



BeyondTrust

Remote Support ServiceNow CSM Integration

Table of Contents


Integrate BeyondTrust with ServiceNow CSM	4
Integration Requirements	5
Review Base Integration Requirements	5
Test the Firewall	6
Configure BeyondTrust Remote Support	7
Verify that the API is Enabled	7
Create a ServiceNow OAuth API Account	7
Add Outbound Events and Links	8
Create Custom Fields	10
Custom Fields for Automatic Case Creation	10
Configure ServiceNow for Integration with BeyondTrust	11
Install BeyondTrust Integration	11
Set Up ServiceNow Integration with Remote Support	12
Assign Users Appropriate Roles	12
Configure Case States	13
Configure Chat Now Form in Service Portal	13
ServiceNow MID Server Option	14
Configure Auto-Create Feature in ServiceNow	16
Enable Auto-Create Functionality	16
Auto-Case Creation Fields	16
Auto-Case Creation Related Lists	16
Update BeyondTrust Case Fields	17
Setup Auto-Create Field Mappings	17
Static Case Fields	17
Dynamic Case Fields	17
Case Lookup Fields	18
Dynamic Case Lookup Fields	18
Common Lookup Fields	19
Configure Other ServiceNow Environments	20
Configure Production Outbound Event	20
Configure Custom Link	20

Test the Remote Support and ServiceNow Integration	21
Test Session Key Generation	21
Test Remote Support Session Import	21
Test Case Creation	21
Use Cases for the ServiceNow CSM Integration with BeyondTrust Remote Support	23
Generate Session Key	23
Import Remote Support Session Data into a ServiceNow Record	23
Customer Initiated Chat from Service Portal	24
Access ServiceNow Records from the Representative Console	24
Associate Sessions with ServiceNow Records Manually	24
Automatic or Manual Case Creation	25
Troubleshooting	26
Support Sessions Are Not Written to ServiceNow	26
Session Key Button is not Showing on the Case Form	26
Session Key Button Gives an "Unknown Host" Message	27
"Invalid Custom Field Code Name" Message	27
"Error Validating Parameter 'snow_case_id': The Session Attribute is Invalid" Message	27
ServiceNow and BeyondTrust Integration Network Diagrams	28
ServiceNow-Remote Support Integration Network Diagram	28
ServiceNow-Remote Support Integration Mid Server Network Diagram	29

Integrate BeyondTrust with ServiceNow CSM

Customer service organizations using ServiceNow and the ServiceNow Customer Service Management (CSM) application can integrate with BeyondTrust Remote Support to improve service levels, centralize support processes, and strengthen compliance. Features of the Remote Support Remote Support and ServiceNow CSM integration are summarized below.

- **Outbound Support Sessions:** Technicians can launch Remote Support sessions from within ServiceNow case records using the **Generate Session Key** button.
- **Session Updates:** Remote Support session data is written back to ServiceNow cases. This includes chat transcripts and basic Remote Support session information, which is written to the case's work notes.
- **Automatic Case Creation:** Cases can be automatically or manually created or associated with an ongoing Remote Support session, and linked to the session for session updates at the conclusion of the session. By default, the auto-create functionality is disabled and not available to users of the integration. There is no additional cost to customers who wish to use the feature.

 *The ServiceNow application is available at the [ServiceNow Store](#).*

Integration Requirements

Outlined below are requirements for the BeyondTrust Remote Support and ServiceNow CSM integration. If any of the integration requirements are not yet met, they must be in place prior to starting the integration setup process, unless the associated features of the integration are not required.

Review Base Integration Requirements

A current version of a ServiceNow release, including:

- A working Customer Service Management application
- A working email configuration

A current release of BeyondTrust Remote Support including:

- At least one usable representative console capable of generating session keys
- A functional Remote Support site through which users can connect to representatives

To configure network firewall rules for this integration, do the following:

- Allow TCP 443 traffic from the B Series Appliance to the appropriate ServiceNow instance.
- Allow TCP 443 traffic from the appropriate ServiceNow instance to the B Series Appliance.
- Optionally, use ServiceNow MID Servers for this integration.



For more information on MID Servers, please see [MID Server Configuration](https://docs.servicenow.com/bundle/utah-servicenow-platform/page/product/mid-server/concept/mid-server-landing.html) at <https://docs.servicenow.com/bundle/utah-servicenow-platform/page/product/mid-server/concept/mid-server-landing.html>.

Test the Firewall

It is important to test all requirements of the integration prior to beginning setup. Most of these can be tested by the Remote Support and ServiceNow administrators within their respective systems, but to test the network firewall, the BeyondTrust admin should take the following steps to confirm that the necessary rules are in place.

1. Log in to a machine either external to the B Series Appliance's network or in the same VPN as the ServiceNow instance, depending on how ServiceNow is connecting to the B Series Appliance's network.
2. Log in to the B Series Appliance's **/appliance** interface.
3. Browse to **Support > Utilities :: TCP Connection Test**.
4. Enter the hostname of the ServiceNow instance, enter the port number of **443**, and then click **Test**. A successful result is a *Connected* status message.



Note: Do not enter the protocol when entering the ServiceNow instance (<https://servicenow.example.com>, for example). Instead, use only the fully qualified domain name (servicenow.example.com). In most environments, BeyondTrust Remote Support resides in a DMZ network and has a public DNS address which ServiceNow contacts over the public internet. In some environments, Remote Support is not publicly accessible. In these cases, contact ServiceNow about implementing a VPN connection to your internal network for ServiceNow. For more information, please see [Virtual Private Network \(VPN\) at \[https://docs.servicenow.com/bundle/utah-platform-security/page/administer/encryption/concept/c_SetUpAVPN4SNowBusNet.html\]\(https://docs.servicenow.com/bundle/utah-platform-security/page/administer/encryption/concept/c_SetUpAVPN4SNowBusNet.html\)](https://docs.servicenow.com/bundle/utah-platform-security/page/administer/encryption/concept/c_SetUpAVPN4SNowBusNet.html).

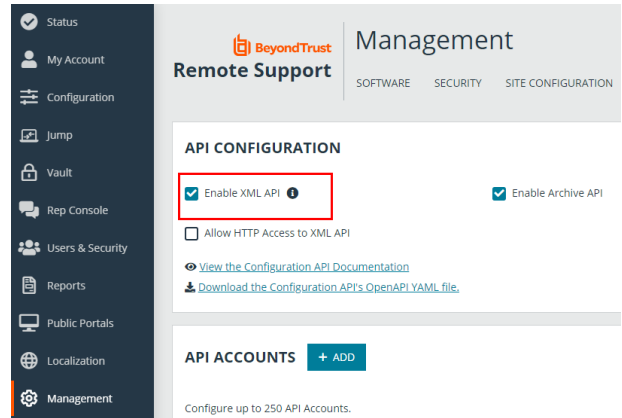
Configure BeyondTrust Remote Support

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

Verify that the API is Enabled

The Remote Support integration requires the Remote Support XML API to be enabled. This feature is used from within the integrating software to communicate with the Remote Support APIs.

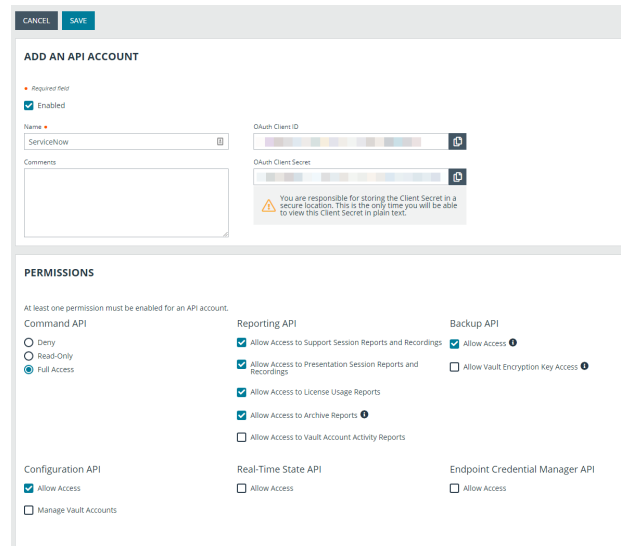
Go to `/login > Management > API Configuration` and verify that **Enable XML API** is checked.



Create a ServiceNow OAuth API Account

The ServiceNow API account is used from within ServiceNow to make Remote Support Command API calls.

1. Go to `/login > Management > API Configuration`.
2. Click **Add** to create a new API account. Name it *ServiceNow* or something similar.
3. Under **Reporting API**, check:
 - **Allow Access to Support Session Reports and Recordings.**
 - **Allow Access to Presentation Session Reports and Recordings.**
4. The **OAuth Client ID** and **OAuth Client Secret** are used during the OAuth configuration step in ServiceNow. Make note of these and store them in a secure location.
5. Click **Save**.



Add Outbound Events and Links

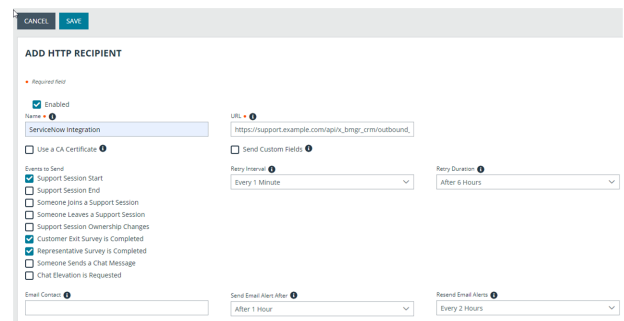
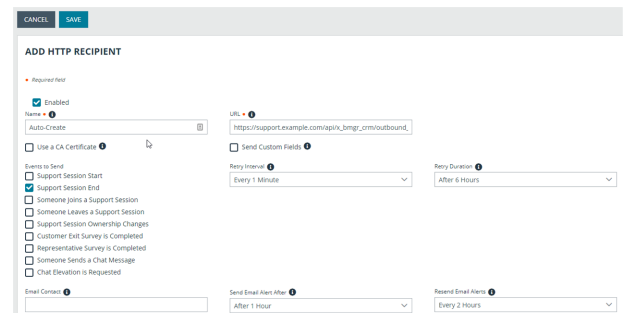
! IMPORTANT!

Options 1 and 2 apply to integration without case creation. If using case creation, you cannot use Option 1 and Option 2 or 3 simultaneously, as Option 1 is automatic and options 2 and 3 are manual. You must choose one or the other based on their use cases and process. You may, however, use options 2 and 3 simultaneously.

Option 1: Add Outbound Events

Outbound events are used to notify ServiceNow that a Remote Support session has finished and is ready to be imported into ServiceNow, or that a representative has joined a Remote Support session and a *create case* request can be sent to ServiceNow.

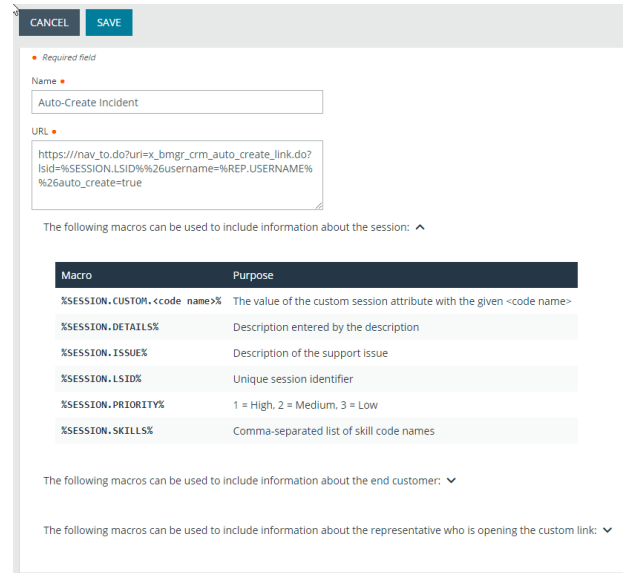
- Go to **/login > Management > Outbound Events**.
- Click **Add** to create a new event. The event names provided below are suggestions. You may use something different, depending on your ServiceNow instance.
 - For session import, name it *ServiceNow Integration*.
 - For case creation, name it *ServiceNow Integration Auto-Create*.
- Set the URL.
 - For session import: `https://example.service-now.com/api/x_bmgr_crm/outbound_event/session_end`, where `example.service-now.com` is the ServiceNow instance name.
 - For case creation: `https://example.service-now.com/api/x_bmgr_crm/outbound_event/case?auto_create=true`, where `example.service-now.com` is the ServiceNow instance name.
- Scroll to **Events to Send** and check the box labeled **Support Session End**
- At the top of the page, click **Save**.

Option 2: Set up the Custom Link

Remote Support custom links can be configured to allow representatives to access or create ServiceNow records from within the Remote Support representative console.

1. Browse to **Rep Console > Custom Links**.
2. Click **Add** to create a new custom link.
3. Enter a name for the link.
4. Set the URL.
 - For access: `https://example.service-now.com/nav_to.do?uri=sn_customerservice_case.do?sys_id=%SESSION.CUSTOM.SNOW_CASE_ID%`, where `example.service-now.com` is the ServiceNow instance name.
 - For case creation: `https://example.service-now.com/nav_to.do?uri=x_bmgr_crm_auto_create_link.do?lsid=%SESSION.LSID%%26username=%REP.USERNAME%%26auto_create=true`, where `example.service-now.com` is the ServiceNow instance name.
 - If needed, you can use any of the available macros to customize the link according to your specifications.



ADD A CUSTOM LINK

Name

Auto-Create Incident

URL

`https://nav_to.do?uri=x_bmgr_crm_auto_create_link.do?lsid=%SESSION.LSID%%26username=%REP.USERNAME%%26auto_create=true`

Macro	Purpose
<code>%SESSION.CUSTOM.<code name>%</code>	The value of the custom session attribute with the given <code name>
<code>%SESSION.DETAILS%</code>	Description entered by the description
<code>%SESSION.ISSUE%</code>	Description of the support issue
<code>%SESSION.LSID%</code>	Unique session identifier
<code>%SESSION.PRIORITY%</code>	1 = High, 2 = Medium, 3 = Low
<code>%SESSION.SKILLS%</code>	Comma-separated list of skill code names



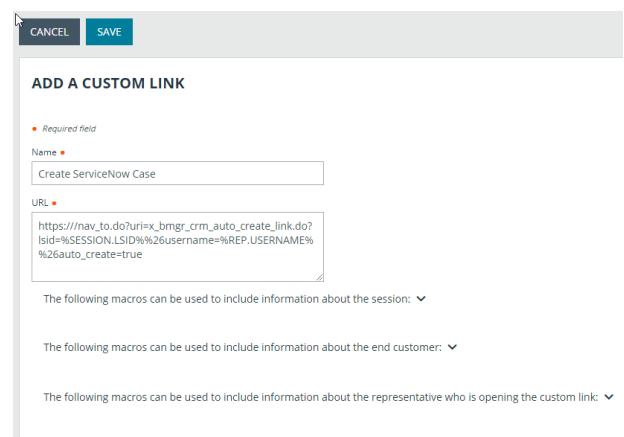
Note: If the customer is using External Key to store the case `sys_id`, then use the macro `%SESSION.CUSTOM.EXTERNAL_KEY%` instead.

5. Click **Save** to save the new link.

Option 3: Set up Advanced Custom Link

Remote Support custom links can be configured to allow Representatives to launch a page in ServiceNow where they can choose to **Associate** the session with an existing case, or **Create** a new case by clicking a button.

1. Browse to **Rep Console > Custom Links**.
2. Click **Add** to create a new custom link.
3. Enter a name for the link, then set the URL to `https://example.service-now.com/nav_to.do?uri=x_bmgr_crm_auto_create_`



ADD A CUSTOM LINK

Name

Create ServiceNow Case

URL

`https://nav_to.do?uri=x_bmgr_crm_auto_create_link.do?lsid=%SESSION.LSID%%26username=%REP.USERNAME%%26auto_create=true`

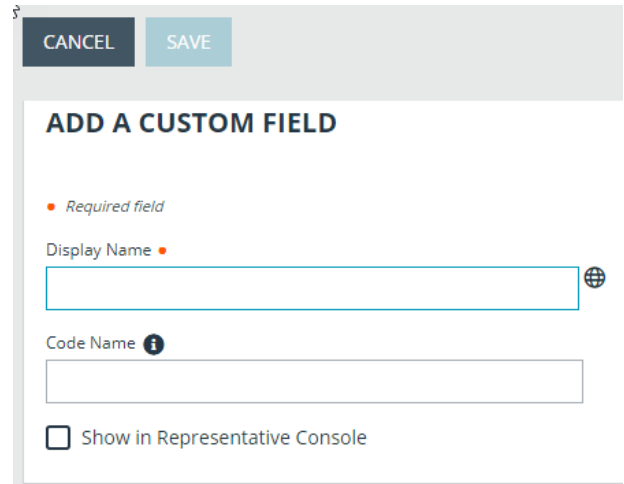
`link.do?lsid=%SESSION.LSID%%26username=%REP.USERNAME%%26auto_create=true`, where `example.service-now.com` is the ServiceNow instance name. If needed, you can use any of the available macros to customize the link according to your specifications.

4. Click **Save** to save the new link.

Create Custom Fields

BeyondTrust custom fields are used to map ServiceNow Tasks (tasks, change requests, problem records, and service catalog requests), call records, and configuration items to BeyondTrust sessions.

1. Browse to **Configuration > Custom Fields**.
2. Click **Add** to create a new custom field.
3. Enter the following values:
 - **Display Name:** ServiceNow Case ID
 - **Code Name:** snow_case_id
 - **Show in Rep Console:** checked
4. Click **Save** to save the new field.



Custom Fields for Automatic Case Creation

There are a number of custom fields that are available to use in an auto-create scenario. These fields can be used to populate values when the session starts, and then configured in ServiceNow to be mapped to case values.

1. Browse to **Configuration > Custom Fields**.
2. Click **Add** for each of the following custom field code names.



Note: The **Code Names** must match, but the **Display Name** is arbitrary.

- customer_email
- customer_username
- snow_case_id



Note: The **snow_case_id** field should already have been created with the base ServiceNow ERS CSM Integration configuration.

- custom_field_1
- custom_field_2
- custom_field_3

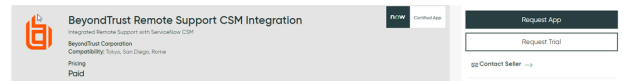
Configure ServiceNow for Integration with BeyondTrust

Unless otherwise noted, all of the steps in this section take place in the ServiceNow interface. We recommend that you initially use development or test instances of ServiceNow before installation in the production instance, so that the integration can be thoroughly tested.

Install BeyondTrust Integration

The Integration must be purchased directly from BeyondTrust. After the purchase, it is installed via the ServiceNow Store. There is no mechanism to purchase the integration at the ServiceNow Store. Follow the steps below to request and install the integration.

1. Log in to your ServiceNow platform with your corporate account. Please contact your company Now Support (HI) admin if you do not have credentials.
2. Search for BeyondTrust Remote Support CSM Integration, or [go to the app's page at https://store.servicenow.com/sn_appstore_store.do#!/store/application/d1918af2db7513000710e7b51b96195b/](https://store.servicenow.com/sn_appstore_store.do#!/store/application/d1918af2db7513000710e7b51b96195b/).
3. Click **Request App**.
4. Request Remote Support with CSM.
5. Within a business day, BeyondTrust approves the request with a \$0 purchase price, and sends a copy of these installation instructions to the requester.
6. Once you receive the confirmation, return to the app's page in the ServiceNow store.
7. Click **Complete Purchase**.
8. Review the contract details.
9. If you agree, check **Accept the Site Terms of Use**.
10. Click **Complete Purchase**.
11. The app is ready for installation on your ServiceNow instance(s).
12. Repeat the steps below for each instance.
13. Login to the ServiceNow instance on which you want to install the app.
14. Navigate to **System Applications > All Available Applications > All**.
15. Search for the app you want to install.
16. Click **Install**.



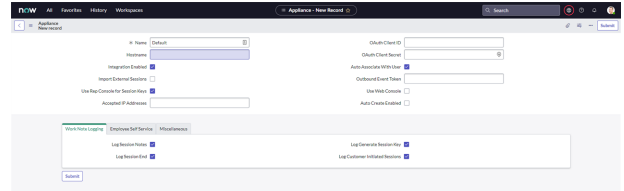
For more information on installing applications, please see [ServiceNow's Help Page at https://store.servicenow.com/sn_appstore_store.do#!/store/helpcenter](https://store.servicenow.com/sn_appstore_store.do#!/store/helpcenter).

Set Up ServiceNow Integration with Remote Support

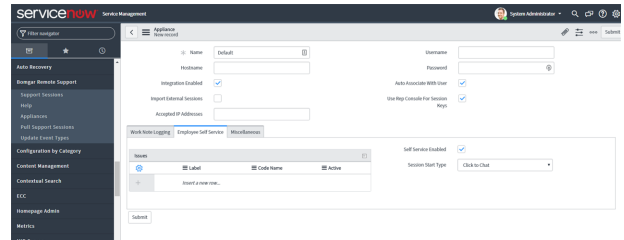
Follow these steps in ServiceNow to connect ServiceNow to Remote Support.

Browse to **BeyondTrust CSM Appliances**.

Click **New** to add a new Remote Support appliance record and enter the following values:




- Name:** Must be **Default**.
- Hostname:** Host name of the Remote Support deployment.
- OAuth Client ID:** The OAuth client ID that is used to authenticate to the Remote Support deployment. This is obtained in a previous step, "[Create a ServiceNow OAuth API Account](#)" on page 7.
- OAuth Client Secret:** The OAuth client secret that is used to authenticate to the Remote Support deployment. This is obtained in a previous step, "[Create a ServiceNow OAuth API Account](#)" on page 7.
- Integration Enabled:** Check this box to enable the integration.
- Outbound Event Token:** The token that is used as an added security measure to confirm outbound events are coming from the Remote Support deployment that is sending the same token. If left blank, this outbound event token process is ignored. If a value is provided, however, the same value must be sent from all outbound events coming from BeyondTrust as a parameter named **outbound_event_token**.
- Accepted IP Addresses:** A comma-separated list of IP addresses from which this integration accepts outbound events.

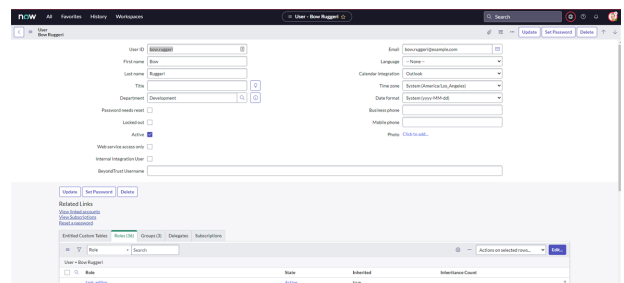
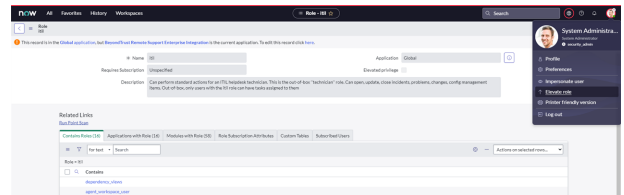


Assign Users Appropriate Roles

Give the **x_bmgr_crm.agent** role to ITIL users who provide technical support using this integration.

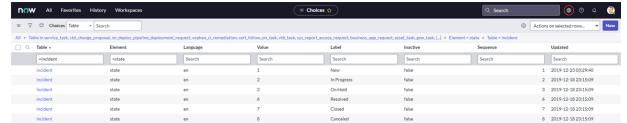
 **Note:** You must elevate the admin's role in order to make the following change.

- Browse to **User Administration > Users**.
- Select a user.
- Click the **Roles** tab, and then click the **Edit** button.
- Add the **x_bmgr_crm.agent** role from the **Collection** list to the **Roles** list.
- Click **Save**.



Configure Case States

Case states are used in the Remote Support Integration, specifically UI Actions like **Session Key**, to control when the button shows up on case forms.



Instance	State	Label	Value	Label	Instance	Instance	Updated
Instance	state	1	Search	New	Instance	Instance	1 2019-10-23 09:38:46
Instance	state	2	Search	In Progress	Instance	Instance	2 2019-10-23 09:38:46
Instance	state	3	Search	On Hold	Instance	Instance	3 2019-10-23 09:38:46
Instance	state	4	Search	Closed	Instance	Instance	4 2019-10-23 09:38:46
Instance	state	5	Search	Resolved	Instance	Instance	5 2019-10-23 09:38:46
Instance	state	6	Search	Cancelled	Instance	Instance	6 2019-10-23 09:38:46

1. Browse to **Case** and click **Open** to see a list of open cases.
2. Click **Case Number** to select a case.
3. Right-click the label for the **State** field or any other field which uses a dropdown or choice list, and click **Show Choice List**.
4. Filter the list by **Table=sn_customerservice_case** and **Element=state**.
5. Take note of the different **sn_customerservice_case** choices, specifically the value of cases with a label of **Resolved** or **Closed**, and any custom case states for which the **Remote Support Session Key** button should not appear.
6. Go back to the case, right-click the **Session Key** button located in the **Case** title bar, and select **Configure > UI Actions**.
7. In the **UI Actions** list, click **Remote Support Session Key**, select the **Condition** field, and ensure **current.state** is configured to include the IDs of the case states for which the session key button should not appear.

The following code in the **UI Action** condition field controls which states the **Session Key** button displays for. By default, the **Session Key** button displays for cases whose states are set to **New**, **Open**, or **Awaiting Info**, for example, cases with states set to **1**, **10**, or **18** respectively.

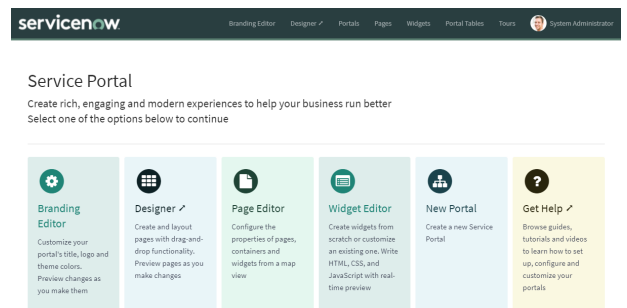
```
x_bmgr_crm.BomgarUtil.showSessionKeyOnCaseForm('Default', current.getUniqueValue(), [1,10,18])
```

For example, if you had a custom state of 27, which represented a **Pending Case**, and you wanted the **Session Key** button to also display for that state, you would modify the code above to the following:

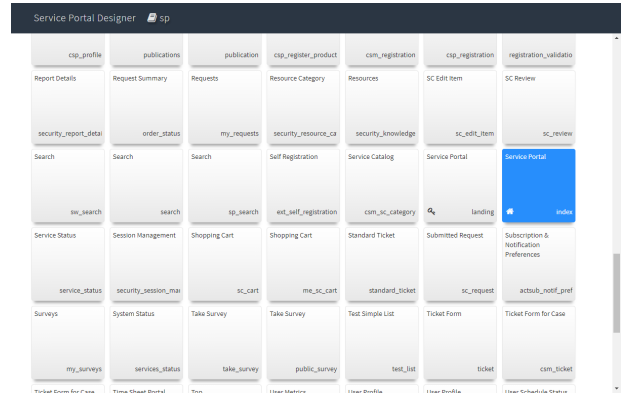
```
x_bmgr_crm.BomgarUtil.showSessionKeyOnCaseForm('Default', current.getUniqueValue(), [1,10,18,27])
```

Configure Chat Now Form in Service Portal

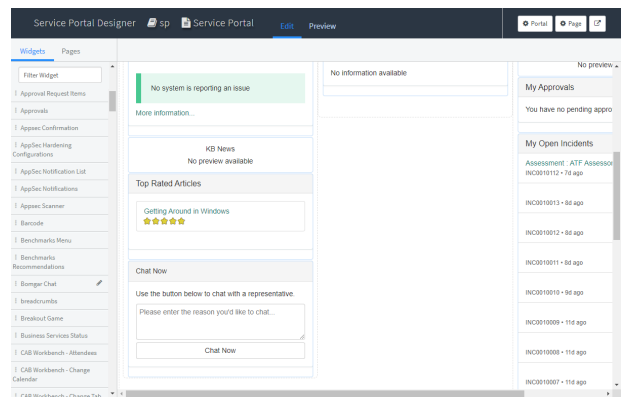
1. Browse to Service Portal and click **Service Portal Configuration** to open the configuration landing page.
2. Click **Designer** to launch the Service Portal Designer.



3. Scroll down the list of pages and click the Service Portal index page (or whichever page is marked with the home icon).



4. In the **Widgets** menu on the left, find the **Bomgar Chat** widget. Drag it onto the page, and place it where you'd like it to display.

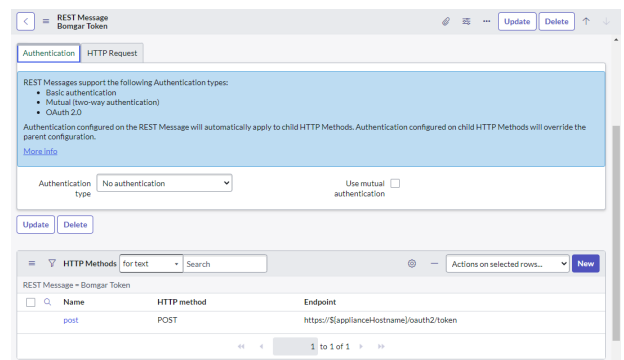


ServiceNow MID Server Option

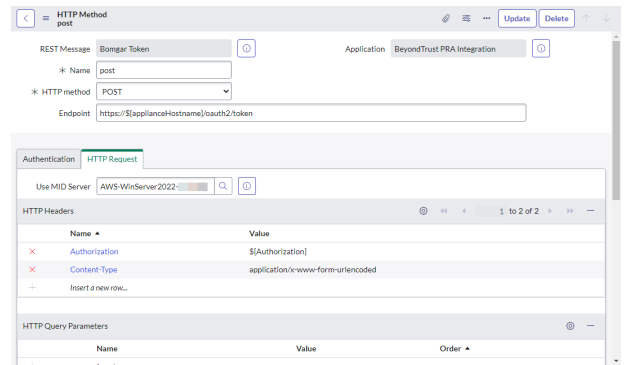
It is possible to avoid direct connection between ServiceNow and Remote Support by using a MID server for internal Remote Support deployments.

To configure the integration to use a MID server for API requests to a Remote Support site, specify the MID server to use on the individual outbound REST messages:

1. In ServiceNow, navigate to **System Web Services > Outbound > REST Message**.
2. Filter to show only messages for the BeyondTrust application being configured.
3. Click the **Name** of one of the messages to edit its properties.
4. In the **HTTP Methods** related list at the bottom, select the **Name** of the method (typically **post**).



5. On the resulting form, select the **HTTP Request** tab and select your MID server in the **Use MID Server** field.
6. Click **Update** to save the changes.



The screenshot shows the configuration for an HTTP Method. The REST Message is 'Bongar Token', the Name is 'post', the HTTP method is 'POST', and the Endpoint is 'https://\$appliance\$hostname/oauth2/token'. The Authentication is set to 'HTTP Request' and the Use MID Server is 'AWS-WinServer2022'. The HTTP Headers section contains two entries: 'Authorization' with value '\$[Authorization]' and 'Content-Type' with value 'application/x-www-form-urlencoded'.

Repeat these steps for each outbound REST message that is a part of the application.



For more information on MID servers, please see [MID Server](https://docs.servicenow.com/bundle/utah-servicenow-platform/page/product/mid-server/concept/mid-server-landing.html) at <https://docs.servicenow.com/bundle/utah-servicenow-platform/page/product/mid-server/concept/mid-server-landing.html>.

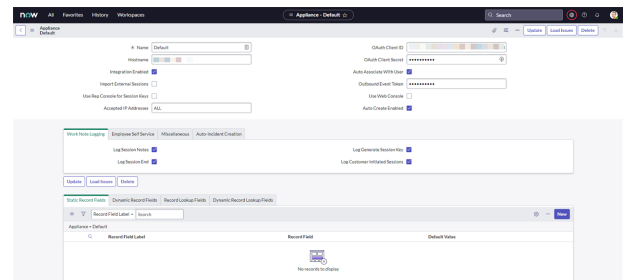
Configure Auto-Create Feature in ServiceNow

Unless otherwise noted, all of the steps in this section take place in the ServiceNow interface. We recommend that you initially use development or test instances of ServiceNow before installation in the production instance, so that the integration can be thoroughly tested.

Enable Auto-Create Functionality

Follow these steps to enable auto-create:

1. Browse to **BeyondTrust CSM** and click **Appliances**. Select the appliance record you are configuring.
2. Right-click on the title bar, and then select **Configure > Form Layout**.
3. Drag the field labeled **Auto Create Enabled** from the **Available** list to the **Selected** list.
4. Click the **Save** button.
5. The **Auto Create Enabled** check box displays on the form. Check this box, and then click the **Update** button on the title bar. The form reloads.



Auto-Case Creation Fields

A section just below the main (top) section of the form labeled **Auto-Case Creation** contains the following fields:

- **Auto-Create For These Portals** (comma-separated list): Auto-creates cases for sessions whose public portal is listed in this field.
- **Required Auto-Create Custom Fields** (comma-separated list): Auto-creates cases for sessions only if *all* of the custom fields listed are populated with a value.
- **Auto-Create for Attended Sessions** (check box): Auto-creates cases for sessions that are attended by a customer when this box is checked.
- **Auto-Create for Unattended Sessions** (check box): Auto-creates cases for sessions that are unattended when this box is checked.

Auto-Case Creation Related Lists

The following **Related Lists** fields at the bottom of the form are used to configure additional auto-create functionality:

- **Static Case Fields**
- **Dynamic Case Fields**
- **Case Lookup Fields**
- **Dynamic Case Lookup Fields**

Update BeyondTrust Case Fields

Case fields are used when mapping static data or BeyondTrust data to ServiceNow case fields during auto-creation of cases. This step updates the database with all the available event types.

Browse to **BeyondTrust CSM** and click **Update BeyondTrust Case Fields**. This loads all of the available **BeyondTrust Case Fields** into the database, in preparation for field mapping.

Setup Auto-Create Field Mappings

Browse to **BeyondTrust CSM** and click **Appliances**. Click the desired appliance record. Scroll down to the related list section. There are four related lists, each representing a type of field mapping:

- **Static Case Fields:** maps a static, default value to a ServiceNow case field.
- **Dynamic Case Fields:** maps a **Remote Support Session** value to a ServiceNow case field.
- **Case Lookup Fields:** maps a static, default value to a ServiceNow case lookup field.
- **Dynamic Case Lookup Fields:** maps a **Remote Support Session** value to a ServiceNow case lookup field.

Static Case Fields

Use this type of mapping when you want to assign the same static value to the same ServiceNow text field or option list for every **Case** created.



Example: For every case that is auto-created, the customer wants the **State** field to be set to **Resolved**. The case field for **State** is selected from the menu and the value for **Resolved** is **6**, so the following values are set:

- **Case Field** = **State** (selected from the menu)
- **Default Value** = **6**

Case Field Label is arbitrary.

- **Case Field Label:** An arbitrary value used to identify the mapping.
- **Case Field:** The case field that is populated upon auto-creation.
- **Default Value:** The static value that is populated in the selected case field.

Dynamic Case Fields

Use this type of mapping when you want to assign a field from the Remote Support session to the same ServiceNow text field or option list for every case created.



Example: For every case that is auto-created, the customer wants the **Short Description** field to be set to the value contained in the **Remote Support Session's External Key** field. The case field for **Short Description** is selected from the **Case Field** menu and the **Remote Support External Key** field is selected from the **Remote Support Field** menu, so the



following values are set:


- **Case Field:** Short Description (selected from the menu)
- **BeyondTrust Field:** External Key (selected from the menu)

The Field Label is arbitrary.

- **Case Field Label:** An arbitrary value used to identify the mapping.
- **Case Field:** The case field that is populated upon auto-creation.
- **BeyondTrust Field:** The Remote Support session value that is populated in the selected case field.

Case Lookup Fields

Use this type of mapping when you want to assign the same static value to the same ServiceNow field, which requires a lookup, for every case created.



Field Label	Record Field	Lookup Table ID	Lookup Field ID	Default Value
Some Lookup Overide	Assigned To	sys_user	user_name	bob.ross



Example: For every case that is auto-created, the customer wants the **Assigned To** field to be set to a specific user whose username is **bob.ross**. The case field for **Assigned To** is selected from the **Case Field** menu. The **Lookup Table ID** is set to the table to search and the **Lookup Field ID** is set to the field in the lookup table to compare the static value with. **Default Value** is to the static value with which to lookup the user, so the following values are set:

- **Case Field:** Assigned To (selected from the menu)
- **Lookup Table ID:** sys_user
- **Lookup Field ID:** user_name
- **Default Value:** bob.ross

The Field Label is arbitrary.

- **Field Label:** A value used to identify the mapping.
- **Case Field:** The case field that should be populated upon auto-creation.
- **Lookup Table ID:** The ServiceNow table to lookup the value for this case field.
- **Lookup Field ID:** The field in the lookup table to search when looking up the value for this case field.
- **Default Value:** The static value to use when looking up the value for this case field.

Dynamic Case Lookup Fields

Use this type of mapping when you want to assign a field from the Remote Support session to the same ServiceNow field, which requires a lookup, for every case created.



Field Label	Record Field	Lookup Table ID	Lookup Field ID	BeyondTrust Field
Color Overide	Color Euler JE	sys_user	user_name	Customer Username



Example: For every case that is auto-created, the customer wants the **Contact** field to be set to a specific user, whose username we will get from the Remote Support session. The case field for **Contact** is selected from the **Case Field** menu. The **Lookup Table ID** is set to the table to search and the **Lookup Field ID** is set to the field in the lookup table to compare

 the *BeyondTrust* value with. **Remote Support Customer Username** field is selected from the **BeyondTrust Field** menu, so the following values would be set:


- **Case Field:** Contact (selected from the menu)
- **BeyondTrust Field:** Customer Username (selected from the menu)
- **Lookup Table ID:** sys_user
- **Lookup Field ID:** user_name

The Field Label is arbitrary

- **Field Label:** A value used to identify the mapping.
- **Case Field:** The case field that is populated upon auto-creation.
- **Lookup Table ID:** The ServiceNow table in which to look up the value for this case field.
- **Lookup Field ID:** The field in the lookup table to search when looking up the value for this case field.
- **BeyondTrust Field:** The BeyondTrust session value to use when looking up the value for this case field.

Common Lookup Fields

The following information is provided to help when creating lookup field mappings. A **Lookup Field** is populated by looking in another ServiceNow table to retrieve a value.

 **Example:** The **Caller** field on a **Case** is a lookup field. The name of the ServiceNow table that is used is **sys_user**. You can lookup a user by querying the **user_name** field with a value of **john.doe**. This populates the **Caller** field with the **sys_user** record for John Doe.

Below are some common ServiceNow record lookup fields that you may encounter in an auto-create scenario.

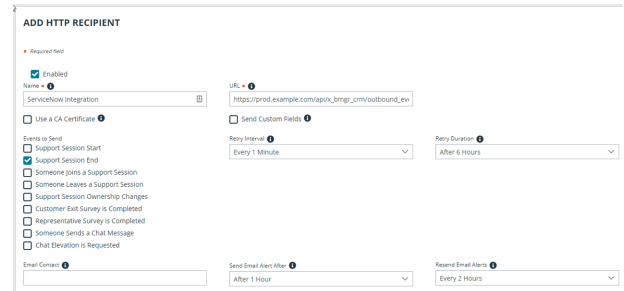
Case Field	Case Field Name	Table	Common Query Fields
Account	account	customer_account	name
Asset	asset	alm_asset	display_name
Assigned To	assigned_to	sys_user	x_bmgr_support_ent_bomgar_username name user_name email
Assignment Group	assignment_group	sys_user_group	name
Configuration Item	cmdb_ci	cmdb_ci	name
Contact	contact	customer_contact	name user_name email

Configure Other ServiceNow Environments

The steps below are typically used after the integration has been installed and configured in a test or development instance of ServiceNow, and is being transferred to a QA or production instance.

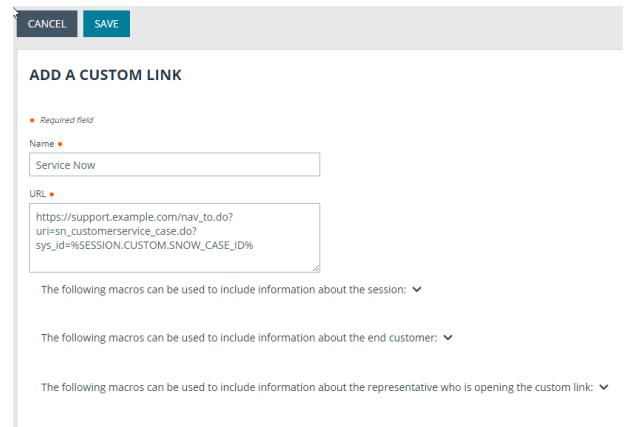
Configure Production Outbound Event

1. Go to **/login > Management > Outbound Events**.
2. Click **Add** and name the new HTTP recipient **ServiceNow Integration** or something similar, depending on your ServiceNow instance.
3. In the **URL** field, paste and replace the name of the original ServiceNow instance with that of the new one, such that `/api/x_bmgr_crm/outbound_event/session_end` is preserved at the end. The result should be similar to `https://example.service-now.com/api/x_bmgr_crm/outbound_event/session_end`, as opposed to `https://example-dev.service-now.com/api/x_bmgr_crm/outbound_event/session_end`.
4. Scroll to **Events to Send** and check **Support Session End**.
5. In the URL field, paste and replace the name of the original ServiceNow instance with that of the new one, such that `/api/x_bmgr_crm/outbound_event/session_end` is preserved at the end. The result should be similar to `https://servicenow_prod.example.com/api/x_bmgr_crm/outbound_event/session_end`, as opposed to `https://servicenow_dev.example.com/api/x_bmgr_crm/outbound_event/session_end`.
6. Locate the outbound event created during testing and click **Edit**. Check the **Disabled** box and save.



Configure Custom Link

1. Go to **/login > Rep Console > Custom Links**.
2. Click **Add** to create a new custom link.
3. Enter a name for the link, and then set the URL to `https://example.service-now.com/nav_to.do?uri=sn_customerservice_case.do?sys_id=%SESSION.CUSTOM.SNOW_CASE_ID%`, where `example.service-now.com` is the ServiceNow instance name. If needed, you can use any of the available macros to customize the link according to your specifications.
4. Test the integration setup in its new location following the same steps used to test the original instance.



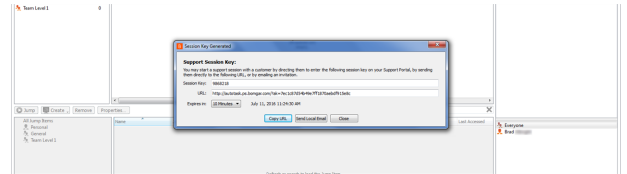

For more information, please see ["Test the Remote Support and ServiceNow Integration" on page 21](#).

Test the Remote Support and ServiceNow Integration

The following steps take place in ServiceNow and Remote Support and are provided to ensure that the integration is working properly. Troubleshooting suggestions are provided with each step in case of failure.

Test Session Key Generation

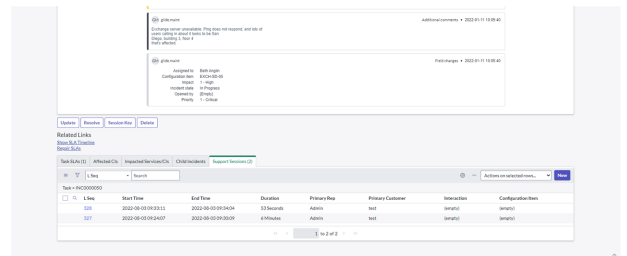
1. Log in to the Remote Support representative console, and then log in to ServiceNow with the same account.
2. Open an active case in ServiceNow. Confirm the **State** field matches one of the values for which the **Session Key** button should appear, and make sure the button appears as expected. Remember that the case must be saved for the button to appear.
3. Click the **Session Key** button. This launches the Remote Support representative console, subsequently opening the **Session Key Dialog** box.



Test Remote Support Session Import

1. Log in to ServiceNow as a customer service agent or an admin. Use the **Session Key** button as described above to start a Remote Support session.
2. End the session from the representative console.

Refresh the ServiceNow case from which the session key was generated, scroll to work notes section of the form, and check that an entry for the completed session is shown. Basic session details should be shown, as well as any chat dialogue. If not, make sure the following items are true:

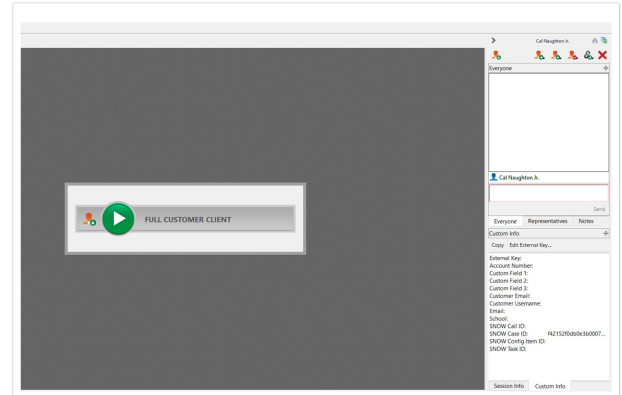


- The **API User Connection** test works correctly as described above.
- There are no Remote Support errors reported for your ServiceNow instance in the Remote Support **Outbound Events** list. Your Remote Support admin can check this in Remote Support from the **/login** web interface under **Management > Outbound Events**.
- The IP address is set up correctly, following the steps below:
 - Log in to ServiceNow as an admin.
 - Browse to **System Logs > Transactions**, remove all existing filters, and add a URL filter of **/x_bmgr_support_ent_bomgar_post.do**.
 - Click one of the results and make sure the originating IP address of the transaction is included in the **IP Address** field of the integration appliance configuration record settings under **BeyondTrust CSM > Appliances > Default**.

Test Case Creation

- You can quickly test the case creation with a URL such as `https://beyondtrust.example.com/api/start_session?issue_menu=1&customer.name=Bob&customer.company=BobCoInc&customer.details=Credentials&codeName=issue_4&session.custom.customer_username=b.smith&c2cjs=1`.

- If using the **Outbound Event** approach **Option 1**, after the representative joins the session a new case appears when you refresh the case list in ServiceNow. Additionally, the ServiceNow Case ID is populated in the representative console after a few seconds with the **sys_id** of the newly created case.
- If using the **Custom Link** approach, **Options 2 or 3**, after the representative clicks the **Create Case** custom link, a new case appears when you refresh the case list in ServiceNow. Additionally, the ServiceNow Case ID should be populated in the representative console after a few seconds with the **sys_id** of the newly created case.



Use Cases for the ServiceNow CSM Integration with BeyondTrust Remote Support

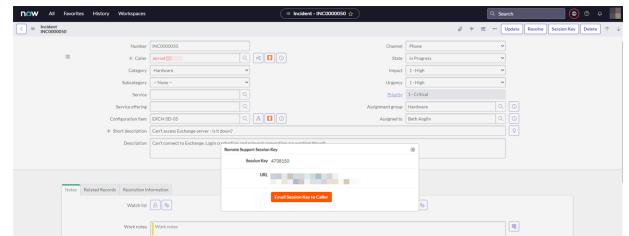
Customer support organizations using ServiceNow Customer Service Management (CSM) can integrate with BeyondTrust Remote Support to improve service levels, centralize support processes, and strengthen compliance. Use cases that are addressed by the BeyondTrust Remote Support and ServiceNow CSM integration are listed below.

Generate Session Key

Technicians can generate a session key from within a case and give the customer the key to initiate a support session.

The integration provides support for case records.

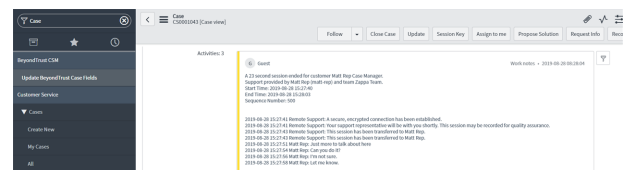
Once the session ends, basic session data, along with the chat dialogue from the session, are imported into ServiceNow and written to the work notes of the case record from which the session key was generated.



Import Remote Support Session Data into a ServiceNow Record

Once the Remote Support session ends, the associated case record in ServiceNow is automatically updated with information gathered during the session, including:

- Chat transcript
- Basic session information

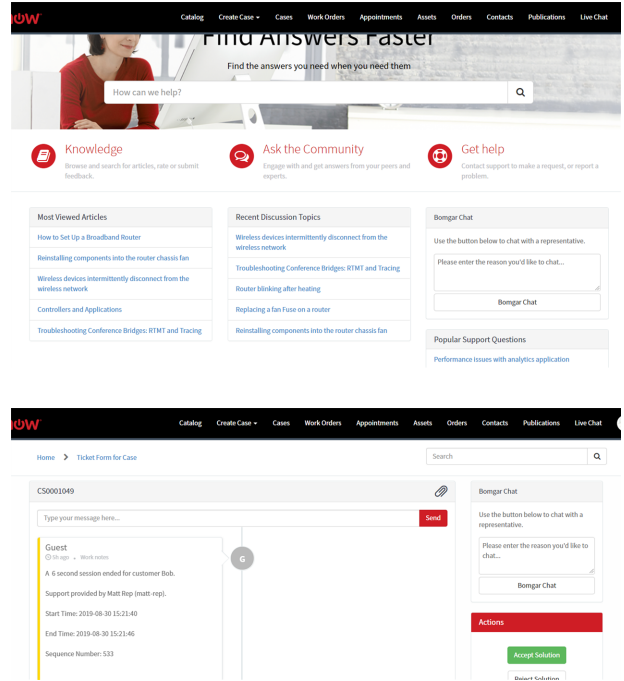


Customer Initiated Chat from Service Portal

Customers can request a support session with a representative from a ServiceNow Service Portal where the **Remote Support Get Support Now** widget has been added. This includes any context within a case, like the case form, or any context outside the scope of a case, like the CSM homepage.

This functionality allows customers an expedient path to resolution, while also providing the technician with the necessary context to effectively assist the customer.

Sessions initiated using Remote Support's click-to-chat functionality can be elevated to full support sessions if the technician deems it necessary.



Access ServiceNow Records from the Representative Console

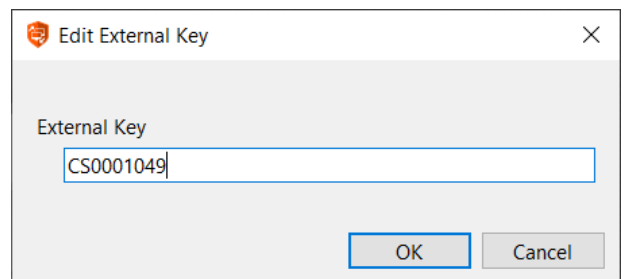
Using Remote Support's custom links ability, a representative can access the associated ServiceNow case record directly from within the representative console. This saves time searching for the case record in ServiceNow and provides the representative with any available issue details, history, or other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

Associate Sessions with ServiceNow Records Manually

Whether a representative has just created a case for the current session or has found that one already exists, even sessions originating outside the scope of a ServiceNow record can be manually associated with the appropriate item, allowing session details to be automatically added to the case when the session ends.

To make this association, the representative enters the numeric ID of the case record into the **External Key** field while in session.

Technicians can use the human readable record number from ServiceNow to easily make this association.



Automatic or Manual Case Creation

For customers who have applicable processes, the integration can be configured to automatically create a ServiceNow case based on Remote Support sessions, either when the representative joins the session or manually via a Remote Support representative console custom link. This process can simplify workflows and reduce the number of clicks necessary for support representatives.

Troubleshooting

This section is designed to assist members of implementation and support teams who are either installing or supporting the ServiceNow integration with Remote Support. The items listed here cover potential issues you may encounter when working with the integration, along with steps to take to investigate and alleviate those issues.

Throughout this section you'll see a reference to *ServiceNow application logs*. This refers to the application logs in ServiceNow, which can be found in **System Logs > System Log > Application Logs**.

ServiceNow has other logs that are also helpful. Those include:

- **Transaction Logs**, found at **System Logs > Transactions**
- **All Logs**, found at **System Logs > System Log > All**
- **Warning Logs**, found at **System Logs > System Log > Warnings**
- **Error Logs**, found at **System Logs > System Log > Errors**



Note: Using this troubleshooting guide requires knowledge of both ServiceNow and Remote Support.

Support Sessions Are Not Written to ServiceNow

If support sessions aren't written in ServiceNow, there could be a communication issue. First check to ensure communication is open from Remote Support to ServiceNow.

1. Log in to the Remote Support **/login** interface and ensure the **Support Session End** option is checked on the **HTTP Recipient for the Outbound Event**. If **Support Session End** is not checked, check the box, save the HTTP recipient, then run another support session and see if the record is written to ServiceNow.
2. If the **Outbound Event** looks good, log in to the **/appliance** interface and run a TCP test to the hostname of the customer's ServiceNow instance on port 443. If the TCP test yields a non-200 response, involve your network team to determine if there is a network block in place, such as a firewall or network rule.
3. If the TCP test yields a 200 response, then start another session from ServiceNow, end the session, then open the ServiceNow transaction logs and filter by records that are **Created on Today** (or Last 15 minutes for fewer results) AND URL starts with **/api/x_bmgr_support_ent/outbound_event/session_end**. If you don't see a record for the session you just ran, then there is a communication issue from Remote Support to ServiceNow and you should recommend the customer involve their network team to determine if there is a network block in place such as a firewall or network rule.
4. If you see the transaction record in the **Transaction Logs**, then double-check and reset the **OAuth Client ID** and **OAuth Client Secret** fields on the appliance configuration record in ServiceNow. It's very common that this **OAuth Client Secret** value is copied to the clipboard, but the **API Account** record is never saved in Remote Support/**login**. We recommend that you copy the value and immediately save the **API Account**, then paste the value in the appliance configuration record in ServiceNow. Run another support session. If the Support Session record still does not show up in ServiceNow, check the ServiceNow application log to see if there are any errors that may explain why the record isn't being written.

Session Key Button is not Showing on the Case Form

If the **Session Key** button is not showing on the case form, there are few things that can be checked.

1. Check to see if there were any adjustments or customizations made to any of the integration's UI actions. Because these UI actions render the buttons, we recommend that these not be edited.

2. If no customizations to the UI actions were made, ensure the appliance configuration record is named **Default**, with matching case and no white space. Any name other than **Default** causes the **Session Key** button to be hidden.
3. If the name of the appliance configuration record is **Default**, view the customer's case states and ensure that the **Condition** field in the UI Action displays the button for the proper case states.

 For more information, please see "[Configure Case States](#)" on page 13.

Session Key Button Gives an "Unknown Host" Message

If you get an *Unknown host* error, there is essentially a communication issue from ServiceNow to the Remote Support deployment, unless the hostname is set incorrectly on the appliance configuration record.

Double-check the hostname on the appliance configuration record in ServiceNow. If it looks correct, contact your network team to determine if there is a network block in place, such as a firewall or network rule.

"Invalid Custom Field Code Name" Message

An *invalid custom field code name* error can occur when there is an attempt to generate a session key, but a custom attribute with that code name doesn't exist in the system. This can happen if a step is missed in the initial integration configuration.

Create the missing custom field and try to start another support session from ServiceNow.

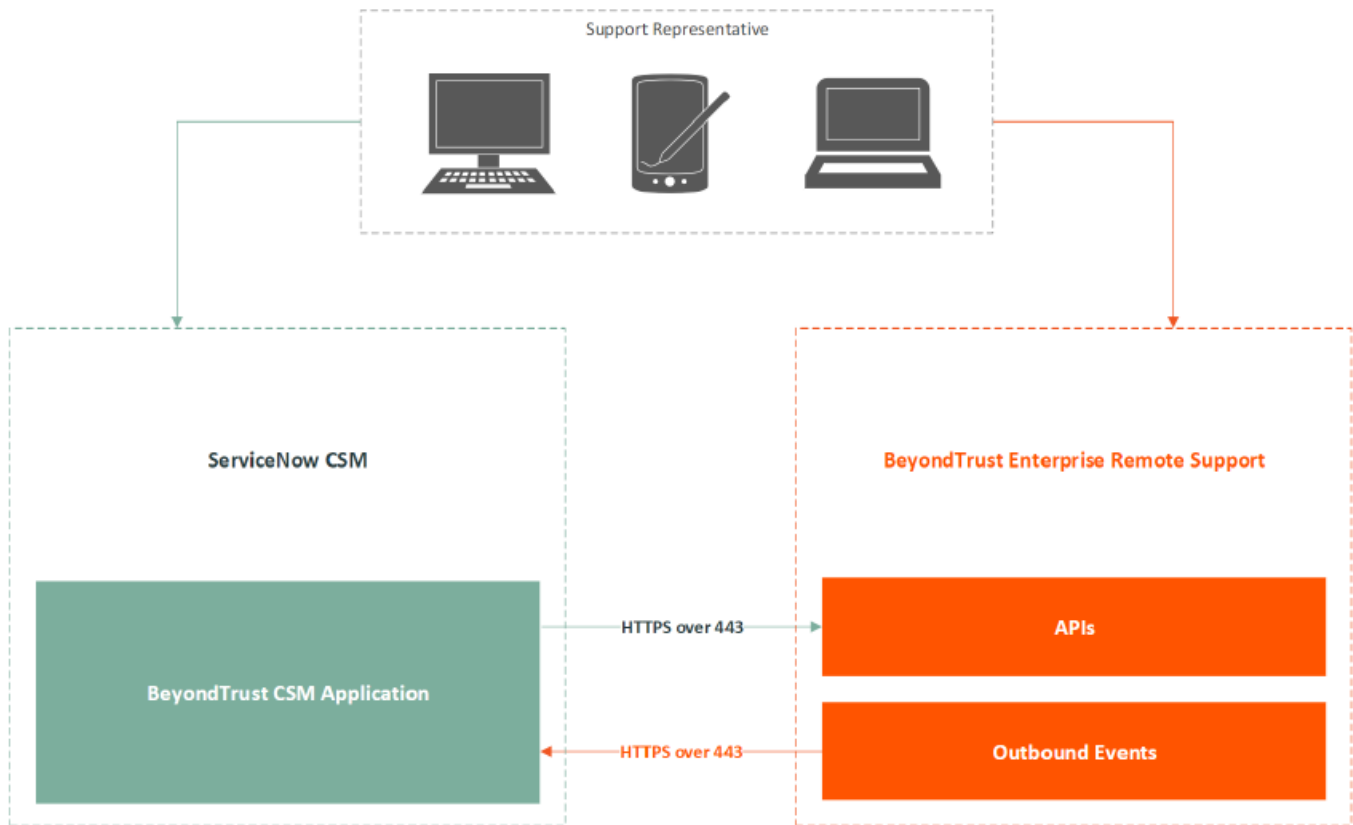
"Error Validating Parameter 'snow_case_id': The Session Attribute is Invalid" Message

An *error validating parameter 'snow_case_id': The session attribute is invalid* message can occur when there is an attempt to generate a session key, but a custom attribute with that code name doesn't exist in the system. This can happen if a step is missed in the initial integration configuration.

Create the missing custom field and try to start another support session from ServiceNow.

ServiceNow and BeyondTrust Integration Network Diagrams

ServiceNow-Remote Support Integration Network Diagram



ServiceNow-Remote Support Integration Mid Server Network Diagram

