.html
- list_zbgr_connection_type.html
- .\site\mods\www\html\web\employee
 - detail_in.html
- .\site\mods\www\wwwroot\scripts
 - start_session.js
- .\site\mods\www\wwwroot\img
 - Bomgar.png



Note: If any file does not exist in these locations on each primary and secondary server, copy the files from the server you made changes on to the primary/secondary servers in the above locations.

Add Web UI Elements

1. Log into CA Service Desk as an administrator.
2. Go to **Administration > Security and Role Management > Role Management > Menu Trees**.
3. Within the **Menu Trees** list, select **admin_tree**.



Note: If your admin tree is already customized, skip steps 4-6 and select your custom admin tree.

4. Select **File > Copy**.
5. Fill in the following values:

a. **M
e
n
u**

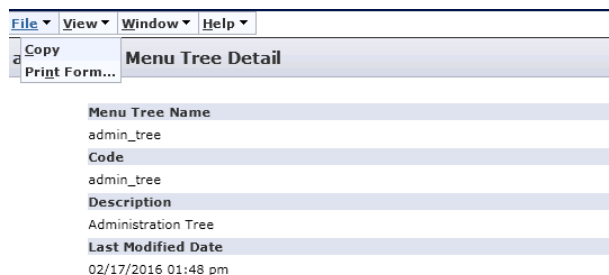
T

ree Name: custom_admin_tree

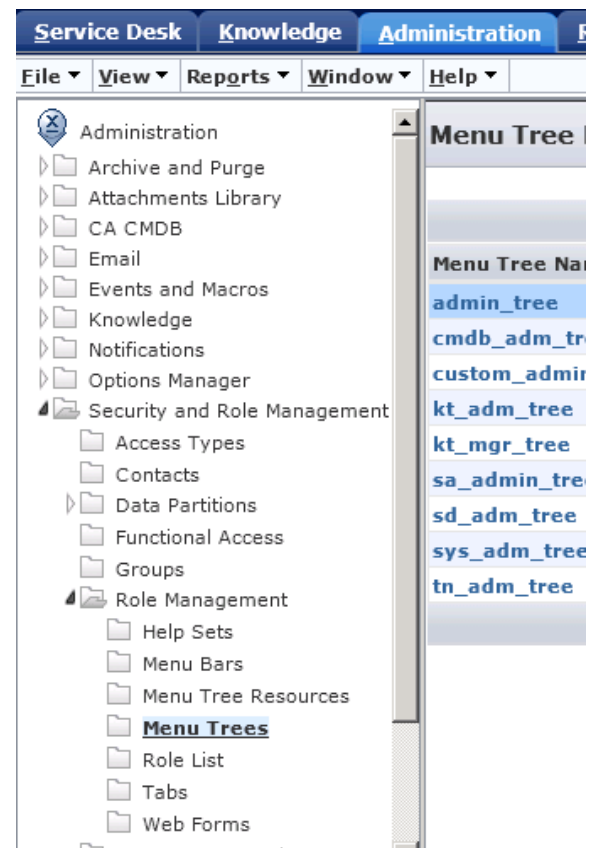
b. **Code:** cat

c. **Internal:** No

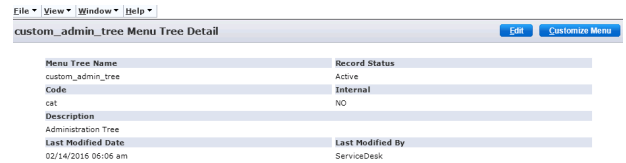
d. **Description:** Administration Tree



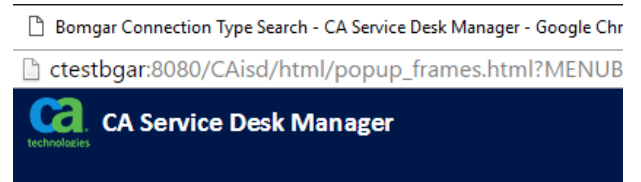
Menu Tree Name
admin_tree
Code
admin_tree
Description
Administration Tree
Last Modified Date
02/17/2016 01:48 pm



6. Fill in the following values:
 - a. **Menu Tree Name:** custom_admin_tree
 - b. **Code:** cat
 - c. **Internal:** No
 - d. **Description:** Administration Tree
7. Click **Save**.
8. Select **Customize Menu**.

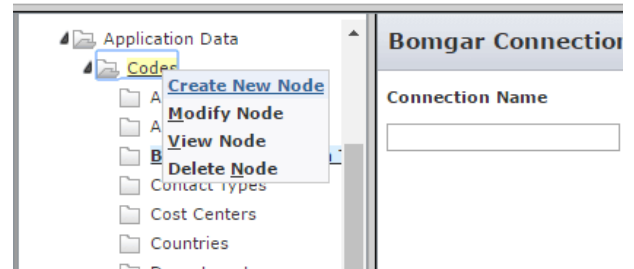


9. Within **ServiceDesk > Application Data > Codes**, right-click **Codes** and click **Create New Node**.
10. Create a new node with the following information:
 - a. **Node Name:** BeyondTrust Connection Type
 - b. **Resource:** BeyondTrust Connection Type

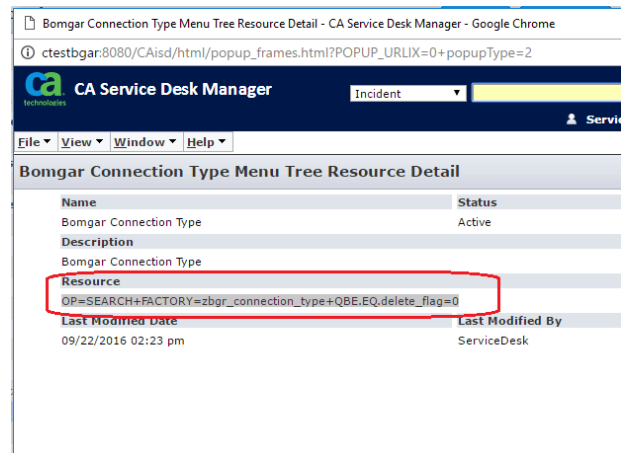
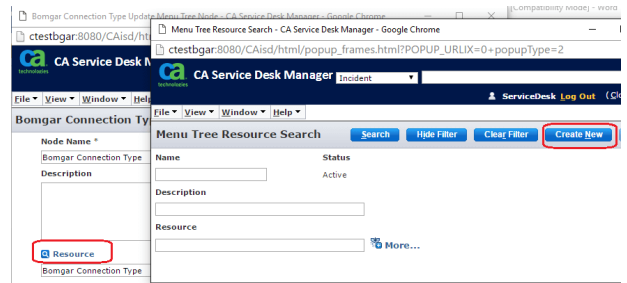


Note: If you do not have a BeyondTrust Connection Type resource, follow step 11 to create one. Otherwise, skip to step 12 once you have saved the record.

11. Click **Save**.

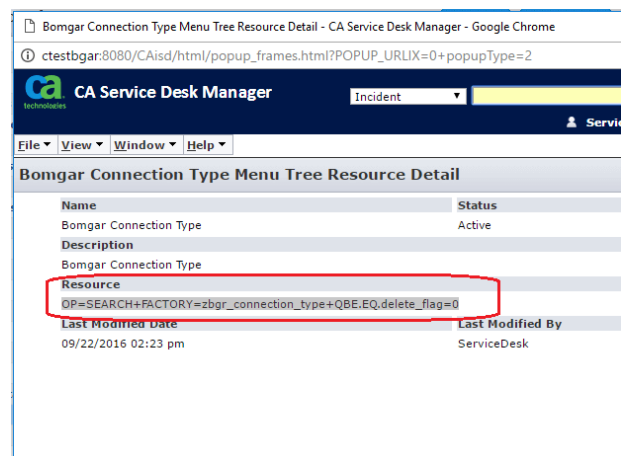
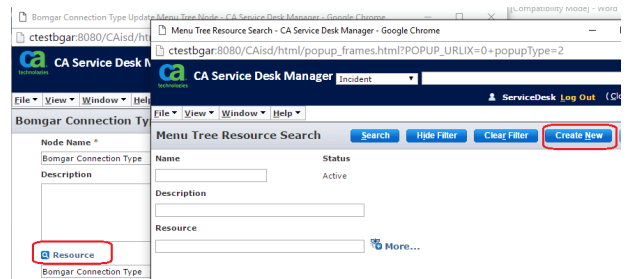


12. Click **Save**.



13. To create a new resource, click the blue **Resource** link, and select the **Create New** button from the displayed screen. Fill in the following information to create the BeyondTrust Connection Type resource:

- Name:** BeyondTrust Connection Type
- Description:** BeyondTrust Connection Type
- Resource:** OP=SEARCH+FACTORY=zbgr_connection_type+QBE.EQ.delete_flag=0



14. Go to **Administrator > Security and Role Management > Role Management > Tabs**.
15. Click on **Administration tab with full menu**.
16. Select the **Administration** link from the starting page and edit the **Administration Web Form**.
17. Modify the **Resource** section to add the following to the end of the existing value: `+KEEP.tree_code=cat`.
18. Restart the CA Service Desk services.

Administration Web Form Detail	
Web Form Name	Record Status
Administration	Active
Code	Type
admin	HTML
Description	
Administration Form	
Resource	
\$qg?SID=\$SESSION.SID+fid=123+OP=DISPLAY_FORM+HTML=admin_main_role.html+KEEP.tree_code=cat	
Last Modified Date	Last Modified By
02/14/2016 06:16 am	ServiceDesk

Configure BeyondTrust for the CA Service Desk Integration

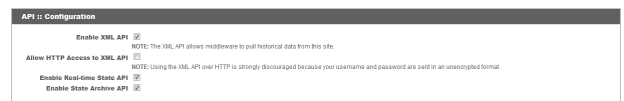
Several configuration changes are necessary on the Secure Remote Access Appliance to integrate with CA Service Desk. You must make these changes on each appliance for which you intend to create a plugin configuration, described in "[Configure the CA Service Desk Plugin for Integration with BeyondTrust Remote Support](#)" on page 16.

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 Secure Remote Access 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.

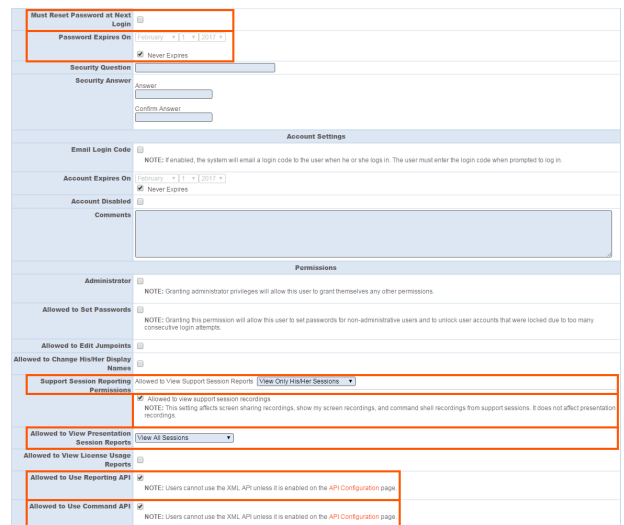
Go to **/login > Management > API Configuration** and verify that **Enable XML API** is checked.



Create an API Service Account - BeyondTrust 16.1 and Earlier

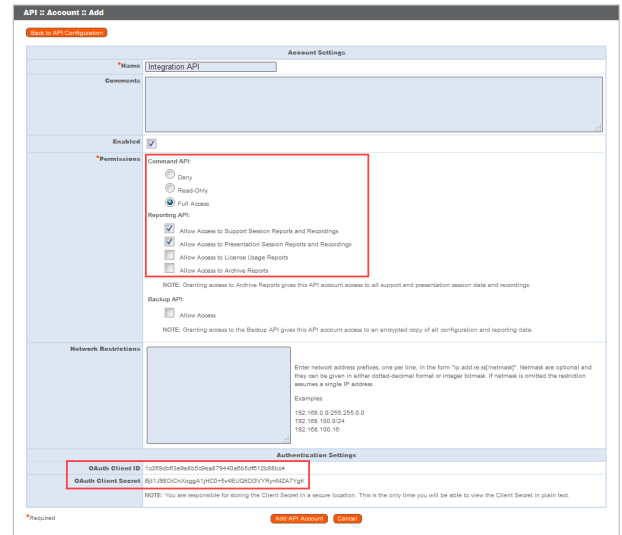
The API user account is used from within the integration to make BeyondTrust Command API calls to BeyondTrust.

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

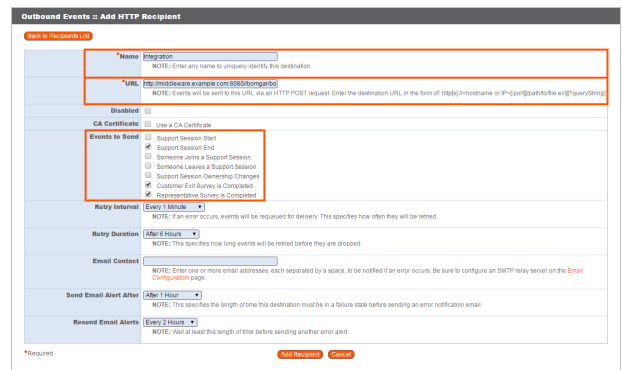
1. Go to **/login > Management > Outbound Events**.
2. Click **Add New HTTP Recipient** and name it **Integration** or something similar.
3. Enter the URL to use:

- 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**.



6. 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.

Outbound Events :: HTTP Recipients

[Add New HTTP Recipient](#)

Configure up to 10 external HTTP servers that will be notified when certain session events occur. These servers must respond to each event with HTTP 200 in order to be considered successful.

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

Configure the CA Service Desk Plugin for Integration with BeyondTrust Remote Support

Now that you have configured CA Service Desk and the Secure Remote Access Appliance, deploy and configure the CA Service Desk 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.

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 **CA Service Desk Plugin** shows in the list of plugins. Click the clipboard icon to add a new configuration.

Secure Remote Access Appliance

The first portion of the plugin configuration provides the necessary settings for communication between the plugin and the Secure Remote Access 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 Secure Remote Access Appliance. If outbound events are not being used, this value is still required, but any value may be used.
3. **Secure Remote Access Appliance Host Name:** The hostname of the Secure Remote Access 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 Secure Remote Access 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 Secure Remote Access 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 CA Service Desk Plugin. The fields are as follows:

- Plugin Configuration Name:** QA Environment
- Appliance Id:** default
- Bonggar Appliance Host Name:** support.example.com
- Bonggar Integration API OAuth Client ID:** 5d73d30c3c38fce77ab94d549b20f1c8910
- Bonggar Integration API OAuth Client Secret:** [Redacted]
- Bonggar Integration API User Name:** [Redacted]
- Bonggar Integration API Password:** [Redacted]
- Locale Used for Bonggar API Calls:** English
- Disabled:** Disabled
- Outbound Event Types:**
 - Support Session End
 - Customer Exit Survey is Completed
 - Representative Survey is Completed
 - Someone Joins a Support Session
- Polling Event Types:**
 - Support Session End
- Polling Interval (in minutes):** [Redacted]

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 Secure Remote Access 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 Secure Remote Access Appliance and the middleware engine such that outbound events cannot be used, an alternative is to use polling. The middleware engine regularly polls the Secure Remote Access 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 Secure Remote Access 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.

CA Service Desk Instance

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

1. **CA Service Desk Services URL:** The services URL for the CA Service Desk instance (e.g., `https://caservicedesk.example.com/axis/services/USD_R11_WebService`)
2. **CA Service Desk Username:** The username of the API account.
3. **CA Service Desk Password:** The password of the above user.
4. **Enable Automatic Incident Creation on Session Start (Rep joins session):** If checked, the plugin processes `support_conference_member_added` events and the external key to determine whether to create a ticket within CA Service Desk or not. The plugin attempts to create the ticket only if this setting enabled, if the conference member joining the conference is a representative, and if the external key is either a JSON string or the literal value **CHAT**.
5. **Enable Automatic Incident Creation on Session End:** If checked, the plugin processes `support_conference_end` events as usual but also examines the external key to determine whether to create a ticket within CA Service Desk or not. The plugin attempts to create the ticket only if this setting is enabled and if the external key is empty, is a JSON string, or is the literal value **CHAT**. If the external key is any other value, it is assumed to be a valid ticket ID, and the session is processed as usual (i.e., no ticket is created).
6. **Ticket Default Data:** A JSON string containing values that can be used to prepopulate certain fields on the newly created ticket.

CA Service Desk Services URL

The services URL for the CA Service Desk instance (ex: https://caservicedesk.example.com/axis/services/USD_R11_WebService)

CA Service Desk Username

The username for API calls to the CA Service Desk instance

CA Service Desk Password

The password for the user account for API calls to the CA Service Desk instance

Enable Automatic Incident Creation on Session Start (Rep joins session)
Enables the automatic creation of an incident when a session STARTS that is not already tied to an existing incident (Note: requires external key to be either a JSON string or the value 'CHAT' when the Representative first joins the session - also requires the plugin to accept the 'Someone Joins a Support Session' outbound event)

Enable Automatic Incident Creation on Session End
Enables the automatic creation of an incident when a session ENDS that is not already tied to an existing incident (Note: requires external key to be a JSON string, the value 'CHAT', or empty when the session ends)

Ticket Default Data

```
{ "summary": "Ticket created via Bomgar Session", "category": "pcat:5101", "priority": "3", "impact": "3", "urgency": "2", "description": "Ticket created via Bomgar Remote Support Session", "customer": "cnt:D8641E2704DFB648B44EF3A0E040E4AB", "status": "OP", "type": "T." }
```

JSON string representing default values for an automatically created ticket

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.



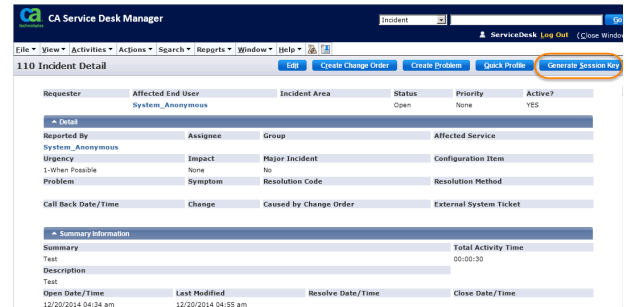
Note: If changes need to be made to a template, it is a good idea to first back up the original in case the changes ever need to be reverted.

For additional information on Handlebars templates, see handlebarsjs.com.

Use Cases for the CA Service Desk Integration with BeyondTrust Remote Support

Generate Session Key

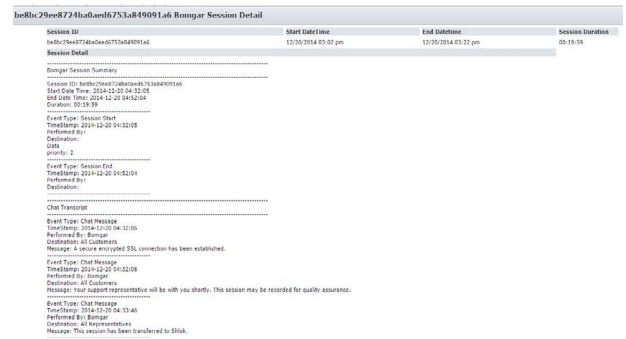
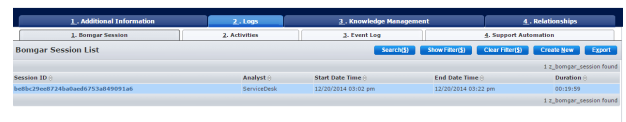
Support staff can generate a session key that can be given to the end user over the phone or via email to initiate a support session that is automatically associated with the selected ticket.



Import BeyondTrust Session Data into Ticket

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

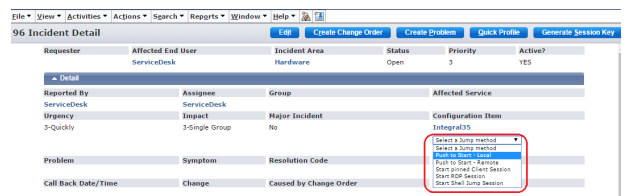
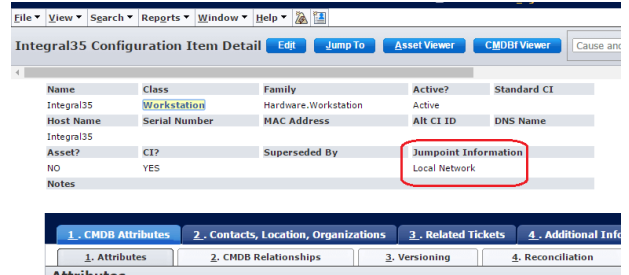
- **Chat Transcript** (including files transferred, special actions, and other events)
- **System Information** (the General section plus other select details such as disk, memory, and network)
- **Session Notes**
- **Surveys** (customer and representative)



Jump to Configuration Item

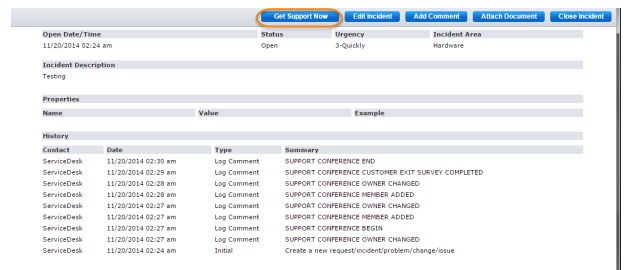
Support staff can leverage BeyondTrust Jump Technology to access a configuration item associated with a ticket directly from the Service Desk ticket. The following Jump types are available:

- **Jump Client** (Pinned Client)
- **Local Jump** (Push and Start Local)
- **Remote Jump** (Push and Start Remote)
- **Remote Desktop Protocol (RDP)**
- **Shell Jump** (Remote Console)



Click-to-Chat for Self Service Users

Self Service users can open their submitted tickets and start a chat support session directly from the Service Desk ticket. This allows the user the quickest path to resolution while also providing the representative with the necessary context to assist the user. Sessions can be elevated to full support sessions if enabled and when necessary.



Auto-Ticket Creation

For previously unreported issues or questions, the end user can submit some basic information and immediately begin a support session. Meanwhile, the integration takes the submitted information from the session and creates a new Service Desk ticket. This saves time and unnecessary steps for the end user and support staff.