

BOMGAR™

API Programmer's Guide

Version 1.12.0 (Bomgar 14.2.x)

Table of Contents

API Programmer's Guide Version 1.12.0 (for Bomgar 14.2.x)	8
Command API	9
API Command: get_logged_in_reps	11
XML Response for get_logged_in_reps Query	11
Element Names and Attributes	11
Query Example: get_logged_in_reps	11
API Command: get_support_teams	13
Optional Parameter for get_support_teams	13
XML Response for get_support_teams Query	13
Element Names and Attributes	13
Query Examples: get_support_teams	14
API Command: generate_session_key	15
Required Parameters for generate_session_key	15
Optional Parameters for generate_session_key	15
XML Response for generate_session_key Query	16
Query Examples: generate_session_key	16
API Command: join_session	18
Required Parameters for join_session	18
XML Response for join_session Query	18
Query Examples: join_session	18
API Command: transfer_session	19
Required Parameters for transfer_session	19
XML Response for transfer_session Query	19
Query Examples: transfer_session	19
API Command: set_session_attributes	20
Required Parameter for set_session_attributes	20
Optional Parameters for set_session_attributes	20
XML Response for set_session_attributes Query	20
Query Examples: set_session_attributes	20
API Command: get_session_attributes	21
Required Parameter for get_session_attributes	21

XML Response for get_session_attributes Query	21
Element Names and Attributes	21
Query Example: get_session_attributes	21
API Command: terminate_session	22
Required Parameter for terminate_session	22
XML Response for terminate_session Query	22
Query Examples: terminate_session	22
API Command: get_appliances	23
XML Response for get_appliances	23
Element Names and Attributes	23
Query Example: get_appliances	23
API Command: get_connected_client_list	24
Optional Parameters for get_connected_client_list	24
XML Response for get_connected_client_list	24
Element Names and Attributes	24
Query Examples: get_connected_client_list	25
API Command: get_connected_clients	26
Required Parameters for get_connected_clients	26
XML Response for get_connected_clients	26
Element Names and Attributes	26
Query Examples: get_connected_clients	29
API Command: check_health	31
XML Response for check_health Query	31
Query Example: check_health	31
HTTP Status Check	31
API Command: set_failover_role	32
Required Parameter for set_failover_role	32
Optional Parameters for set_failover_role	32
XML Response for set_failover_role Query	32
Query Examples: set_failover_role	32
API Command: get_api_info	34
XML Response for get_api_info Query	34
Element Names and Attributes	34

Query Example: get_api_info	34
Representative Console Scripting and Client Scripting API	35
The Bomgar Representative Console Script File	35
New and Deprecated Command Line Parameters for the Representative Console ..	36
The Bomgar Client Scripting API	36
API Script Command: start_pinned_client_session	39
Required Parameters for the start_pinned_client_session Command	39
Optional Parameters for the start_pinned_client_session Command	39
Query Examples: start_pinned_client_session	39
API Script Command: generate_session_key	40
Optional Parameters for the generate_session_key Command	40
Query Examples: generate_session_key	40
API Script Command: push_and_start_local	41
Required Parameter for push_and_start_local	41
Optional Parameter for push_and_start_local	41
Query Examples: push_and_start_local	41
API Script Command: start_vpro_session	42
Required Parameters for the start_vpro_session Command	42
Optional Parameters for the start_vpro_session Command	42
Query Examples: start_vpro_session	42
Session Generation API	43
Optional Parameters	44
Query Examples	45
Start Sessions with Session Key Acceptance	46
Start Sessions with Click-To-Chat or Collaborative Browser Sharing	47
Using JavaScript	47
Examples	47
Starting a Browser Session with a Session Key	47
Starting a Browser Session with Representative Information	48
Starting a Browser Session with a Session Key and Custom Fields	48
Starting a Browser Session with a Session Key and a Specified Language	48
Starting a Browser Session with an Issue Object (by ID) and Customer Information	49
Starting a Browser Session with an Issue Object (by Code Name) and Skills	49

Starting a Browser Session with an Issue Form Element	50
Starting a Click-to-Chat Session with a Session Key	50
Starting a Click-to-Chat Session with a Session Key and fallbackToFullWindow ..	50
options Properties	51
rep Properties	51
issue Properties	52
customer Properties	52
attributes Properties	52
uiOptions Properties	53
Collaborative Browser Sharing Functionality	53
Click-to-Chat Without Using JavaScript	53
Start Sessions with External Keys (TicketID)	55
Manual Entry	55
Programmatic Assignment	55
Using the External Key	55
Start Sessions with an Embedded Bomgar Button	56
Reporting API	57
Download Reports with SupportSession	59
Parameters for SupportSession	59
Optional Parameter for SupportSession	60
XML Response for SupportSession Query	60
Element Names and Attributes	60
Query Examples for SupportSession	67
Download Reports with SupportSessionListing	69
Parameters for SupportSessionListing	69
XML Response for SupportSessionListing Query	69
Element Names and Attributes	69
Query Examples for SupportSessionListing	70
Download Reports with SupportSessionSummary	71
Parameters for SupportSessionSummary	71
XML Response for SupportSessionSummary Query	71
Element Names and Attributes	71
Query Examples	72

Download Reports with SupportSessionRecording	73
Parameter for SupportSessionRecording	73
Query Example for SupportSessionRecording	73
Download Reports with ShowMyScreenRecording	74
Parameters for ShowMyScreenRecording	74
Query Examples for ShowMyScreenRecording	74
Download Reports with CommandShellRecording	75
Parameters for CommandShellRecording	75
Query Examples for CommandShellRecording	75
Download Reports with PresentationRecording	76
Parameter for PresentationRecording	76
Query Example for PresentationRecording	76
Download Exit Survey Reports with SupportCustExitSurvey and SupportRepExitSurvey	77
Parameters for SupportCustExitSurvey and SupportRepExitSurvey	77
Optional Parameter	77
XML Response for SupportCustExitSurvey and SupportRepExitSurvey Queries	78
Element Names and Attributes	78
Query Examples for SupportCustExitSurvey and SupportRepExitSurvey	79
Download Reports with SupportTeam	82
Parameters for SupportTeam	82
Optional Parameter for SupportTeam	82
XML Response for SupportTeam Query	82
Element Names and Attributes	83
Query Examples for SupportTeam	85
Download Reports with LicenseUsage	86
Parameters for LicenseUsage	86
XML Response for LicenseUsage Query	86
Element Names and Attributes	86
Query Examples for LicenseUsage	87
Backup API	88
Required Parameters for Backup API	88
Query Example	88
Test Scenario	89

API Change Log	90
API Version Reference	91
Disclaimers, Licensing Restrictions and Tech Support	92

API Programmer's Guide | Version 1.12.0 (for Bomgar 14.2.x)

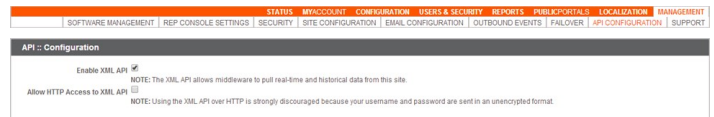
Front-end integration of the Bomgar API enables customers to correlate Bomgar support sessions with third-party or in-house developed applications to pull report data, issue commands, or automatically save a backup of the Bomgar Appliance's software configuration on a recurring basis.

One common example of API integration would be linking a help desk ticketing system to Bomgar sessions to track issue resolution.

You could also add a feature to an application to enable the representative to generate a session from directly within that program instead of the Bomgar representative console.

To use the Bomgar API, ensure that the **Enable XML API** option is checked on the **Management > API Configuration** page of the `/login` administrative interface.

For the examples in the following pages, a sample URL of **support.example.com** is used. Please replace this URL with your Bomgar Appliance's public site URL.



Starting with Bomgar 14.1, the command and reporting APIs return XML responses that declare a namespace. If you are parsing these responses with a namespace-aware parser, you will need to set the namespace appropriately or ignore the namespace while parsing the XML.

- Reporting API: `http://www.bomgar.com/namespaces/API/reporting`
- Command API: `http://www.bomgar.com/namespaces/API/command`

Note: The above namespaces are returned XML data and are not functional URLs.

IMPORTANT! If you have been using the API prior to 14.1, this change could break backward compatibility with existing integrations if namespaces are not handled properly. **If possible, be sure to test your code against a site running Bomgar 14.1.1 or higher before upgrading.**

Command API

The Bomgar command API is designed to enable you to send commands to your Bomgar Appliance from an outside application. This can be used to start or transfer a Bomgar support session without using the standard representative console, to get a list of logged-in representatives, to obtain a list of support teams and issues, or to get information about your Bomgar API version. You can also check the health of your appliance or change an appliance's failover role.

Commands are executed by sending a simple HTTP request to the Bomgar Appliance. The request can be sent using any HTTPS-capable socket library or scripting language module, a web browser, or a URL fetcher such as **cURL** or **wget**. Either **GET** or **POST** may be used as the request method.

IMPORTANT! When making consecutive API calls, you must close the connection after each API call.

Note: By default, access to the API is SSL-encrypted; however, you can choose to allow HTTP access by checking the **Allow HTTP Access to XML API** option on the **Management > API Configuration** page of the **/login** administrative interface. **It is highly recommended that HTTP remain disallowed as a security best practice.**

The command API URL is <https://support.example.com/api/command>.

An XML schema which formally describes the format of the command API response is available at <https://support.example.com/api/command.xsd>.

Required Parameters for Command API

username=[string]	The username to use when issuing commands. For all commands except get_api_info , this user must have permission to use the command API. The user must have permission to perform remote support in order to issue generate_session_key and must be an administrator in order to issue join_session , set_session_attributes , get_session_attributes , terminate_session , transfer_session , check_health , or set_failover_role .
password=[string]	The password associated with this username.
action=[string]	The type of action to perform. Can be get_logged_in_reps , get_support_teams , generate_session_key , join_session , transfer_session , set_session_attributes , get_session_attributes , terminate_session , get_appliances , get_connected_client_list , get_connected_clients , check_health , set_failover_role , or get_api_info .

IMPORTANT! If you are experiencing a high volume of support requests, repeatedly calling a command such as **get_logged_in_reps** or **get_support_teams** can potentially bottleneck your system. Therefore, a best practice is not to request a list of representatives or teams with each support request. Instead, if you are making the same API call in succession, consider caching the results for a period of time and reusing them. New sessions requests should reference this cached list instead of calling for the list each time.

Starting with Bomgar 14.1, the command API returns XML responses that declare a namespace. If you are parsing these responses with a namespace-aware parser, you will need to set the namespace appropriately or ignore the namespace while parsing the XML.

- Command API: <http://www.bomgar.com/namespaces/API/command>

Note: The above [namespace](#) is returned XML data and is not a functional URL.

IMPORTANT! If you have been using the API prior to 14.1, this change could break backward compatibility with existing integrations if namespaces are not handled properly. **If possible, be sure to test your code against a site running Bomgar 14.1.1 or higher before upgrading.**

API Command: get_logged_in_reps

The `get_logged_in_reps` request returns XML data about all logged-in representatives. It requires no additional parameters.

In order to issue the `get_logged_in_reps` command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API**.

XML Response for get_logged_in_reps Query

```
<logged_in_reps>
```

Returns a `<rep>` element for each logged-in representative. If no representatives are logged in, this element will contain no `<rep>` elements. If an error occurs, it will contain an `<error>` element describing the problem.

Element Names and Attributes

/logged_in_reps/rep

id (attribute)	Unique ID assigned to the representative.
<display_name>	This element is deprecated as of API version 1.10.0 but still exists for backwards compatibility. Its value is the same as that of <code><public_display_name></code> .
<public_display_name>	The public display name currently assigned to the representative.
<private_display_name>	The private display name currently assigned to the representative.
<type>	The type of representative logged in. Types include Normal , Embassy and Invited .
<direct_link>	An HTML anchor tag containing the URL that customers can use to download the customer client to connect directly to the representative.
<logged_in_since>	The date and time at which the representative logged in.
<presentation_count>	The number of active presentations the representative is currently running.
<support_session_count>	The number of active sessions the representative is currently running.
<showing_on_rep_list>	Integer value (1 or 0) indicating if the rep currently has permission to show on the public site and has their Showing On Representative List preference checked in the rep console.

Query Example: get_logged_in_reps

```
get_logged_in_reps
```

```
https://support.example.com/api/command?username=test&password=test&action=get_logged_in_reps
```

IMPORTANT! *If you are experiencing a high volume of support requests, repeatedly calling a command such as `get_logged_in_reps` or `get_support_teams` can potentially bottleneck your system. Therefore, a best practice is not to request a list of representatives or teams with each support request. Instead, if you are making the same API call in succession, consider caching the results for a period of time and reusing them. New sessions requests should reference this cached list instead of calling for the list each time.*

API Command: get_support_teams

The **get_support_teams** request returns XML data containing all configured support teams and all the issues configured for each team.

In order to issue the **get_support_teams** command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API**.

Optional Parameter for get_support_teams

showmembers

Causes the output to also list all the representatives who are members of each team. Depending on team configuration, showing all members could add a significant amount of data to the output and should be used sparingly.

XML Response for get_support_teams Query

<support_teams>

Contains a <support_team> element for each support team. If no support teams have been created, this element will contain no <support_team> elements. If an error occurs, it will contain an <error> element describing the problem.

Element Names and Attributes

/support_teams/support_team

id (attribute)	Unique ID assigned to the support team.
<name>	The name of the support team.
<embassy>	Integer value (1) present only if the team is an Embassy.
<issues>	Contains an <issue> element for each issue associated with this support team, as described below. If no issues have been configured for this team, the <issue> element will be blank.
<support_session_count>	The number of sessions waiting in this team queue.
<members>	Displayed only if the showmembers parameter has been included in the request. Contains a <representative> element for each member of this team. If no representatives have been assigned to this team, the <members> element will be blank.

/support_teams/support_team/issues/issue

id (attribute)	Unique ID assigned to this issue.
<title>	The title of the issue.

/support_teams/support_team/members/representative

id (attribute)	Unique ID assigned to the representative.
<username>	The username assigned to the representative.

<display_name>	This element is deprecated as of API version 1.10.0 but still exists for backwards compatibility. Its value is the same as that of <public_display_name>.
<public_display_name>	The public display name currently assigned to the representative.
<private_display_name>	The private display name currently assigned to the representative.

Query Examples: `get_support_teams`

Show names and issues	<code>https://support.example.com/api/command?username=test&password=test&action=get_support_teams</code>
Show names, issues, and members	<code>https://support.example.com/api/command?username=test&password=test&action=get_support_teams&showmembers</code>

IMPORTANT! If you are experiencing a high volume of support requests, repeatedly calling a command such as `get_logged_in_reps` or `get_support_teams` can potentially bottleneck your system. Therefore, a best practice is not to request a list of representatives or teams with each support request. Instead, if you are making the same API call in succession, consider caching the results for a period of time and reusing them. New sessions requests should reference this cached list instead of calling for the list each time.

API Command: generate_session_key

The **generate_session_key** command creates a new session key to be used in starting a support session. Note that if your Bomgar Appliance has multiple public sites, the session key created may be associated with any of these sites, depending on the method used to download the customer client.

For example, Site A has a hostname of support.example.com, and Site B has a hostname of remote.example.com. When a **generate_session_key** request is made to support.example.com with a **url_hostname** of remote.example.com, both a session key code and a unique session key URL will be generated.

If the customer goes to the generated URL to download the customer client, then the session will be associated with Site B, because the session key URL points to the hostname designated by the **url_hostname** parameter.

However, the customer could also download the customer client by submitting the session key code on either site. Therefore, if the customer goes to Site A to submit the code, then the session will be associated with Site A, while if he or she goes to Site B, the session will be associated with Site B.

In order to issue the **generate_session_key** command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API** along with the permission **Allowed to Provide Remote Support**.

Required Parameters for generate_session_key

type=[string]	The type of session for which you would like to generate a session key. Currently, the only supported value is support .
queue_id=[string]	<p>The queue in which the session should be placed. Can be one of general, rep:[id], team:[id], or embassy:[id], where [id] is the numeric ID for the representative, team, or Embassy queue in which you wish to place this session.</p> <p>Can also be rep_username:[username]. This call will work only if a single user with the given username exists; otherwise, an error message will be returned.</p> <p>Can also be issue:[issue_code_name].</p> <p>To get a representative's ID, see "API Command: get_logged_in_reps" on page 11. To get a team's ID, see "API Command: get_support_teams" on page 13.</p>

Optional Parameters for generate_session_key

session.custom.external_key=[string]	An arbitrary string that can link this session to an identifier on an external system, such as a help desk ticket ID. This has a maximum length of 1024 characters.
session.custom.[custom field]=[string]	<p>The code name and value of any custom fields. These fields must first be configured in /login > Management > API Configuration.</p> <p>Each custom field has a maximum length of 1024 characters. The maximum total size of all combined custom fields, including the external key, must be limited to 10KB.</p>
session.priority=[integer]	The priority of the session, from 1 to 3 . 1 = high, 2 = medium, and 3 = low.
session.skills=[string]	A comma-separated list of the code names of skills to assign to a session.

ttl=[integer]	Time in seconds for which this key should be valid. If omitted, the maximum session key timeout set in the administrative interface will be used.
url_hostname=[string]	Hostname to use in the URL generated for the session key. Defaults to the primary hostname for your Bomgar Appliance.

Note: The parameter **external_key** has been deprecated as of 14.2 and has been replaced with **session.custom.external_key**. However, it is still available for backward compatibility.

Note: If a parameter is set via **generate_session_key** and is then overwritten via another API (e.g., **start_session**) then the second attribute will take precedence.

XML Response for generate_session_key Query

<type>	The type of session for which this key was generated. Currently, the only supported value is support .
<ttl>	Time in seconds for which this key is valid.
<expires>	The timestamp at which this session key expires.
<queue>	The queue in which this session will be placed. Will be general , rep , team or embassy . Also contains an available attribute. For a session key targeting a representative, the value is 1 if that representative is logged in. For a session key targeting a team, embassy, or general queue, the value is 1 if at least one representative is logged in for that queue or the queue is persistent. If no representative is available for the targeted queue, the value is 0 .
<queue_id>	The numeric ID of the queue.
<external_key>	A string that links this session to an identifier on an external system, such as a help desk ticket ID.
<short_key>	The seven-character string that the customer can enter on your public site to start a session.
<key_url>	The session key url to which the customer can go to start a session.
<mail_subject>	The subject line of the session key email invitation.
<mail_body>	The body of the session key email invitation.

Query Examples: generate_session_key

General queue	https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=general
Specific representative by ID	https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=rep:1

Specific team	<code>https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=team:1</code>
Specific Embassy	<code>https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=embassy:1</code>
Specific representative by username	<code>https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=rep_username:admin</code>
Specific issue	<code>https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=issue:other</code>
General queue, 1 hour time to live	<code>https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=general&ttl=3600</code>
General queue, external key and custom field	<code>https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=general&session.custom.external_key=ABC1234&session.custom.custom_field1=Custom%20Value</code>
General queue, specific Hostname	<code>https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=general&url_hostname=remote.example.com</code>
General queue, skills and priority set	<code>https://support.example.com/api/command?username=test&password=test&action=generate_session_key&type=support&queue_id=general&session.priority=1&session.skills=codename1,codename2</code>

API Command: join_session

The `join_session` command adds a logged in representative to an existing support session.

In order to issue the `join_session` command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API** along with the permission **Is Administrator**.

Required Parameters for join_session

<code>lsid=[string]</code>	The ID of the session to which the representative will be added.
<code>rep_id=[string]</code>	Unique ID assigned to the representative. To get a representative's ID, see " API Command: get_logged_in_reps " on page 11.

XML Response for join_session Query

<code><success></code>	Returns a message of Representative successfully added if successful.
<code><error></code>	Returns an error message if not successful.

Query Examples: join_session

Add representative to session <code>c69a8e10bea9428f816cfababe9815fe</code>	<code>https://support.example.com/api/command?username=test&password=test&action=join_session&lsid=c69a8e10bea9428f816cfababe9815fe&rep_id=151</code>
--	---

API Command: transfer_session

The **transfer_session** command transfers an active session from one queue to another.

In order to issue the **transfer_session** command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API** along with the permission **Is Administrator**.

Required Parameters for transfer_session

Isid=[string]	The ID of the session you wish to transfer.
queue_id=[string]	The queue to which this session should be transferred. Can be one of general , rep:[id] , team:[id] , or embassy:[id] , where [id] is the numeric ID for the representative, team, or Embassy queue to which you wish to transfer this session. To get a representative's ID, see " API Command: get_logged_in_reps " on page 11. To get a team's ID, see " API Command: get_support_teams " on page 13.

XML Response for transfer_session Query

<success>	Returns a message of Successfully transferred if the transfer was successful.
<error>	Returns an error message if the transfer was not successful.

Query Examples: transfer_session

Session c69a8e10bea9428f816cfababe9815fe to general queue	https://support.example.com/api/command?username=test&password=test&action=transfer_session&Isid=c69a8e10bea9428f816cfababe9815fe&queue_id=general
Session c69a8e10bea9428f816cfababe9815fe to specific representative	https://support.example.com/api/command?username=test&password=test&action=transfer_session&Isid=c69a8e10bea9428f816cfababe9815fe&queue_id=rep:1
Session c69a8e10bea9428f816cfababe9815fe to specific team	https://support.example.com/api/command?username=test&password=test&action=transfer_session&Isid=c69a8e10bea9428f816cfababe9815fe&queue_id=team:1

API Command: set_session_attributes

The **set_session_attributes** command sets the external key and other custom attributes for an active support session.

In order to issue the **set_session_attributes** command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API** along with the permission **Is Administrator**.

Required Parameter for set_session_attributes

Isid=[string]	The ID of the session whose attributes you wish to set. The session must currently be active.
---------------	---

Optional Parameters for set_session_attributes

session.custom.external_key=[string]	An arbitrary string that can link this session to an identifier on an external system, such as a help desk ticket ID. This has a maximum length of 1024 characters.
session.custom.[custom field]=[string]	The code name and value of any custom fields. These fields must first be configured in /login > Management > API Configuration . Each custom field has a maximum length of 1024 characters. The maximum total size of all combined custom fields, including the external key, must be limited to 10KB.

Note: The parameter **custom.external_key** has been deprecated as of 14.2 and has been replaced with **session.custom.external_key**. However, it is still available for backward compatibility.

Note: If an attribute is not listed in the URL, it will keep its existing value. To clear an attribute, you must set the attribute to an empty string.

XML Response for set_session_attributes Query

<success>	Returns a message of Session attributes were set if the attributes were set successfully.
<error>	Returns an error message if the attributes were not set successfully.

Query Examples: set_session_attributes

Set external key for session c69a8e10bea9428f816cfababe9815fe	https://support.example.com/api/command?username=test&password=test&action=set_session_attributes&Isid=c69a8e10bea9428f816cfababe9815fe&session.custom.external_key=ABC123
Set a custom value for session c69a8e10bea9428f816cfababe9815fe	https://support.example.com/api/command?username=test&password=test&action=set_session_attributes&Isid=c69a8e10bea9428f816cfababe9815fe&session.custom.custom_field1=Custom%20Value

API Command: get_session_attributes

The `get_session_attributes` command returns attributes set for an active support session.

In order to issue the `get_session_attributes` command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API** along with the permission **Is Administrator**.

Required Parameter for get_session_attributes

<code>Isid=[string]</code>	The ID of the session whose attributes you wish to get. The session must currently be active.
----------------------------	---

XML Response for get_session_attributes Query

<code><custom_attributes></code>	Contains a <code><custom_attribute></code> element for each custom attribute set for the session.
<code><error></code>	Returns an error message if the attributes were not retrieved successfully.

Element Names and Attributes

/custom_attributes/custom_attribute

<code>display_name (attribute)</code>	The display name assigned to the custom attribute.
<code>code_name (attribute)</code>	The code name assigned to the custom attribute.

Query Example: get_session_attributes

Get custom attributes for session c69a8e10bea9428f816cfababe9815fe	<code>https://support.example.com/api/command?username=test&password=test&action=get_session_attributes&Isid=c69a8e10bea9428f816cfababe9815fe</code>
---	--

API Command: terminate_session

The **terminate_session** command terminates a support session that is in progress.

In order to issue the **terminate_session** command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API** along with the permission **Is Administrator**.

Required Parameter for terminate_session

Isid=[string]	The unique ID representing the session you wish to terminate.
---------------	---

XML Response for terminate_session Query

<success>	Returns a message of Successfully terminated if the termination was successful.
<error>	Returns an error message if the termination was not successful.

Query Examples: terminate_session

Session da4b510978a541d49398e88c66e28475 terminated	https://support.example.com/api/command?action=terminate_session&Isid=da4b510978a541d49398e88c66e28475&username=username&password=password
---	---

API Command: get_appliances

The **get_appliances** command returns XML data containing all configured Bomgar Appliances in a failover relationship or cluster.

In order to issue the **get_appliances** command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API**.

XML Response for get_appliances

<appliances>	Contains an <appliance> element for each connected appliance. If an error occurs, it will contain an <error> element describing the problem.
--------------	--

Element Names and Attributes

<i>/appliances/appliance</i>	
id (attribute)	The appliance's GUID.
<name>	The name of the appliance.
<public_hostname>	The public hostname of the appliance.
<cluster_role>	The role the appliance plays in a cluster. Can be one of single (if not part of a cluster), master , or traffic .
<failover_role>	The role the appliance plays in a failover relationship. Can be one of none (if failover is not configured), primary , or backup .
<accepting_connections>	Integer value (1 or 0) indicating if this appliance accepts new client connections. If not part of a cluster, this will always be 1 .
<timezone_offset>	The number of seconds away from UTC.

Query Example: get_appliances

get_appliances	https://support.example.com/api/command?username=test&password=test&action=get_appliances
----------------	---

API Command: get_connected_client_list

The `get_connected_client_list` command returns XML data containing a summary or list of all connected Bomgar clients.

In order to issue the `get_connected_client_list` command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API**.

Optional Parameters for get_connected_client_list

<code>type=[string]</code>	The types of clients to return in the results. Can be a comma-separated list of values. Supported values are all (default), representative , support_customer , presentation_attendee , and push_agent . <i>Note: Currently, pinned_client is not a possible value. If the count of pinned Jump Clients is needed in the summary, then all must be specified.</i>
<code>summary_only=[boolean]</code>	To return only a summary, set this to 1 .

XML Response for get_connected_client_list

<code><connected_client_list></code>	Contains a <code><connected_client_summary></code> element with a summary of the data. Also contains a <code><connected_client></code> element for each client currently connected to the appliance. If an error occurs, it will contain an <code><error></code> element describing the problem.
--	--

Element Names and Attributes

`/connected_client_list/connected_client_summary`

<code><appliance_summary></code>	An <code><appliance_summary></code> element is created for each connected appliance.
--	--

`/connected_client_list/connected_client_summary/appliance_summary`

<code>id</code> (attribute)	The appliance's GUID.
<code><count></code>	A <code><count></code> element is created for each type of client connected to this appliance.

`/connected_client_list/connected_client_summary/appliance_summary/count`

<code>type</code> (attribute)	The type of client connected to the appliance. Can be one of representative , support_customer , presentation_attendee , push_agent , or pinned_client .
-------------------------------	---

`/connected_client_list/connected_client`

<code>type</code> (attribute)	The type of client connected to one of the clustered appliances. Can be one of representative , support_customer , presentation_attendee , or push_agent .
<code>id</code> (attribute)	A unique identifier which remains valid only while the client is connected.

Query Examples: get_connected_client_list

Get a list of all connected clients	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_client_list</code>
Get a list of all connected representatives	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_client_list&type=representative</code>
Get a list of all connected representatives and support customers	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_client_list&type=representative,support_customer</code>
Get a summary of all connected clients	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_client_list&summary_only=1</code>
Get a summary of all connected representatives	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_client_list&summary_only=1&type=representative</code>
Get a summary of all connected representatives and support customers	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_client_list&summary_only=1&type=representative,support_customer</code>

API Command: get_connected_clients

The `get_connected_clients` command returns XML data containing details of all connected Bomgar clients.

In order to issue the `get_connected_clients` command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API**.

Required Parameters for get_connected_clients

<code>type=[string]</code>	The types of clients to return in the results. Can be a comma-separated list of values. Supported values are all (default), representative , support_customer , presentation_attendee , and push_agent .
<code>id=[string]</code>	The ID of the client. To get client IDs, see " API Command: get_connected_client_list " on page 24. Can be a comma-separated list of values. A maximum of 100 IDs is supported. This ID is a unique identifier which remains valid only while the client is connected.
<code>include_connections=[boolean]</code>	If this is set to 1 , then the client's list of connections to Bomgar Appliances and an event log about those connections will be included in the results.

XML Response for get_connected_clients

<code><connected_clients></code>	Contains a child element for each connected client, including <code><connected_representative></code> , <code><connected_support_customer></code> , <code><connected_presentation_attendee></code> , and <code><connected_push_agent></code> .
--	--

Element Names and Attributes

/connected_clients/connected_representative

<code>id</code> (attribute)	A unique identifier which remains valid only while the client is connected.
<code><client_connections></code>	Contains a <code><client_connections></code> element and an <code><event_log></code> element. This element is returned only if the query specifies include_connections .
<code><hostname></code>	The hostname of the representative's computer.
<code><platform></code>	The operating system of the representative's computer. Also contains an <code>id</code> attribute that briefly notes the selected platform for the client.
<code><timezone_offset></code>	The number of seconds away from UTC.
<code><connected_since></code>	The date and time at which this connection was made. Data is returned in ISO 8601 format. Also contains a <code>ts</code> attribute which displays the connection start time as a UNIX timestamp (UTC). This element is returned only if the query specifies include_connections .
<code><user_id></code>	Unique ID assigned to the representative.
<code><type></code>	The type of account the representative is using. Can be one of Normal , Embassy , or Invited .

<username>	The username assigned to the representative.
<public_display_name>	The public display name assigned to the representative. Note that this field contains the public display name's value at the time of the conference, which may not match the current value if the public_display_name has subsequently been changed.
<private_display_name>	The private display name assigned to the representative. Note that this field contains the private display name's value at the time of the conference, which may not match the current value if the private_display_name has subsequently been changed.
<start_session_url>	A URL that can be sent to a customer to start a support session with the representative.
<presentation_count>	The number of presentations the representative is performing. Can be either 0 or 1 .
<support_session_count>	The number of sessions the representative is participating in.
<showing_on_rep_list>	Integer value (1 or 0) indicating if the representative appears in the representative list on the public site.
<routing_idle>	Integer value (1 or 0) indicating if the representative has a status of idle.
<routing_busy>	Integer value (1 or 0) indicating if the representative has a status of busy.
<routing_enabled>	Integer value (1 or 0) indicating if the representative has automatic session assignment enabled or disabled.
<routing_available>	Integer value (1 or 0) indicating if the representative is available to have sessions automatically assigned.
<support_license>	The type of license used by the representative.
<support_session_lsids>	Contains an <lsid> element for each session in which the representative is participating. This field corresponds with the <lsid> field of the <connected_support_customer> element.

/connected_clients/connected_support_customer

id (attribute)	A unique identifier which remains valid only while the client is connected.
<client_connections>	Contains a <client_connections> element and an <event_log> element. This element is returned only if the query specifies include_connections .
<hostname>	The hostname of the customer's computer.
<platform>	The operating system of the customer's computer. Also contains an id attribute that briefly notes the selected platform for the client.
<timezone_offset>	The number of seconds away from UTC.
<connected_since>	The date and time at which this connection was made. Data is returned in ISO 8601 format. Also contains a ts attribute which displays the connection start time as a UNIX timestamp (UTC). This element is returned only if the query specifies include_connections .

<name>	The name which the customer entered in the Your Name field of the front-end survey or which was assigned programmatically.
<non_interactive>	Indicates if the session is a remote desktop protocol (RDP) session or a Shell Jump session. Can be either rdp or shelljump . If neither, this element is not returned.
<lsid>	A string which uniquely identifies this session. This field corresponds with the <lsid> field of the <connected_representative> element.

/connected_clients/connected_presentation_attendee

id (attribute)	A unique identifier which remains valid only while the client is connected.
<client_connections>	Contains a <client_connections> element and an <event_log> element. This element is returned only if the query specifies include_connections .
<hostname>	The hostname of the attendee's computer.
<platform>	The operating system of the attendee's computer. Also contains an id attribute that briefly notes the selected platform for the client.
<timezone_offset>	The number of seconds away from UTC.
<connected_since>	The date and time at which this connection was made. Data is returned in ISO 8601 format. Also contains a ts attribute which displays the connection start time as a UNIX timestamp (UTC). This element is returned only if the query specifies include_connections .
<name>	The name which the attendee entered when joining the presentation or which was assigned programmatically.

/connected_clients/connected_push_agent

id (attribute)	A unique identifier which remains valid only while the client is connected.
<client_connections>	Contains a <client_connection> element and an <event_log> element. This element is returned only if the query specifies include_connections .
<hostname>	The hostname of the Jumpoint's host computer.
<platform>	The operating system of the Jumpoint's host computer. Also contains an id attribute that briefly notes the selected platform for the client.
<timezone_offset>	The number of seconds away from UTC.
<connected_since>	The date and time at which this connection was made. Data is returned in ISO 8601 format. Also contains a ts attribute which displays the connection start time as a UNIX timestamp (UTC). This element is returned only if the query specifies include_connections .
<name>	The Jumpoint's name.

/client_connection

<appliance_id>	The GUID of the appliance to which the client is connected.
<purpose>	The reason the representative is connected to this appliance. Can be either master or traffic . If not part of a cluster, this will always be master .
<receive_traffic_node>	Integer value (1 or 0) indicating whether this is the client's default traffic node or not. If not part of a cluster, this will always be 0 .
<connected_since>	The date and time at which the client connected. Data is returned in ISO 8601 format. Also contains a ts attribute which displays the connection start time as a UNIX timestamp (UTC).
<private_ip>	The client's private IP address that was used to connect to the appliance.

/event_log

<event>	<p>An <event> element is created for each event that took place during this connection. Up to the last 20 events are returned.</p> <p>Events detail when and why a client connected to an appliance. Events also include failures to connect to nodes and normal disconnects.</p> <p>Includes a ts attribute which displays the timestamp of the event.</p>
---------	---

Query Examples: get_connected_clients

Get a detailed list of all connected clients	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_clients</code>
Get a detailed list of all connected representatives	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&type=representative</code>
Get a detailed list of all connected representatives and support customers	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&type=representative,support_customer</code>
Get a detailed list of all clients with IDs 101, 102, and 103	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&id=101,102,103</code>
Get a detailed list of all clients with IDs 101, 102, and 103 AND whose type is representative or customer	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&id=101,102,103&type=representative,support_customer</code>
Get a detailed list, with connection information, of all connected clients	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&include_connections=1</code>
Get a detailed list, with connection information, of all connected representatives	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&type=representative&include_connections=1</code>
Get a detailed list, with connection information, of all connected representatives and support customers	<code>https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&type=representative,support_customer&include_connections=1</code>

Get a detailed list, with connection information, of all clients with IDs 101, 102, and 103

```
https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&id=101,102,103&include_connections=1
```

Get a detailed list, with connection information, of all clients with IDs 101, 102, and 103 AND whose type is representative or customer

```
https://support.example.com/api/command?username=test&password=test&action=get_connected_clients&id=101,102,103&type=representative,support_customer&include_connections=1
```

API Command: check_health

The **check_health** command returns XML data containing information about the Bomgar Appliance, specifically including information needed for failover purposes.

In order to issue the **check_health** command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API** along with the permission **Is Administrator**.

XML Response for check_health Query

<appliance>	The hostname of the appliance. Also contains an id attribute that contains the appliance's GUID.
<version>	The version number and build number of the Bomgar software running on the appliance.
<success>	Integer value (1 or 0) indicating if the health check of the appliance was successful.
<error_message>	Returns an error message if a problem is found. If no error is found, this element will not be returned.
<failover_role>	The role the appliance plays in the failover relationship. Can be one of none (if failover is not configured), primary , or backup .
<enabled_shared_ips>	Contains an <ip> element for each IP address which is shared between the primary and backup appliances. If no shared IP addresses are enabled, this element will not be returned.
<last_data_sync_time>	The date and time at which the last data sync occurred between the primary and backup appliances. Data is returned in ISO 8601 format. Also contains a ts attribute which displays the data sync time as a UNIX timestamp (UTC).
<last_data_sync_status>	Contains a string showing the status of the last data sync.

Query Example: check_health

check_health	https://support.example.com/api/command?username=test&password=test&action=check_health
--------------	---

HTTP Status Check

In addition to using the API command above, you can use https://support.example.com/check_health to check the health of an appliance. This returns an HTTP status of 200 if the probe is successful and 500 (Server Error) if not. While you will see a simple human-readable message showing success or failure, no other data is exposed.

API Command: set_failover_role

The `set_failover_role` command sets the failover role of an appliance to either primary or backup.

In order to issue the `set_failover_role` command, you will need to supply the username and password for a Bomgar user account. That account must have the permission **Allowed to Use Command API** along with the permission **Is Administrator**.

Required Parameter for set_failover_role

<code>role=[string]</code>	The role to assign to this appliance. Can be either primary or backup .
----------------------------	---

Optional Parameters for set_failover_role

<code>data_sync_first=[boolean]</code>	To perform a data sync with the peer appliance before failing over, set this to 1 . All users on the existing primary appliance will be disconnected during the data sync, and no other operations will be available until the swap is complete. To fail over without a final data sync, set this to 0 .
<code>force=[boolean]</code>	This option is only applicable when contacting the primary appliance and attempting to set its role to backup. If this is set to 1 , then this appliance will become the backup even if the peer appliance cannot be contacted.

XML Response for set_failover_role Query

<code><success></code>	If a data sync is being performed first, returns a message of Successfully started data sync. Role change will occur upon successful completion . Otherwise, returns a message of Successfully changed role .
<code><error></code>	Returns an error message if the role was not set successfully.

Query Examples: set_failover_role

Set failover role to primary	<code>https://support.example.com/api/command?username=test&password=test&action=set_failover_role&role=primary</code>
Set failover role to backup	<code>https://support.example.com/api/command?username=test&password=test&action=set_failover_role&role=backup</code>
Set failover role to primary and perform a data sync	<code>https://support.example.com/api/command?username=test&password=test&action=set_failover_role&role=primary&data_sync_first=1</code>
Set failover role to backup and perform a data sync	<code>https://support.example.com/api/command?username=test&password=test&action=set_failover_role&role=backup&data_sync_first=1</code>
Set failover role to backup even if the primary appliance cannot be contacted	<code>https://support.example.com/api/command?username=test&password=test&action=set_failover_role&role=backup&force=1</code>

Set failover role to backup even if the primary appliance cannot be contacted, and perform a data sync

```
https://support.example.com/api/command?username=test&password=test&action=set_failover_role&role=backup&data_sync_first=1&force=1
```

API Command: get_api_info

The `get_api_info` request returns XML data containing the current API version information.

XML Response for get_api_info Query

<api_version>	The software version of the current Bomgar API.
<timestamp>	The server's current timestamp at the time this report was pulled.
<permissions>	The permissions of the user account used to issue this command. The permissions shown are detailed below.
<user_id>	The numeric rep ID of the Bomgar user making this API call.

Element Names and Attributes

<i>/get_api_info/permissions/permission</i>	
perm_use_command_api	Integer value (1 or 0) indicating if the user has permission to use the command API.
perm_use_reporting_api	Integer value (1 or 0) indicating if the user has permission to use the reporting API.
perm_admin	Integer value (1 or 0) indicating if the user is an administrator.
perm_view_reports	Indicates if the user has permission to view reports. Can be one of the following: <ul style="list-style-type: none"> none Cannot view any reports. user_sessions Can view reports in which he or she was the primary representative. team_sessions Can view reports in which one of the user's teammates was the primary representative or one of the user's teams was the primary team. all_sessions Can view all reports.
perm_view_sd_recordings	Integer value (1 or 0) indicating if the user has permission to view support session recordings.
perm_sd_allowed	Integer value (1 or 0) indicating if the user is allowed to provide remote support.

Query Example: get_api_info

get_api_info	https://support.example.com/api/command?username=test&password=test&action=get_api_info
--------------	---

Representative Console Scripting and Client Scripting API

The Bomgar Representative Console scripting feature is composed of three parts:

1. The Bomgar Rep Console Script file format
2. New and deprecated command line parameters for the rep console
3. The Bomgar client scripting API

The Bomgar Representative Console Script File

A Bomgar Representative Console Script (BRCS) is a file that contains a sequence of commands to be executed by the Bomgar Rep Console. The file extension is in the format "brcs-*<companySiteName>*" (Company Site Name is the name used to access your support site). During installation the Bomgar Rep Console will use the OS to associate the rep console with the BRCS file type. Therefore users can double-click a BRCS file and have it automatically executed by the Bomgar Rep Console.

BRCS files have the following format:

```
BRCS1.0
<command>
<command>
...
```

This is more formally expressed as:

```
brcs_file = header , newline , commands ;
header = "BRCS" , version ;
version = digit , "." , digit ;
commands = command { newline , command } ;
digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;
newline = "\n" | "\r\n" ;
```

Note that script files can have a maximum of 10 commands.

Each command consists of a set of key-value pairs separated by "&". The key in each pair is separated from the value by "=". Keys and values use the percent-encoding algorithm described in [RFC3986 section 2.1](#). This is commonly referred to as url-encoding or url-escaping. It is commonly seen in the address bar of web browsers to represent the parameters passed to a web server. Commands have the following format:

```
action=<action>&parameter1=value1&parameter2=value2...
```

This is more formally expressed as:

```
command = "action=", value, [ parameters ] ;
parameters = "&", parameter, [ parameters ] ;
parameter = url_encoded_string, "=", url_encoded_string ;
```

```
url_encoded_string = { * see RFC 3986 * } ;
```

New and Deprecated Command Line Parameters for the Representative Console

Two command line parameters have been added to the representative console to support BRCS:

```
run-script <BRCS command>
run-script-file <path to BRCS file>
```

These command line parameters allow customers to implement BRCS login via the command line. These two new parameters overlap with two existing parameters. Therefore, the "-jump" and "-push" command line parameters are now deprecated and will be removed in a future release.

Example

Old Command Line	New Command Line
push<hostname>	run-script "action=push_and_start_local&hostname=<hostname>"
jump<search string>	run-script "action=start_pinned_client_session&search_string=<search_string>"

Different behaviors can be seen when running a script from the command line depending on the state of the representative console:

- If the representative console is not running, then attempting to run a script from the command line causes the representative console to start the login dialog. After the representative successfully logs in, the script is run.
- If the representative console is already running, but the representative is not logged in, then the login dialog is shown. After the representative logs in, the script is run.
- If the representative console is already running and the representative is already logged in, then attempting to run a script from the command line causes the existing instance of the representative console to run the script.

Representative console exit status:

- If an invalid script is given on the command line, then the representative console will terminate with an exit status > 0.
- If a valid script is given on the command line, then the representative console will terminate with an exit status of 0.

Examples:

```
bomgar-rep-x64.exe --run-script "action=generate_session_key&session.custom.external_key=123456789"
bomgar-rep-x64.exe --run-script-file my_script_file.brsc-beta60
```

The Bomgar Client Scripting API

The client scripting API enables you to generate a Bomgar Representative Console Scripting (BRCS) file which allows you to send commands to the Bomgar Rep Console from external applications. The client scripting API was introduced in API version 1.6.0 and BRCS functionality was introduced in 11.1.

Customers can use the client scripting API to generate BCRC files that can start a support session with a specific Jump Client, push and start a session with a Windows system within a local network, prompt representatives to generate a session key, start a vPro session with a specified system, or to simply log into the representative console.

The client scripting API URL is https://support.example.com/api/client_script.

This API accepts a client type (**rep**), an operation to perform (**generate**), a command to put in the script file, and a set of parameters to pass to the command. Here is an example of a valid Client Scripting API request:

```
https://support.example.com/api/client_script?type=rep&operation=generate&action=start_pinned_client_session&search_string=ABCDEFG02
```

The above request will prompt the user to download a Bomgar representative console script file. After downloading the script file, the user can run it using the representative console. In this case, the script file will contain commands to start a session with the Jump Client whose hostname, comments, public IP, or private IP matches the search string "ABCDEFG02".

Note: By default, access to the API is SSL-encrypted; however, you can choose to allow HTTP access by checking the **Allow HTTP Access to XML API** option on the **Management > API Configuration** page of the **/login** administrative interface. **It is highly recommended that HTTP remain disallowed as a security best practice.**

Required Parameters for Client Scripting API

type=rep	The Bomgar client to which the command applies. Currently the API only supports rep as the client type.
operation=generate	The operation to perform. Currently the API only supports generate as the operation.
action=<command> or action=<command>¶meter=[value]	The name of the command to run. Beginning with API version 1.7.1 and Bomgar 12.1.4, two actions are automatically added to the BRCS file: login and delete_script_file . See below for the full list of available commands.

Parameters and Values Available for action=<command>

action=start_pinned_client_session&	search_string=[string] session.custom.external_key =[string]
action=start_vpro_session&	target=[string] jumpoint=[strinstarting] session.custom.external_key=[string] (optional)

Note: To initiate a vPro session using Bomgar Client Scripting, you must specify the target machine's hostname or private IP. (If Kerberos is used for vPro authentication, then the FQDN must be specified.) The Jumpoint name must also be specified. An example is as follows: `action=start_vpro_session&target=mycomputer&Jumpoint=My%Jumpoint`

<p>action=generate_session_key&</p>	<p>session.custom.external_key=[string] public_portal_hostname=[string]</p> <div data-bbox="618 401 1508 531" style="border: 1px solid orange; padding: 5px;"> <p>Note: Parameters are optional for the generate_session_key command. Omitting them will simply cause the representative console to show the Generate Session Key dialog.</p> </div>
<p>action=push_and_start_local&</p>	<p>hostname=[string] session.custom.external_key=[string]</p>
<p>action=login&</p>	<p>mechanism=username_password "currently only username_password is supported" username=[string] myusername&mypassword=[string]</p> <div data-bbox="618 831 1508 1171" style="border: 1px solid orange; padding: 5px;"> <p>Note: Command attempts to use last saved credentials. Command has no effect if the representative console is already logged in. This must be the first command in a script file. Only one login command is appropriate per BRCS file. This command will pass the login mechanism and, optionally, a username and password. If no options are specified command will simply launch the rep console and attempt to login with any save credentials. The mechanism parameter currently only supports username_password as its argument. The username and password parameters are required, only if mechanism == username_password. Both username and password parameters are sent in plain text, unencrypted.</p> </div>
<p>action=delete_script_file</p>	<p>This command has no parameters</p>

API Script Command: start_pinned_client_session

The **start_pinned_client_session** command attempts to start a support session with a Bomgar Jump Client. Representatives may run this command for all Jump Clients they are permitted to access via the Jump Client management interface in the representative console.

Required Parameters for the start_pinned_client_session Command

search_string =[string]

The search criteria used to select a Jump Client. The private IP, public IP, hostname, and comments fields of the Jump Client will be matched against the search string. This field has a maximum length of 1024 characters.

Optional Parameters for the start_pinned_client_session Command

session.custom.external_key =[string]

The external key to associate with the support session spawned from the Jump Client. This field has a maximum length of 1024 characters.

Query Examples: start_pinned_client_session

Start a support session with the Jump Client whose hostname is "ABCDEFGH02"

https://support.example.com/api/client_script?type=rep&operation=generate&action=start_pinned_client_session&search_string=ABCDEFGH02

Start a support session with the Jump Client whose comments field contains the string "Example Co"

https://support.example.com/api/client_script?type=rep&operation=generate&action=start_pinned_client_session&search_string=Example%20Co

or

https://support.example.com/api/client_script?type=rep&operation=generate&action=start_pinned_client_session&search_string=Example+Co

Start a support session with the Jump Client whose private IP address or public IP address is "192.168.9.14" and assign the external key "BMC0000001275" to the session:

https://support.example.com/api/client_script?type=rep&operation=generate&action=start_pinned_client_session&search_string=192.168.9.14&session.custom.external_key=BMC0000001275

Note: If more than one Jump Client matches the search criteria then a dialog will open , giving the user the option to select the appropriate Jump Client.

API Script Command: generate_session_key

In the context of the client scripting API, the **generate_session_key** command causes the representative console to show the Generate Session Key dialog. Parameters can be passed to the command to customize the behavior.

Optional Parameters for the generate_session_key Command

session.custom.external_key=[string]	The external key to associate with the support session started with the session key or URL shown on the Generate Session Key dialog. This field has a maximum length of 1024 characters.
public_portal_hostname=[string]	The hostname of the public portal that should be selected by default when the Generate Session Key dialog is shown. The representative will still have the option to change the public portal on the dialog. This field has a maximum length of 255 characters.

Query Examples: generate_session_key

Show the Generate Session Key dialog	<code>https://support.example.com/api/client_script?type=rep&operation=generate&action=generate_session_key</code>
Show the Generate Session Key dialog with the public portal hostname "support.example.com" selected	<code>https://support.example.com/api/client_script?type=rep&operation=generate&action=generate_session_key&public_portal_hostname=support.example.com</code>
Show the Generate Session Key dialog and associate the external key "BMC0000001275" with any support sessions started using the session key or URL shown on the dialog	<code>https://support.example.com/api/client_script?type=rep&operation=generate&action=generate_session_key&session.custom.external_key=BMC0000001275</code>

API Script Command: push_and_start_local

The **push_and_start_local** command attempts to push the customer client to a computer on the local network to start a support session. This can also be described as a local Jump.

Required Parameter for push_and_start_local

hostname=[string]

The hostname of the computer that is the target of the push and start operation. This field has a maximum length of 255 characters.

Optional Parameter for push_and_start_local

session.custom.external_key=[string]

The external key to associate with the support session started with the session key or URL shown on the Generate Session Key dialog. This field has a maximum length of 1024 characters.

Query Examples: push_and_start_local

Jump to the local network computer called "ABCDEFGH02"

`https://support.example.com/api/client_script?type=rep&operation=generate&action=push_and_start_local&hostname=ABCDEFGH02`

Jump to the local network computer called "ABCDEFGH02" and pass an external key

`https://support.example.com/api/client_script?type=rep&operation=generate&action=push_and_start_local&hostname=ABCDEFGH02&session.custom.external_key=BMC0000001275`

API Script Command: start_vpro_session

The **start_vpro_session** command initiates a vPro session on the target machine using the specified Jumpoint and optional external key.

Required Parameters for the start_vpro_session Command

target=[string]	The hostname or IP address of the machine targeted for a vPro session.
jumpoint=[string]	The Jumpoint configured for vPro support.

Note: To initiate a vPro session using Bomgar Client Scripting, you must specify the target machine's hostname or private IP. (If Kerberos is used for vPro authentication, then the FQDN must be specified.) The Jumpoint name must also be specified. An example is as follows: `action=start_vpro_session&target=mycomputer&Jumpoint=My%Jumpoint`
See query example below.

Optional Parameters for the start_vpro_session Command

session.custom.external_key=[string]	The external key to associate with the support session started with the session key or URL shown on the Generate Session Key dialog. This field has a maximum length of 1024 characters.
--------------------------------------	--

Query Examples: start_vpro_session

Use the start_vpro_session command	<code>https://support.example.com/api/client_script?type=rep&operation=generate&action=start_vpro_session&target=ABCDEFGHG02.example.com&jumpoint=ABC01&session.custom.external_key=BMC0000001275</code>
---	--

Session Generation API

The public site is a collection of forms and links which generates an HTTP request each time a session is requested, resulting in the customer client's being downloaded to the remote computer. The request generated depends on the configuration of the public site and the method used by the customer to request support:

1. By selecting a name from a list of logged-in representatives
2. By entering a unique session key
3. By submitting an issue

The session generation API URL is https://support.example.com/api/start_session

The queue in which to place the customer after connecting can be specified in one of three ways:

Session Generation by Representative Name and ID

id=[integer]	The numeric ID for the representative with whom to start the session. To get a representative's ID, see "API Command: get_logged_in_reps" on page 11 .
name=[string]	The public display name for this same representative.

Session Generation by Session Key

short_key=[string]	The seven-character string used to start a session.
--------------------	---

Session Generation by Issue Submission Survey

issue_menu=[integer]	Must be set to 1.
codeName=[string]	The code name of the selected issue. Use either codeName or id , not both. If neither codeName nor id is specified, the session will go to the general queue if this queue is enabled.
id=[integer]	The numeric ID for the selected menu item. If Display Reps in Issues Menu is enabled from the Public Site Configuration page of the /login web interface, this will be the unique ID for a representative in the list. Otherwise, it will be the unique ID for an issue found in the issues list. To get a representative's ID, see "API Command: get_logged_in_reps" on page 11 . To get a list of valid issue IDs, see "API Command: get_support_teams" on page 13 . Use either codeName or id , not both. If neither codeName nor id is specified, the session will go to the general queue if this queue is enabled.

You may also add optional parameters to pass additional information to the session.

Optional Parameters

c2cjs=[integer]	If set to 1, causes the session to start as a click-to-chat session. This method of starting a click-to-chat session is less preferred to using JavaScript. (See " Start Sessions with Click-To-Chat or Collaborative Browser Sharing " on page 47.)
customer.name=[string]	Customer's name (maximum of 100 characters).
customer.company=[string]	Customer's company name (maximum of 100 characters).
customer.company_code=[string]	Customer's company code (maximum of 100 characters).
customer.details=[string]	Customer's problem description (maximum of 1024 characters).
locale=[string]	The locale code of the language to be used for this customer client. The language must be installed on your Bomgar Appliance. To see which languages are installed, go to /login > Localization > Languages . Available language packages may include English (en-us), German (de), Latin American Spanish (es), EU Spanish (es-sp), EU French (fr), Italian (it), Dutch (nl), Brazilian Portuguese (pt-br), EU Portuguese (pt-pt), Swedish (sv-se), Turkish (tr), Japanese (ja), Simplified Chinese (zh-hans), and Traditional Chinese (zh).
platform=[string]	The specific Windows® platform (such as Windows 32-bit or 64-bit) for which to download the customer client. Use the correct parameters for the desired platform, winNT-32 or winNT-64 .
session.custom.external_key=[string]	A key to an external help desk ticket system (maximum of 1024 characters).
session.custom.[custom field]=[string]	The code name and value of any custom fields. These fields must first be configured in /login > Management > API Configuration . Each custom field has a maximum length of 1024 characters. The maximum total size of all combined custom fields, including the external key, must be limited to 10KB.
session.priority=[integer]	The priority of the session, from 1 to 3 . 1 = high, 2 = medium, and 3 = low. The priority set by the API overrides the priority already set for the selected issue. If this parameter is not set by the API, the priority assigned to the selected issue will be used.
session.skills=[string]	A comma-separated list of the code names of skills to assign to a session. The skills set by the API override the skills already set for the selected issue. If this parameter is not set by the API, the skills assigned to the selected issue will be used.

Note: The parameters **customer_name**, **customer_company**, **customer_company_code**, **customer_desc**, and **external_key** have been deprecated as of 14.2 and have been replaced with the parameters above. However, they are still available for backward compatibility.

Note that if your Bomgar Appliance has multiple public sites, the session created will be associated with the public site whose domain name matches the request's domain name. For example, if Site A has a hostname of support.example.com and Site B has

a hostname of remote.example.com, a session generation request made to support.example.com will create a session associated with Site A.

Also note the session key behavior if the option to prompt customers before downloading the customer client is enabled from **/login > Public Portals > Public Sites**. If prompting is enabled, then only the external key can be passed along with the session key. If the prompt is disabled, you can include other parameters with the session key.

Query Examples

Specific representative	<code>https://support.example.com/api/start_session?id=1&name=Admin</code>
Session key	<code>https://support.example.com/api/start_session?short_key=1234567&session.custom.external_key=1234</code>
Issue submission survey using issue ID	<code>https://support.example.com/api/start_session?issue_menu=1&customer.name=John%20Doe&customer.company=Company%20Name&customer.company_code=1234&customer.details=!%20need%20support&id=1&session.custom.external_key=1234&session.custom.custom_field1=Custom%20Value</code>
Issue submission survey using issue code name	<code>https://support.example.com/api/start_session?issue_menu=1&customer.name=John%20Doe&customer.company=Company%20Name&customer.company_code=1234&customer.details=!%20need%20support&codeName=issue_code_name&session.custom.external_key=1234&session.custom.custom_field1=Custom%20Value</code>
Issue submission survey using ID, with skills and priority	<code>https://support.example.com/api/start_session?issue_menu=1&customer.name=John%20Doe&customer.company=Company%20Name&customer.details=!%20need%20support&id=1&session.priority=1&session.skills=codename1,codename2</code>
Locale	<code>https://support.example.com/api/start_session?id=1&name=Admin&locale=en-us</code>
Platform	<code>https://support.example.com/api/start_session?id=1&name=Admin&platform=winNT-32</code>

Start Sessions with Session Key Acceptance

An alternative method of starting a session is to create a web form where your customers can enter short session key strings to start sessions with you.

To create a session key entry form, create a web form with the action of https://support.example.com/api/start_session and a method of either **GET** or **POST**. You must also use a text box with the name of **short_key**, as shown in the example below.

```
<form action="https://support.example.com/api/start_session" method="get">
  Session Key: <input type="text" name="short_key" /><br />
  <input type="submit" value="Submit" />
</form>
```

You may also include an external key to start a session.

```
<form action="https://support.example.com/api/start_session" method="get">
  Session Key: <input type="text" name="short_key" /><br />
  External Key: <input type="text" name="session.custom.external_key" /><br />
  <input type="submit" value="Submit" />
</form>
```

Using this form, your customer can enter a generated seven-character session key and an optional external key to start a session with you.

Note: The **external_key** parameter has been deprecated as of 14.2 and has been replaced with **session.custom.external_key**. However it is still available for backward compatibility.

Start Sessions with Click-To-Chat or Collaborative Browser Sharing

Using JavaScript

To start a collaborative browser sharing session or a click-to-chat session, you must first reference an external JavaScript file that is included on your Bomgar Appliance. You must then tell the API the hostname from which the JavaScript files and other resources should be lazily loaded. This hostname should not include the protocol (e.g., support.example.com). Both of these elements should be included in the head of your web page, as shown in the example below.

```
<head>
  <script type="text/javascript" src="https://support.example.com/api/start_session.js"></script>
  <script type="text/javascript">BG.setSite("support.example.com");</script>
</head>
```

Then, within the body, you must include an event attribute to trigger a session start. In most cases, this will be an **onclick** event attribute on an anchor or button element. With this event, call the **BG.start** method, using the arguments to start the session with session key submission, representative selection, or issue submission.

```
BG.start(startType, options)
```

The `startType` can be either **BG.START_TYPE.CHAT** or **BG.START_TYPE.BROWSER**. This determines which type of session will be launched. The `options` parameter expects a generic JavaScript object which contains set properties (see "[options Properties](#)" on page 51).

IMPORTANT! Your HTML page must have a valid standards-compliant Doctype. View recommended Doctype declarations at <http://www.w3.org/QA/2002/04/valid-dtd-list.html>.

Below are several examples of the script.

Examples

Several common examples are listed below, though more are possible.

Starting a Browser Session with a Session Key

```
BG.start(BG.START_TYPE.BROWSER, {
  sessionKey: '1728331'
});
```

Starting a Browser Session with Representative Information

```
BG.start(BG.START_TYPE.BROWSER, {  
  rep: {  
    id: 30,  
    name: 'Admin'  
  }  
});
```

Starting a Browser Session with a Session Key and Custom Fields

```
BG.start(BG.START_TYPE.BROWSER, {  
  sessionKey: '8679485',  
  attributes: {  
    external_key: 'abc123',  
    custom_field1: 'Custom Value',  
    custom_field2: 123  
  }  
});
```

Starting a Browser Session with a Session Key and a Specified Language

```
BG.start(BG.START_TYPE.BROWSER, {  
  sessionKey: '8679485',  
  locale: 'en-us'  
});
```


Starting a Browser Session with an Issue Object (by ID) and Customer Information

```
BG.start(BG.START_TYPE.BROWSER, {
  issue: {
    id: 12
  },
  customer:
  {
    name: 'Customer Name',
    company: 'Company Name',
    company_code: 'Company Code',
    details: 'Issue Details'
  }
});
```

Starting a Browser Session with an Issue Object (by Code Name) and Skills

```
BG.start(BG.START_TYPE.BROWSER, {
  issue: {
    codeName: 'Other'
  }
  skills: ["skill11", "skill12"]
});
```

Starting a Browser Session with an Issue Form Element

```
BG.start(BG.START_TYPE.BROWSER, {  
  issue: document.getElementById('formID')  
});
```

Example Issue Form

```
<form id="id">  
  <input type="hidden" name="codeName" value="Other" />  
  <input type="hidden" name="skills" value="skill1,skill2" />  
  <input type="hidden" name="customer.name" value="Customer Name" />  
  <input type="hidden" name="customer.company" value="Company Name" />  
  <input type="hidden" name="customer.company_code" value="Company Code" />  
  <input type="hidden" name="customer.details" value="Issue Details" />  
  <input type="hidden" name="session.custom.external_key" value="abc123" />  
  <input type="hidden" name="session.custom.custom_field1" value="Custom Value" />  
  <input type="hidden" name="session.custom.custom_field2" value="123" />  
</form>
```

*Any of the sessions above can be called as click-to-chat sessions by changing the start type to **BG.START_TYPE.CHAT**.*

Starting a Click-to-Chat Session with a Session Key

```
BG.start(BG.START_TYPE.CHAT, {  
  sessionKey: '8347482'  
});
```

*Click-to-chat sessions may have an additional **uiOptions** object.*

Starting a Click-to-Chat Session with a Session Key and fallbackToFullWindow

```
BG.start(BG.START_TYPE.CHAT, {  
  sessionKey: '7683902',  
  uiOptions: {  
    fallbackToFullWindow: false  
  }  
});
```

options Properties

The **options** parameter accepts the following possible arguments:

Name	Type	Required or Exclusive?	Description
sessionKey	String	Exclusive – Start Method	The numeric session key.
rep	Object	Exclusive – Start Method	An object identifying the representative with whom to start the session. See the table below.
issue	DOM Element	Exclusive – Start Method	A DOM element specifying the support issue. The DOM element can be retrieved using document.getElementById('id') . See the table below.
issue	Object	Exclusive – Start Method	An object specifying the support issue. See the table below.
skills	Array	No	The skills to apply to this session for the purpose of routing. Can be combined with any start method.
customer	Object	No	An object providing information about the customer. Can be combined with any start method. See the table below.
locale	String	No	The locale code of the language to be used for this customer client. The language must be installed on your Bomgar Appliance. To see which languages are installed, go to /login > Localization > Languages . Available language packages may include English (en-us), German (de), Latin American Spanish (es), EU Spanish (es-sp), EU French (fr), Italian (it), Dutch (nl), Brazilian Portuguese (pt-br), EU Portuguese (pt-pt), Swedish (sv-se), Turkish (tr), Japanese (ja), Simplified Chinese (zh-hans), and Traditional Chinese (zh).
attributes	Object	No	Session attributes. Can be combined with any start method. See the table below.
uiOptions	Object	No	User interface options. Available only for Click-to-Chat sessions. See the table below.

Note: Only one of the properties marked as **Exclusive – Start Method** should be specified.

rep Properties

The **rep** object must have the following properties:

Name	Type	Required or Exclusive?	Description
id	Integer	Required	The representative's unique ID number. To get a representative's ID, see "API Command: get_logged_in_reps" on page 11 .
name	String	Required	The representative's public display name.

issue Properties

The **issue** object may be a JavaScript object with defined properties. Alternatively, it may be a DOM element, which should be a form. This DOM element must have one or more child inputs with defined names. If unnecessary elements are within the form, they will be ignored by the server. In either case, the accepted properties or input names are:

Name	Type	Required or Exclusive?	Description
id	Integer	Exclusive – Issue Identifier	The support issue's unique ID number. To get a list of issue IDs, see "API Command: get_support_teams " on page 13.
codeName	String	Exclusive – Issue Identifier	The support issue's unique code name.

Note: Only one of the properties marked as **Exclusive – Issue Identifier** should be specified.

Note: The **issue** properties **customerName**, **companyName**, **companyCode**, and **details** have been deprecated as of 14.2 and have been replaced with the **customer** properties below. However, they are still available for backward compatibility.

customer Properties

The **customer** object has the following properties:

Name	Type	Required or Exclusive?	Description
name	String	No	The customer's name.
company	String	No	The customer's company name.
company_code	String	No	The customer's company code.
details	String	No	A description of the customer's problem.

attributes Properties

The **attributes** object has the following properties:

Name	Type	Required or Exclusive?	Description
external_key	String	No	The external key to associate with the session.
[custom field]	String	No	The code name and value of any custom fields. These fields must first be configured in /login > Management > API Configuration .

Note: The attribute **externalKey** has been deprecated as of 14.2 and has been replaced with **external_key**. However, it is still available for backwards compatibility.

uiOptions Properties

The **uiOptions** object has the following property:

Name	Type	Required or Exclusive?	Description
fallbackToFullWindow	Boolean	No	Only used with click-to-chat sessions. If true , then a full-screen browser window will be used to render the click-to-chat client if a pop-up window cannot be created.

Collaborative Browser Sharing Functionality

Starting a collaborative browser sharing session using the API methods above will open a dialog box in the customer's browser, determine the optimal download method for this client, and initiate the download. If the Java applet is launched, then the dialog will automatically close once the download and execution are complete. In all other cases, the user should close the dialog by clicking outside of it. The experience will vary depending on which download mechanism JavaScript has determined will work best. One of four different displays will be shown:

- Java applet download – A Java applet will launch, which will attempt to download and run the Bomgar customer client.
 - If the user has previously selected to remember the Bomgar customer client applet (even from another site), then the applet will launch with no prompts.
 - Otherwise, the user will be prompted to run the applet or to cancel.
 - If the user chooses to run the applet, then the applet will proceed to launch the customer client.
 - If the user chooses to cancel, the launch method will fall back to the JavaScript launch.
- JavaScript launch – If the user does not have Java installed or if the user cancels the Java applet, the launch method will fall back to JavaScript.
 - JavaScript will tell the browser to download the Bomgar customer client, which will pop up a browser download dialog along with instructions for completing the download.
 - If the user cancels the JavaScript download dialog, the launch method will fall back to the manual launch.
- Manual launch – If the user cancels all dialogs, he or she can click a link to re-trigger the download.
- Mobile display – Because collaborative browser sharing is not currently supported on mobile devices, users will receive a message indicating that browser sharing is not available on mobile devices.

The Java applet requires Java 5+. Users who do not already have Java will not be prompted to download or install Java; the launch method will merely fall back to another download mechanism. If Java cannot be reliably detected, it will be assumed that Java is unavailable.

Click-to-Chat Without Using JavaScript

If you need to start a session with click-to-chat from an external site without writing any JavaScript, you may add the parameter **c2cjs=1** to any of the documented **start_session** API URLs. This will cause the request to respond with a click-to-chat page instead of the full customer client download, regardless of the settings for the public site.

For example, to redirect the current page to start a click-to-chat session with a specific representative:

```
<a href="https://support.example.com/api/start_session?id=12&name=John&c2cjs=1">Chat with John</a>
```

To open click-to-chat for a specific representative in a new browser tab – not a new window – in most browsers:

```
<a href="https://support.example.com/api/start_session?id=12&name=John&c2cjs=1" target="_blank">Chat with John</a>
```

Please note that if you wish to open click-to-chat in a new, smaller browser window instead of the current window or a new browser tab, you must either use **start_session.js** or write your own JavaScript to create and correctly size the new window.

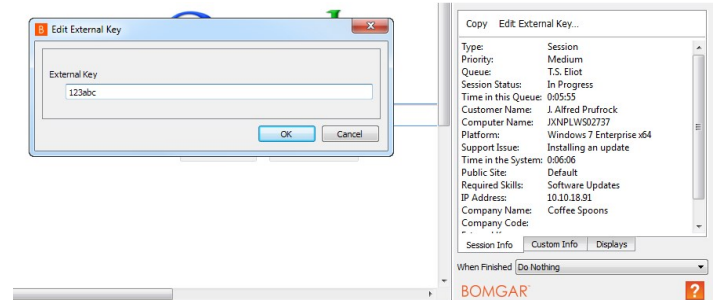
Note: For the sake of appearance, opening click-to-chat in an appropriately sized window is the recommended method. A window that is not properly sized will function correctly but will result in disproportionate margins.

Start Sessions with External Keys (TicketID)

The external key is a text string sent to the Bomgar Appliance to be logged as a property of a particular support session. This key typically originates from software separate from the Bomgar software. It does not need to be a unique value but usually is. The external key can be specified in one of two ways.

Manual Entry

The support representative who has ownership of a Bomgar session can manually assign a key value from within the representative console. The summary pane of a session displays that session's external key. Click the **Edit External Key** button to modify this value. You can edit the external key from either the **Session Info** tab or the **Custom Info** tab. Editing the external key in one location will change it in both.



Programmatic Assignment

The second, more useful way of designating an external key is from within the URL sent to the Bomgar Appliance by the customer client. The external key can be specified with any of the session start methods documented above, as well as with the **generate_session_key** command detailed below.

The issue submission survey generates an HTTP request similar in format to the following example:

```
https://support.example.com/api/start_session?issue_
menu=1&customer.name=John%20Doe&customer.company=Company%20Name&customer.company_
code=1234&customer.details=I%20need%20support&id=1&session.custom.external_key=1234
```

Note the **session.custom.external_key** parameter specified in the sample request provided. If an external key is specified in this manner, the representative console will automatically populate its external key field with the given value.

Note: The **external_key** parameter has been deprecated as of 14.2 and has been replaced with **session.custom.external_key**. However it is still available for backward compatibility.

The API allows creation of a custom web site or application that can be used instead of the public site to establish a support session. Code within this custom software must generate HTTP requests in the format displayed in the example above to initiate the session and pre-populate the external key within the representative console.

Using the External Key

Once a support session has an external key associated with it, you can use the reporting API to generate XML session data containing the external key. This provides the means for middleware to be developed to provide a relationship between the reporting data used by Bomgar and the correlating ticketing system's ticket numbers.

Start Sessions with an Embedded Bomgar Button

Define custom links within your in-house developed applications to call a pre-installed Bomgar Button. An embedded Bomgar Button gives support providers the ability to streamline the support path for specific applications.

The command line options can leverage the same session initiation methods as configured in the Bomgar Button profile, or they can bypass the user interface altogether. You can configure this embedded Bomgar Button to point to a specific issue so that when a customer clicks the button, a session will immediately start with the team best suited to handle that type of problem. The command also can assign an external key to sessions started from this embedded Bomgar Button.

To create an embedded Bomgar Button, a Bomgar Button must first be deployed on the remote system. You may wish to define the Bomgar Button profile so that neither the desktop shortcut nor the menu shortcut is created. Because a Bomgar Button must be pre-installed, using an embedded Bomgar Button prevents users from having to download the customer client each time they request support. Embedded Bomgar Buttons are a Windows-only feature.

To start an embedded Bomgar Button programmatically, first open the registry editor and locate the Bomgar Button registry entry. Then copy the value data. The registry entry can be found in one of two places:

Single User Install: HKEY_CURRENT_USER\Software\Bomgar\CallbackButton\<site_name>\cb-install-cmdline

All Users Install: HKEY_LOCAL_MACHINE\Software\Bomgar\CallbackButton\<site_name>\cb-install-cmdline

After pasting the value data into a text editor or the command prompt, add optional arguments.

Embedded Bomgar Button Command Line Options

<code>--session-value <variable_name> <value></code>	Currently, the only supported variable is session.custom.external_key .
<code>--issue-code-name</code>	This selects the issue for the support session. If --issue-code-name is provided without --present-front-end-survey , a session started from this Bomgar Button will start immediately without offering the customer the issue submission form or any additional session start methods.
<code>--present-front-end-survey</code>	If provided with --issue-code-name , a session started from this Bomgar Button will take the customer to the issue submission form with the issue pre-selected. The customer will not be given any additional session start methods.

Finally, run the command line.

Example:

```
C:\Users\admin\AppData\Local\bomgar-scc-cb\support.example.com\bomgar-cb-
w0dc30ji8hz65yji8hz65yji8hz65yji8hz65yc40ic90.exe --cbdir C:\Users\admin\AppData\Local\bomgar-scc-
cb\support.example.com\ --session-value session.custom.external_key "abc123" --issue-code-name
install_update --present-front-end-survey
```

Note: The command line option **--cbdir <bomgar_button_install_directory>** is required.

Reporting API

The Bomgar reporting API is designed to enable you to pull reporting data in XML format, suitable for importing into external databases and applications. The data presented is the same as in the session and exit survey reports of the **/login** administrative interface.

XML data is pulled by sending a simple HTTP request to the Bomgar Appliance. The request can be sent using any HTTPS-capable socket library or scripting language module, a web browser, or a URL fetcher such as **cURL** or **wget**. Either **GET** or **POST** may be used as the request method. Note that even if your Bomgar Appliance has multiple public sites, all reports return data associated with all public sites unless the request contains a specific parameter to limit the sites pulled.

IMPORTANT! When making consecutive API calls, you must close the connection after each API call.

Note: By default, access to the API is SSL-encrypted; however, you can choose to allow HTTP access by checking the **Allow HTTP Access to XML API** option on the **Management > API Configuration** page of the **/login** administrative interface. **It is highly recommended that HTTP remain disallowed as a security best practice.**

The reporting API URL is <https://support.example.com/api/reporting>.

An XML schema which formally describes the format of the returned reporting data is available at <https://support.example.com/api/reporting.xsd>.

In order to issue an API request to the reporting API, you will need to supply the username and password for a Bomgar user account. That account must have the reporting permission **Allowed to use reporting API**. The account must also have one or more of the following permissions, depending upon which type of reports you wish to run: **Allowed to View Support Session Reports** for only his or her sessions, his or her teams' sessions, or all sessions; **Allowed to view support session recordings**; **Allowed to View Presentation Session Reports** for only his or her sessions, his or her teams' sessions, or all sessions; and **Allowed to View the License Usage Reports**. To run **ArchiveListing** or **Archive**, the account must also be an administrator.

IMPORTANT! To run **ArchiveListing** or **Archive** reports, ensure that the **Enable State Archive API** option is checked on the **Management > API Configuration** page of the **/login** administrative interface.

Required Parameters for Reporting API

username=[string]	The username to use when retrieving the reports. This user must have permission to use the reporting API. The user must also have permission to view reports. Reports returned will depend on the user's specific reporting permissions.
password=[string]	The password associated with this username.

```
generate_report=[string]
```

The type of report to be generated. Report types can be any of the following:

SupportSession	PresentationRecording
SupportSessionListing	SupportCustExitSurvey
SupportSessionSummary	SupportRepExitSurvey
SupportSessionRecording	SupportTeam
ShowMyScreenRecording	LicenseUsage
CommandShellRecording	

Starting with Bomgar 14.1, the reporting API returns XML responses that declare a namespace. If you are parsing these responses with a namespace-aware parser, you will need to set the namespace appropriately or ignore the namespace while parsing the XML.

- Reporting API: <http://www.bomgar.com/namespaces/API/reporting>

Note: The above [namespace](#) is returned XML data and is not a functional URL.

IMPORTANT! If you have been using the API prior to 14.1, this change could break backward compatibility with existing integrations if namespaces are not handled properly. **If possible, be sure to test your code against a site running Bomgar 14.1.1 or higher before upgrading.**

Download Reports with SupportSession

The **SupportSession** query returns full information for all sessions which match given search parameters. You may use any of the following sets of parameters to generate reports:

- **start_date** and **duration**
- **start_time** and **duration**
- **end_date** and **duration**
- **end_time** and **duration**
- **Isid**
- **Isids**

Parameters for SupportSession

start_date=[YYYY-MM-DD]	Specifies that the report should return all sessions, even those still in progress, that began on or after this date and that are within the duration specified below.
start_time=[timestamp]	Specifies that the report should return all sessions, even those still in progress, that began at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
end_date=[YYYY-MM-DD]	Specifies that the report should return only closed sessions that ended on or after this date and that are within the duration specified below.
end_time=[timestamp]	Specifies that the report should return only closed sessions that ended at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
duration=[integer]	Length of time from the specified date or time for which you wish to pull reports, or 0 to pull from the specified date to present. If start_date or end_date is specified, duration will represent days; if start_time or end_time is specified, duration will represent seconds.
Isid=[string]	The ID of the session for which you wish to see details.
Isids=[comma-separated strings]	A comma-delimited list of the IDs of sessions for which you wish to see details.

Optional Parameter for SupportSession

limit=[string]	The category by which to filter results. Can be one of the following:	
	all	Returns all results.
	rep:[id]	Returns sessions owned by a representative, specified by user ID. To get a representative's ID, see "API Command: get_logged_in_reps" on page 11.
	team:[all]	Returns sessions owned by any team or Embassy.
	team:[id]	Returns sessions owned by a team or Embassy specified by team ID. To get a team's ID, see "API Command: get_support_teams" on page 13.
	members:[id]	Returns sessions owned by members of a team or Embassy specified by team ID. To get a team's ID, see "API Command: get_support_teams" on page 13.
	site:[id]	Returns sessions run through a public site specified by site ID. The default public site always has an ID of 1.
<p>Note: The <i>limit</i> parameter cannot be used in conjunction with either <i>Isid</i> or <i>Isids</i>. If it is used with either of these parameters, the <i>limit</i> parameter will be ignored.</p>		

XML Response for SupportSession Query

<session_list>	Contains a <session> element for each session that matches the given criteria. If no sessions are returned, this element will contain no <session> elements. If an error occurs during the search, it will contain an <error> element describing the problem.
----------------	---

Element Names and Attributes

<i>/session_list/session</i>	
Isid (attribute)	A string which uniquely identifies this session.
<session_type>	Indicates the type of session for which the report was run. The value will always be support in the current Bomgar API version.
<lseq>	An incrementing number used to represent support sessions in a non-string format. <div style="border: 1px solid orange; padding: 5px;">Note: The LSEQ element is not guaranteed to be unique or strictly sequential.</div>
<start_time>	The date and time the session was begun either by the customer's running the customer client or by the representative's initiating a Jump session. Data is returned in ISO 8601 format. Also contains a timestamp attribute which displays the start time as a UNIX timestamp (UTC).

<end_time>	The date and time the session was ended either by the customer's closing the customer client or by the representative's closing the session. Data is returned in ISO 8601 format. Also contains a timestamp attribute which displays the end time in UNIX timestamp (UTC). This element will be empty for sessions which are still in progress when the report was run or which closed abnormally.
<duration>	Session length in HH:MM:SS format.
<public_site>	The name of the public site associated with the session. Also contains an id attribute, which displays the unique ID assigned to the public site.
<jumpoint>	The name of the Jumpoint through which this session was initiated, if any. Also contains an id attribute, which displays the unique ID assigned to the Jumpoint.
<external_key>	An arbitrary string that can link this session to an identifier on an external system, such as a help desk ticket ID. This can be input from within the representative console or defined programmatically.
<custom_attributes>	Contains a <custom_attribute> element for each custom field assigned to a session. This element displays only if custom fields have been defined. The format of each <custom_attribute> element is described below.
<session_chat_view_url>	The URL at which this session's chat transcript can be viewed in a web browser. This element is displayed only for sessions that have successfully ended.
<session_chat_download_url>	The URL at which this session's chat transcript can be downloaded. This element is displayed only for sessions that have successfully ended.
<session_recording_view_url>	The URL at which the video of the session may be viewed in a web browser. This element is displayed only if screen sharing recording was enabled at the time of the session. It is available only for sessions that have successfully ended and only if the requesting user has permission to view session recordings.
<session_recording_download_url>	The URL at which the video of the session may be downloaded. This element is displayed only if screen sharing recording was enabled at the time of the session and only if the rep initiated screen sharing during the session. It is available only for sessions that have successfully ended and only if the requesting user has permission to view session recordings.
<show_my_screen_recordings>	Contains a <show_my_screen_recording> element for each Show My Screen session that was initiated during the session. This element is displayed only if the representative shared his or her screen during the session, if Show My Screen recording was enabled at the time of the session, and only if the requesting user has permission to view Show My Screen recordings. Each <show_my_screen_recording> element contains the child elements <download_url> and <view_url> as described below.
<command_shell_recordings>	Contains a <command_shell_recording> element for each command shell that was initiated during the session. This element is displayed only if the representative opened a remote command shell during the session, if command shell recording was enabled at the time of the session, and if the requesting user has permission to view session recordings. Each <command_shell_recording> element contains the child elements <download_url> and <view_url> as described below.

<file_transfer_count>	The number of file transfers which occurred during the session.
<file_move_count>	The number of files renamed via the File Transfer interface during the session.
<file_delete_count>	The number of files deleted via the File Transfer interface during the session.
<primary_customer>	Lists the gsnumber as an attribute and as an element the name of the customer who initiated the session or, for a Jump session, the computer name of the remote system accessed by the representative.
<primary_rep>	Lists the gsnumber and id as attributes, and as an element the name of the final representative to own the session. If the session closed before it was transferred to a representative, this element will not be displayed.
<primary_team>	Lists the team ID and name of the final team to which this session was transferred. If the session was never transferred to a team, this element will not be displayed.
<customer_list>	A list of all customers who participated in the session. There should always be exactly one customer per session in the current Bomgar API version. The format of each <customer> element is described below.
<rep_list>	A list of all representatives who participated in the session, whether as session owners or conference members. The format of each <representative> element is described below. If the customer closed the session before it was transferred to a representative, this element will be empty.
<team_list>	A list of all teams to which the session belonged, whether by the session being initiated in a team queue, by a representative's explicitly transferring the session to a team, or by a session falling back into a team queue after a lost connection. This element may be empty, or it may contain one or more <team> elements as described below.
<cust_survey_list>	Contains a <cust_exit_survey> element if a customer exit survey was completed. This element is displayed only for sessions that have successfully ended and only if the customer submitted the survey. This element contains several child elements.
<rep_survey_list>	Contains a <rep_exit_survey> element if a representative exit survey was completed. This element is displayed only for sessions that have successfully ended and only if the representative submitted the survey. This element contains several child elements.
<session_details>	Contains a chronological list of all events which occurred during the session. This element contains one or more child <event> elements.

/session_list/session/custom_attributes/custom_attribute

display_name (attribute)	The display name assigned to the custom attribute.
code_name (attribute)	The code name assigned to the custom attribute.

/session_list/session/show_my_screen_recordings/show_my_screen_recording

instance (attribute)	The instance of the Show My Screen session, starting with 0 .
----------------------	--

<download_url>	The URL at which the video of the Show My Screen session may be downloaded.
<view_url>	The URL at which the video of the Show My Screen session may be viewed in a web browser.

/session_list/session/command_shell_recordings/command_shell_recording

instance (attribute)	The instance of the command shell session, starting with 0 .
<download_url>	The URL at which the video of the command shell session may be downloaded.
<view_url>	The URL at which the video of the command shell session may be viewed in a web browser.

/session_list/session/customer_list/customer

gsnumber (attribute)	Uniquely identifies the customer in regards to his or her current connection to the Bomgar Appliance. A gsnumber may be recycled, so while two people connected at the same time will never have the same gsnumber, one person may have a gsnumber that was assigned to another person in the past. Can be used to correlate a <customer> element with a <primary_cust> or with an event's <performed_by> or <destination> element.
<username>	The name used to identify the customer during the session. The method used to obtain this name varies depending on how the session was started.
<public_ip>	The customer's public IP address.
<private_ip>	The customer's private IP address.
<hostname>	The hostname of the customer's computer.
<os>	The operating system of the customer's computer.
<primary_cust>	Integer value (1 or 0) indicating if this customer was the first customer of the session. In the current version of the Bomgar API, this value is always 1 .
<info>	Contains detailed information about the customer as either entered in the front-end survey or designated programmatically. This field contains several child elements as described below.

/session_list/session/customer_list/customer/info

<name>	The name which the customer entered in the Your Name field of the front-end survey or which was assigned programmatically.
<company>	The company name which the customer entered in the Company field on the front-end survey or which was assigned programmatically.
<company_code>	The code which the customer entered in the Company Code field on the front-end survey or which was assigned programmatically.
<issue>	The numeric ID of the issue or the representative which the customer selected from the dropdown of the front-end survey or which was designated programmatically.

<details>	The description of the problem as entered by the customer in the Describe Your Issue text area field of the front-end survey or as programmatically assigned.
<i>/session_list/session/rep_list/representative</i>	
gsnumber (attribute)	<p>Uniquely identifies the representative in regards to his or her current connection to the Bomgar Appliance. A gsnumber is assigned on a per-connection basis, so if a representative leaves a session and then rejoins without logging out of the Bomgar Appliance, his or her gsnumber will remain the same.</p> <p>However, if the representative's connection is terminated for any reason, when that representative logs back into the Bomgar Appliance, he or she will be assigned a new gsnumber and will also appear multiple times in the <rep_list> element.</p> <p>A gsnumber may be recycled, so while two people connected at the same time will never have the same gsnumber, one person may have a gsnumber that was assigned to another person in the past. Can be used to correlate a <representative> element with a <primary_rep> or with an event's <performed_by> or <destination> element.</p>
id (attribute)	Unique ID assigned to the representative.
<username>	The username assigned to the representative.
<display_name>	This element is deprecated as of API version 1.10.0 but still exists for backwards compatibility. Its value is the same as that of <private_display_name>.
<public_display_name>	The public display name assigned to the representative. Note that this field contains the public display name's value at the time of the conference, which may not match the current value if the public_display_name has subsequently been changed.
<private_display_name>	The private display name assigned to the representative. Note that this field contains the private display name's value at the time of the conference, which may not match the current value if the private_display_name has subsequently been changed.
<display_number>	The display number assigned to the representative. Like <display_name>, this is the display number at the time of the session and may not match the current value.
<public_ip>	The representative's public IP address.
<private_ip>	The representative's private IP address.
<hostname>	The hostname of the representative's computer.
<os>	The operating system of the representative's computer.
<session_owner>	Integer value (1 or 0) indicating whether the representative was an actual owner of the session or was merely a conference member.
<primary_rep>	Integer value (1 or 0) indicating if the representative was the final representative to own the session.
<seconds_involved>	Integer value indicating the number of seconds the representative was involved in this session.

<embassy>	Integer value (1) present only if the representative is an Embassy user.
<invited>	Integer value (1) present only if the representative is an invited user.

/session_list/session/team_list/team

[value]	The display name of the support team. Note that this field contains the team name as it currently appears, which may not match the value at the time of the session if the team name has been subsequently changed.
id (attribute)	Integer value representing the team's unique ID.
primary_team (attribute)	Integer value (1 or 0) indicating if this team was the last team to which the session was transferred.

/session_list/session/session_details/event

timestamp (attribute)	The system time at which the event occurred.																																		
event_type (attribute)	<p>The type of event which occurred. Event types include the following:</p> <table border="1"> <tr><td>Callback Button Deployed</td><td>Files Shared</td></tr> <tr><td>Callback Button Removed</td><td>Legal Agreement Response</td></tr> <tr><td>Chat Message</td><td>Registry Exported</td></tr> <tr><td>Command Shell Session Started*</td><td>Registry Imported</td></tr> <tr><td>Conference Member Added</td><td>Registry Key Added</td></tr> <tr><td>Conference Member Departed</td><td>Registry Key Deleted</td></tr> <tr><td>Conference Member State Changed</td><td>Registry Key Renamed</td></tr> <tr><td>Conference Owner Changed</td><td>Representative Exit Survey</td></tr> <tr><td>Customer Exit Survey</td><td>Service Access Allowed</td></tr> <tr><td>Directory Created</td><td>Session Assigned</td></tr> <tr><td>External Key</td><td>Session Assignment Response</td></tr> <tr><td>File Deleted</td><td>Session End</td></tr> <tr><td>File Download</td><td>Session Note Added</td></tr> <tr><td>File Download Failed</td><td>Session Start</td></tr> <tr><td>File Moved</td><td>Show My Screen Recording</td></tr> <tr><td>File Upload</td><td>System Information Retrieved</td></tr> <tr><td>File Upload Failed</td><td></td></tr> </table> <p>*Will only appear if recording is enabled for this session.</p>	Callback Button Deployed	Files Shared	Callback Button Removed	Legal Agreement Response	Chat Message	Registry Exported	Command Shell Session Started*	Registry Imported	Conference Member Added	Registry Key Added	Conference Member Departed	Registry Key Deleted	Conference Member State Changed	Registry Key Renamed	Conference Owner Changed	Representative Exit Survey	Customer Exit Survey	Service Access Allowed	Directory Created	Session Assigned	External Key	Session Assignment Response	File Deleted	Session End	File Download	Session Note Added	File Download Failed	Session Start	File Moved	Show My Screen Recording	File Upload	System Information Retrieved	File Upload Failed	
Callback Button Deployed	Files Shared																																		
Callback Button Removed	Legal Agreement Response																																		
Chat Message	Registry Exported																																		
Command Shell Session Started*	Registry Imported																																		
Conference Member Added	Registry Key Added																																		
Conference Member Departed	Registry Key Deleted																																		
Conference Member State Changed	Registry Key Renamed																																		
Conference Owner Changed	Representative Exit Survey																																		
Customer Exit Survey	Service Access Allowed																																		
Directory Created	Session Assigned																																		
External Key	Session Assignment Response																																		
File Deleted	Session End																																		
File Download	Session Note Added																																		
File Download Failed	Session Start																																		
File Moved	Show My Screen Recording																																		
File Upload	System Information Retrieved																																		
File Upload Failed																																			
<performed_by>	The entity that performed the action. Indicates the entity's gsnumber and also its type , indicating whether this action was performed by the system , a customer , or a representative .																																		

<destination>	The entity to which the event was directed. Indicates the entity's gsnumber and also its type , indicating whether this action was directed to the system , a customer , or a representative .
<body>	The text of the message as displayed in the chat log area.
<encoded_body>	Can be shown in place of the <body> element above. Contains the base64 (RFC 2045 section 6.8) encoded value of what would have been shown in the <body> element, and is shown ONLY if the <body> text contains characters that are invalid according to XML specification. These characters are typically the result of binary data being sent through chat messages.
<filename>	The name of the transferred file.
<filesize>	An integer indicating the size of the transferred file.
<system_information>	Applies only to System Information Retrieved events wherein the system information is pulled automatically upon session start. This element contains multiple <category> child elements as described below. <i>Note: System information is logged only when pulled automatically at the beginning of the session and not when specifically requested by the representative. This is to prevent overload with the large amount of dynamic data that can be retrieved from the remote system.</i>
<files>	If this event involved the transferring of files, then this element will contain a <file> element for every file transferred.
<data>	Contains an arbitrary number of <value name="_" value="_" /> elements. The name and number of these elements varies based on event_type . For example, when a representative joins the session, a Conference Member Added event would contain <value> elements for the representative's name , username , private_ip , public_ip , embassies , hostname , os , support_teams , and user_id .

/session_list/session/session_details/event/system_information/category

<description>	Contains multiple <field> elements, each of which contains a descriptor for the specific data field. For example, the Drives category would have <field> elements Drive , Type , Percent Used , etc. These <field> elements can be compared to table header cells.
<data>	Contains multiple <row> elements, each of which contains multiple <field> elements that correspond to the <field> elements above. For example, the Drives category would have a separate <row> for each drive on the remote computer. An example <row> might contain <field> elements C:\ , Local Disk , 60% , etc. These <row> elements can be compared to table rows, with each <field> element a table cell.

Query Examples for SupportSession

Sessions started May 1 2014 to present	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_date=2014-05-01&duration=0</code>
Sessions started the month of May 2014	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_date=2014-05-01&duration=31</code>
Sessions started 8:00 AM May 1 2014 to present	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_time=1398931200&duration=0</code>
Sessions started 8:00 AM May 1 2014 to 6:00 PM May 1 2014	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_time=1398931200&duration=36000</code>
Sessions ended May 1 2014 to present	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&end_date=2014-05-01&duration=0</code>
Sessions ended the month of May 2014	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&end_date=2014-05-01&duration=31</code>
Sessions ended 8:00 AM May 1 2014 to 6:00 PM May 1 2014	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&end_time=1398931200&duration=36000</code>
Session c69a8e10bea9428f816cfababe9815fe	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&lsid=c69a8e10bea9428f816cfababe9815fe</code>
Sessions c69a8e10bea9428f816cfababe9815fe, a5eeaa58591047b88556f944804227b0, 5bf07601298b495b87310da9ce571e22	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&lsids=c69a8e10bea9428f816cfababe9815fe,a5eeaa58591047b88556f944804227b0,5bf07601298b495b87310da9ce571e22</code>
Sessions started May 1 2014 to present for all sessions	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_date=2014-05-01&duration=0&limit=all</code>
Sessions started May 1 2014 to present for a specific rep	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_date=2014-05-01&duration=0&limit=rep:1</code>
Sessions started May 1 2014 to present for all teams	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_date=2014-05-01&duration=0&limit=team:all</code>
Sessions started May 1 2014 to present for a specific team	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_date=2014-05-01&duration=0&limit=team:1</code>

Sessions started May 1 2014 to present for members of a specific team

```
https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_date=2014-05-01&duration=0&limit=members:1
```

Sessions started May 1 2014 to present for a specific public site

```
https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSession&start_date=2014-05-01&duration=0&limit=site:1
```

Download Reports with SupportSessionListing

The **SupportSessionListing** query returns a list of session IDs, external keys, and availability of a recording for sessions which match given search parameters. You may use any of the following sets of parameters to generate reports:

- **start_date** and **duration**
- **start_time** and **duration**
- **end_date** and **duration**
- **end_time** and **duration**

Parameters for SupportSessionListing

start_date=[YYYY-MM-DD]	Specifies that the report should return all sessions, even those still in progress, that began on or after this date and that are within the duration specified below.
start_time=[timestamp]	Specifies that the report should return all sessions, even those still in progress, that began at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
end_date=[YYYY-MM-DD]	Specifies that the report should return only closed sessions that ended on or after this date and that are within the duration specified below.
end_time=[timestamp]	Specifies that the report should return only closed sessions that ended at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
duration=[integer]	Length of time from the specified date or time for which you wish to pull reports, or 0 to pull from the specified date to present. If start_date or end_date is specified, duration will represent days; if start_time or end_time is specified, duration will represent seconds.

XML Response for SupportSessionListing Query

<session_summary_list>	Contains a <session_summary> element for each session that matches the given criteria. If no sessions are returned, this element will contain no <session_summary> elements. If an error occurs during the search, it will contain an <error> element describing the problem.
------------------------	---

Element Names and Attributes

/session_summary_list/session_summary

Isid (attribute)	The session ID for the given support session.
<lseq>	An incrementing number used to represent support sessions in a non-string format. <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;">Note: The LSEQ element is not guaranteed to be unique or strictly sequential.</div>
has_recording (attribute)	Integer (1 or 0) indicating if the given session has a session recording.

external_key (attribute)

An arbitrary string that can link this session to an identifier on an external system, such as a help desk ticket ID. This can be input from within the representative console or defined programmatically. This element will be displayed only if an external key has been defined.

Query Examples for SupportSessionListing

Sessions started May 1 2014 to present	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionListing&start_date=2014-05-01&duration=0</code>
Sessions started the month of May 2014	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionListing&start_date=2014-05-01&duration=31</code>
Sessions started 8:00 AM May 1 2014 to present	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionListing&start_time=1398931200&duration=0</code>
Sessions started 8:00 AM May 1 2014 to 6:00 PM May 1 2014	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionListing&start_time=1398931200&duration=36000</code>
Sessions ended May 1 2014 to present	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionListing&end_date=2014-05-01&duration=0</code>
Sessions ended the month of May 2014	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionListing&end_date=2014-05-01&duration=31</code>
Sessions ended 8:00 AM May 1 2014 to present	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionListing&end_time=1398931200&duration=0</code>
Sessions ended 8:00 AM May 1 2014 to 6:00 PM May 1 2014	<code>https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionListing&end_time=1398931200&duration=36000</code>

Download Reports with SupportSessionSummary

The **SupportSessionSummary** query returns an overview of support session statistics for representatives, teams or sites. You may use any of the following sets of parameters to generate reports:

- **start_date**, **duration**, and **report_type**
- **start_time**, **duration**, and **report_type**
- **end_date**, **duration**, and **report_type**
- **end_time**, **duration**, and **report_type**

Parameters for SupportSessionSummary

start_date=[YYYY-MM-DD]	Specifies that the report should return all sessions, even those still in progress, that began on or after this date and that are within the duration specified below.
start_time=[timestamp]	Specifies that the report should return all sessions, even those still in progress, that began at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
end_date=[YYYY-MM-DD]	Specifies that the report should return only closed sessions that ended on or after this date and that are within the duration specified below.
end_time=[timestamp]	Specifies that the report should return only closed sessions that ended at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
duration=[integer]	Length of time from the specified date or time for which you wish to pull reports, or 0 to pull from the specified date to present. If start_date or end_date is specified, duration will represent days; if start_time or end_time is specified, duration will represent seconds.
report_type=[string]	Accepted values are rep (to show representative summary statistics), team (to show team and Embassy summary statistics), or site (to show public site summary statistics).

XML Response for SupportSessionSummary Query

<summary_list>	Contains a <summary> element for each record that matches the given criteria. If no sessions are returned, this element will contain no <summary> elements. If an error occurs during the search, it will contain an <error> element describing the problem.
----------------	--

Element Names and Attributes

/summary_list/summary

id (attribute)	Returns the representative's, team's, or site's unique ID.
type (attribute)	Specifies the report type being generated: rep , team , or site .

<display_name>	The display name of the team or site, or the private display name of the representative. Note that since summary reports represent an aggregation of sessions over a period of time, the display name used is the current value for the representative, team, or site, which may have been edited since the time of the first returned session.
<total_sessions>	The total number of sessions run by the representative, team, or site in the time specified.
<avg_sessions_per_weekday>	The average number of sessions conducted on Monday through Friday by the representative, team, or site, expressed as a decimal rounded to the nearest point.
<avg_duration>	The average length of each session, expressed as HH:II:SS.

Query Examples

Sessions started May 1 2014 to present, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&start_date=2014-05-01&duration=0&report_type=rep
Sessions started May 1 2014 to present, by team	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&start_date=2014-05-01&duration=0&report_type=team
Sessions started May 1 2014 to present, by site	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&start_date=2014-05-01&duration=0&report_type=site
Sessions started the month of May 2014, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&start_date=2014-05-01&duration=31&report_type=rep
Sessions started 8:00 AM May 1 2014 to present, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&start_time=1398931200&duration=0&report_type=rep
Sessions started 8:00 AM May 1 2014 to 6:00 PM May 1 2014, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&start_time=1398931200&duration=36000&report_type=rep
Sessions ended May 1 2014 to present, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&end_date=2014-05-01&duration=0&report_type=rep
Sessions ended the month of May 2014, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&end_date=2014-05-01&duration=31&report_type=rep
Sessions ended 8:00 AM May 1 2014 to present, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&end_time=1398931200&duration=0&report_type=rep
Sessions ended 8:00 AM May 1 2014 to 6:00 PM May 1 2014, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionSummary&end_time=1398931200&duration=36000&report_type=rep

Download Reports with SupportSessionRecording

The **SupportSessionRecording** query returns the requested support session recording file. Depending on your browser, this query will either immediately begin download or prompt you to open or save the file. Note that the requesting user must have permission to view support session recordings.

Parameter for SupportSessionRecording

Isid=[string]

The session ID for which you wish to download the video recording of the support session.

Query Example for SupportSessionRecording

SupportSessionRecording: Session
c69a8e10bea9428f816cfababe9815fe

https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportSessionRecording&Isid=c69a8e10bea9428f816cfababe9815fe

Download Reports with ShowMyScreenRecording

The **ShowMyScreenRecording** query returns the requested Show My Screen recording. Depending on your browser, this query will either immediately begin download or prompt you to open or save the file. Note that the requesting user must have permission to view support session recordings.

Parameters for ShowMyScreenRecording

Isid=[string]	The session ID for which you wish to download the video recording of the Show My Screen session.
instance=[integer]	The instance number of the Show My Screen recording you wish to download. Instances are enumerated starting with 0 . The instance number can be obtained from the SupportSession report.

Query Examples for ShowMyScreenRecording

ShowMyScreenRecording: First instance of session c69a8e10bea9428f816cfababe9815fe	https://support.example.com/api/reporting?username=test&password=test&generate_report=ShowMyScreenRecording&Isid=c69a8e10bea9428f816cfababe9815fe&instance=0
ShowMyScreenRecording: Third instance of session c69a8e10bea9428f816cfababe9815fe	https://support.example.com/api/reporting?username=test&password=test&generate_report=ShowMyScreenRecording&Isid=c69a8e10bea9428f816cfababe9815fe&instance=2

Download Reports with CommandShellRecording

The **CommandShellRecording** query returns the requested command shell recording. Depending on your browser, this query will either immediately begin download or prompt you to open or save the file. Note that the requesting user must have permission to view support session recordings.

Parameters for CommandShellRecording

Isid=[string]	The session ID for which you wish to download the video recording of the command shell.
instance=[integer]	The instance number of the command shell recording you wish to download. Instances are enumerated starting with 0 . The instance number can be obtained from the SupportSession report.

Query Examples for CommandShellRecording

CommandShellRecording: First shell instance of session c69a8e10bea9428f816cfababe9815fe	https://support.example.com/api/reporting?username=test&password=test&generate_report=CommandShellRecording&Isid=c69a8e10bea9428f816cfababe9815fe&instance=0
CommandShellRecording: Third shell instance of session c69a8e10bea9428f816cfababe9815fe	https://support.example.com/api/reporting?username=test&password=test&generate_report=CommandShellRecording&Isid=c69a8e10bea9428f816cfababe9815fe&instance=2

Download Reports with PresentationRecording

The **PresentationRecording** query returns the requested presentation recording. Depending on your browser, this query will either immediately begin download or prompt you to open or save the file. Note that the requesting user must have permission to view presentation session reports.

Parameter for PresentationRecording

Isid=[string]

The session ID for which you wish to download the video recording of the presentation session.

Query Example for PresentationRecording

PresentationRecording: Session
c69a8e10bea9428f816cfababe9815fe

https://support.example.com/api/reporting?username=test&password=test&generate_report=PresentationRecording&Isid=c69a8e10bea9428f816cfababe9815fe

Download Exit Survey Reports with SupportCustExitSurvey and SupportRepExitSurvey

The **SupportCustExitSurvey** and **SupportRepExitSurvey** queries return the questions and answers to the customer or representative exit survey. You may use any of the following sets of parameters to generate reports:

- **start_date**, **duration**, **report_type**, and **id**
- **start_time**, **duration**, **report_type**, and **id**
- **end_date**, **duration**, **report_type**, and **id**
- **end_time**, **duration**, **report_type**, and **id**

Parameters for SupportCustExitSurvey and SupportRepExitSurvey

start_date=[YYYY-MM-DD]	Specifies that the report should return all sessions, even those still in progress, that began on or after this date and that are within the duration specified below.
start_time=[timestamp]	Specifies that the report should return all sessions, even those still in progress, that began at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
end_date=[YYYY-MM-DD]	Specifies that the report should return only closed sessions that ended on or after this date and that are within the duration specified below.
end_time=[timestamp]	Specifies that the report should return only closed sessions that ended at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
duration=[integer]	Length of time from the specified date or time for which you wish to pull reports, or 0 to pull from the specified date to present. If start_date or end_date is specified, duration will represent days; if start_time or end_time is specified, duration will represent seconds.
report_type=[string]	Enter rep to filter results according to the representative who last owned the session or team to filter according to team and Embassy.
id=[integer]	May be the numeric ID of the representative or team that you wish to view or "all" to display data for all representatives or teams. To get a representative's ID, see " API Command: get_logged_in_reps " on page 11. To get a team's ID, see " API Command: get_support_teams " on page 13.

Optional Parameter

site_id=[integer]	The numeric ID of the public site by which to filter results. Only exit surveys whose support sessions are associated with the given public site will be returned. If this parameter is not specified, results from only the default public site will be returned. The default public site always has an ID of 1 .
-------------------	---

XML Response for SupportCustExitSurvey and SupportRepExitSurvey Queries

<exit_survey_list>	Contains an <exit_survey> element for each session that matches the given criteria. If no sessions are returned, this element will contain no <exit_survey> elements. If an error occurs during the search, it will contain an <error> element describing the problem.
--------------------	--

Element Names and Attributes

<i>/exit_survey_list/exit_survey</i>	
Isid (attribute)	The unique ID of the session for which this survey was submitted.
ts (attribute)	The start time of the session for which this exit survey was submitted.
<session_type>	Indicates the type of session for which the report was submitted. This value will always be support in the current Bomgar API version.
<public_site>	The name of the public site associated with the session. Also contains an id attribute, which displays the unique ID assigned to the public site..
<submitted_by>	The name of the customer or private display name of the representative who submitted the survey. This element also has a type attribute with the value of cust or rep , indicating whether this survey was submitted by a customer or a representative.
<primary_customer>	The display name of the customer who initiated the session. This element also has an id attribute, the value of which is always 0 .
<primary_rep>	The private display name of the final representative to own the session, as it appeared at the time of the session. This element also has an id attribute, which is the representative's unique ID. This element will be absent if the customer closed the session before it was accepted by a representative.
<primary_team>	The display name of the last team to which the session was transferred. This element also has an id attribute, which is the team's unique ID. This element will be absent if the session was never transferred to a team.
<customer_list>	Listing of all customers who participated in this session. For full details, see the descriptions of the <customer_list> and <customer> elements in the SupportSession section.
<rep_list>	Listing of all representatives who participated in this session. For full details, see the descriptions of the <rep_list> and <representative> elements in the SupportSession section.
<team_list>	Listing of all teams to which the session was transferred. For full details, see the descriptions of the <team_list> and <team> elements in the SupportSession section.
<rep_resolved>	This element is present for backwards compatibility. In the Bomgar API versions 1.0.0 and above, this value will always be 0 .

<code><question_list></code>	Contains a <question> element for each question in this survey. This element contains several child elements as described below. Note that the <question> elements and their child <answer> elements are displayed as they are currently configured in the administrative interface. If a question was edited since the time of the first returned survey, the answers may not appear exactly as they were submitted.
------------------------------------	--

/exit_survey_list/exit_survey/question_list/question

id (attribute)	The unique ID of this question.
<code><name></code>	The name of the question as used to identify it within the web interface.
<code><type></code>	The type of question, which can be radio , checkbox , select , text or textarea .
<code><label></code>	The question text as displayed to the user taking the survey.
<code><report_header></code>	The value used to identify this question in the report.
<code><answer_list></code>	Listing of <answer> elements entered by the user. Radio , text , and textarea questions have a maximum of one <answer> . Checkbox and select questions may have more than one <answer> if multiple selection is enabled.

/exit_survey_list/exit_survey/question_list/question/answer_list

<code><answer></code>	The answer entered by the user. For radio , checkbox and select questions, this is the logged value for the selected options. For text and textarea types, it is the text typed by the user. If the question is unanswered, it will be blank.
-----------------------------	--

Query Examples for SupportCustExitSurvey and SupportRepExitSurvey

Customer surveys for sessions started May 1 2014 to present for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&start_date=2014-05-01&duration=0&report_type=rep&id=all
Customer surveys for sessions started May 1 2014 to present for all teams, by team	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&start_date=2014-05-01&duration=0&report_type=team&id=all
Customer surveys for sessions started May 1 2014 to present for a specific rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&start_date=2014-05-01&duration=0&report_type=rep&id=1
Customer surveys for sessions started May 1 2014 to present for a specific team	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&start_date=2014-05-01&duration=0&report_type=team&id=1
Customer surveys for session started the month of May 2014 for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&start_date=2014-05-01&duration=31&report_type=rep&id=all
Customer surveys for sessions started 8:00 AM May 1 2014 to present for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&start_time=1398931200&duration=0&report_type=rep&id=all

Customer surveys for session started 8:00 AM May 1 2014 to 6.00 PM May 1 2014 for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&start_time=1398931200&duration=36000&report_type=rep&id=all
Customer surveys for sessions ended May 1 2014 to present for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&end_date=2014-05-01&duration=0&report_type=rep&id=all
Customer surveys for session ended the month of May 2014 for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&end_date=2014-05-01&duration=31&report_type=rep&id=all
Customer surveys for sessions ended 8:00 AM May 1 2014 to present for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&end_time=1398931200&duration=0&report_type=rep&id=all
Customer surveys for session ended 8:00 AM May 1 2014 to 6.00 PM May 1 2014 for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&end_time=1398931200&duration=36000&report_type=rep&id=all
Customer surveys for sessions started May 1 2014 to present for all reps, by rep, for a specific site	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportCustExitSurvey&start_date=2014-05-01&duration=0&report_type=rep&id=all&site_id=1
Representative surveys for sessions started May 1 2014 to present for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&start_date=2014-05-01&duration=0&report_type=rep&id=all
Representative surveys for sessions started May 1 2014 to present for all teams, by team	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&start_date=2014-05-01&duration=0&report_type=team&id=all
Representative surveys for sessions started May 1 2014 to present for a specific rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&start_date=2014-05-01&duration=0&report_type=rep&id=1
Representative surveys for sessions started May 1 2014 to present for a specific team	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&start_date=2014-05-01&duration=0&report_type=team&id=1
Representative surveys for session started the month of May 2014 for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&start_date=2014-05-01&duration=31&report_type=rep&id=all
Representative surveys for sessions started 8:00 AM May 1 2014 to present for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&start_time=1398931200&duration=0&report_type=rep&id=all
Representative surveys for session started 8:00 AM May 1 2014 to 6.00 PM May 1 2014 for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&start_time=1398931200&duration=36000&report_type=rep&id=all
Representative surveys for sessions ended May 1 2014 to present for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&end_date=2014-05-01&duration=0&report_type=rep&id=all

Representative surveys for session ended the month of May 2014 for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&end_date=2014-05-01&duration=31&report_type=rep&id=all
Representative surveys for sessions ended 8:00 AM May 1 2014 to present for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&end_time=1398931200&duration=0&report_type=rep&id=all
Representative surveys for session ended 8:00 AM May 1 2014 to 6.00 PM May 1 2014 for all reps, by rep	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&end_time=1398931200&duration=36000&report_type=rep&id=all
Representative surveys for sessions started May 1 2014 to present for all reps, by rep, for a specific site	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportRepExitSurvey&start_date=2014-05-01&duration=0&report_type=rep&id=all&site_id=1

Download Reports with SupportTeam

The **SupportTeam** query returns information about activity within a support team. You may use any of the following sets of parameters to generate reports:

- **start_date** and **duration**
- **start_time** and **duration**
- **end_date** and **duration**
- **end_time** and **duration**

Parameters for SupportTeam

start_date=[YYYY-MM-DD]	Specifies that the report should return all sessions, even those still in progress, that began on or after this date and that are within the duration specified below.
start_time=[timestamp]	Specifies that the report should return all sessions, even those still in progress, that began at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
end_date=[YYYY-MM-DD]	Specifies that the report should return only closed sessions that ended on or after this date and that are within the duration specified below.
end_time=[timestamp]	Specifies that the report should return only closed sessions that ended at or after this time and that are within the duration specified below. The time must be a UNIX timestamp (UTC).
duration=[integer]	Length of time from the specified date or time for which you wish to pull reports, or 0 to pull from the specified date to present. If start_date or end_date is specified, duration will represent days; if start_time or end_time is specified, duration will represent seconds.

Optional Parameter for SupportTeam

team_id=[integer]	The numeric ID of the team or Embassy by which to filter results. Only the activity within the specified team or Embassy will be returned. If this parameter is not specified, results from all teams and embassies will be returned. To get a team's ID, see " API Command: get_support_teams " on page 13.
-------------------	--

XML Response for SupportTeam Query

<team_activity_list>	<p>Contains a <team_activity> element for each team with any activity within the given parameters. If no teams are returned, this element will contain no <team_activity> elements. If an error occurs during the search, it will contain an <error> element describing the problem.</p> <p>Also contains <start_time> and <end_time> elements displaying the time parameters in the system time and with a timestamp attribute in UTC.</p>
----------------------	---

Element Names and Attributes

/team_activity_list/team_activity

id (attribute)	Integer representing the team's unique ID.
name (attribute)	The display name of the support team. Note that this field contains the team name as it currently appears, which may not match the value at the time of the conference if the team name has been subsequently changed.
<logged_in_representatives>	Contains a <representative> element for each representative in that team who was logged into the representative console before the first event in the report occurred. If no representatives were logged in at the start time, this element will be empty.
<events>	Contains an <event> element for each event that occurred within this team.

/team_activity_list/team_activity/logged_in_representatives/representative

gsnumber (attribute)	<p>Uniquely identifies the representative in regards to his or her current connection to the Bomgar Appliance. A gsnumber is assigned on a per-connection basis, so if a representative leaves a session and then rejoins without logging out of the Bomgar Appliance, his or her gsnumber will remain the same.</p> <p>However, if the representative's connection is terminated for any reason, when that representative logs back into the Bomgar Appliance, he or she will be assigned a new gsnumber.</p> <p>A gsnumber may be recycled, so while two people connected at the same time will never have the same gsnumber, one person may have a gsnumber that was assigned to another person in the past. Can be used to correlate a <representative> element with an event's <performed_by> or <destination> element.</p>
id (attribute)	Unique ID assigned to the representative.
<display_name>	This element is deprecated as of API version 1.10.0 but still exists for backwards compatibility. Its value is the same as that of <private_display_name>.
<public_display_name>	The public display name assigned to the representative. Note that this field contains the public display name's value at the time of the conference, which may not match the current value if the public_display_name has subsequently been changed.
<private_display_name>	The private display name assigned to the representative. Note that this field contains the private display name's value at the time of the conference, which may not match the current value if the private_display_name has subsequently been changed.
<public_ip>	The representative's public IP address.
<private_ip>	The representative's private IP address.

/team_activity_list/team_activity/events/event

timestamp (attribute)	The system time at which the event occurred.																				
event_type (attribute)	<p>The type of event which occurred. Event types include the following:</p> <table border="1"> <tr> <td>Chat Message</td> <td>Pinned Session Moved Away from Queue</td> </tr> <tr> <td>Conference Member Added</td> <td>Pinned Session Moved to Queue</td> </tr> <tr> <td>Conference Member Departed</td> <td>Pinned Session Password Modified</td> </tr> <tr> <td>Conference Member State Changed</td> <td>Representative Monitoring Started</td> </tr> <tr> <td>Conference Owner Changed</td> <td>Representative Monitoring Stopped</td> </tr> <tr> <td>File Download</td> <td>Session Pinned to Queue</td> </tr> <tr> <td>File Download Failed</td> <td>Session Transferred Away from Queue</td> </tr> <tr> <td>File Upload</td> <td>Session Transferred to Queue</td> </tr> <tr> <td>File Upload Failed</td> <td>Session Unpinned from Queue</td> </tr> <tr> <td>Files Shared</td> <td></td> </tr> </table>	Chat Message	Pinned Session Moved Away from Queue	Conference Member Added	Pinned Session Moved to Queue	Conference Member Departed	Pinned Session Password Modified	Conference Member State Changed	Representative Monitoring Started	Conference Owner Changed	Representative Monitoring Stopped	File Download	Session Pinned to Queue	File Download Failed	Session Transferred Away from Queue	File Upload	Session Transferred to Queue	File Upload Failed	Session Unpinned from Queue	Files Shared	
Chat Message	Pinned Session Moved Away from Queue																				
Conference Member Added	Pinned Session Moved to Queue																				
Conference Member Departed	Pinned Session Password Modified																				
Conference Member State Changed	Representative Monitoring Started																				
Conference Owner Changed	Representative Monitoring Stopped																				
File Download	Session Pinned to Queue																				
File Download Failed	Session Transferred Away from Queue																				
File Upload	Session Transferred to Queue																				
File Upload Failed	Session Unpinned from Queue																				
Files Shared																					
<performed_by>	The entity that performed the action. Indicates the entity's gsnumber and also its type , indicating whether this entity was the system or a representative.																				
<destinations>	If this event was targeted to one or more specific representatives, it will contain one or more <destination> elements as described below.																				
<files>	If this event involved the transferring of files, then this element will contain a <file> element for every file transferred.																				
<data>	<p>Contains an arbitrary number of <value name="_" value="_" /> elements. The name and number of these elements varies based on the event_type. For example, when a representative logs into the representative console, a Conference Member Added event would contain <value> elements for the hostname, name, os, private_ip, public_ip, support_teams and user_id.</p> <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p>Note: The XML data returned for the value of the element named support_teams includes an extra ; delimiter at the beginning of the data stream.</p> </div>																				
<body>	The text of the chat message as displayed in the chat log area.																				
<encoded_body>	Can be shown in place of the <body> element above. Contains the base64 (RFC 2045 section 6.8) encoded value of what would have been shown in the <body> element, and is shown ONLY if the <body> text contains characters that are invalid according to XML specification. These characters are typically the result of binary data being sent through chat messages.																				

/team_activity_list/team_activity/events/event/destinations/destination

gsnumber (attribute)	Indicates the gsnumber of the entity to which the event was destined.
[value]	The name of the entity to which the event was destined.

/team_activity_list/team_activity/events/event/files/file

name (attribute)	The name of the transferred file.
size (attribute)	An integer indicating the size of the transferred file.

Query Examples for SupportTeam

Activity started May 1 2014 to present	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&start_date=2014-05-01&duration=0
Activity started the month of May 2014	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&start_date=2014-05-01&duration=31
Activity started 8:00 AM May 1 2014 to present	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&start_time=1398931200&duration=0
Activity started 8:00 AM May 1 2014 to 6:00 PM May 1 2014	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&start_time=1398931200&duration=36000
Activity started May 1 2014 to present for a specific team	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&start_date=2014-05-01&duration=0&team_id=1
Activity ended May 1 2014 to present	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&end_date=2014-05-01&duration=0
Activity ended the month of May 2014	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&end_date=2014-05-01&duration=31
Activity ended 8:00 AM May 1 2014 to present	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&end_time=1398931200&duration=0
Activity ended 8:00 AM May 1 2014 to 6:00 PM May 1 2014	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&end_time=1398931200&duration=36000
Activity ended May 1 2014 to present for a specific team	https://support.example.com/api/reporting?username=test&password=test&generate_report=SupportTeam&end_date=2014-05-01&duration=0&team_id=1

Download Reports with LicenseUsage

The **LicenseUsage** query returns an overview of peak license usage times, grouped by hour, day, or month. Data is added to this report when at least 90% of your Bomgar licenses are in use. Note that the requesting user must have permission to view license usage reports. You may use any of the following sets of parameters to generate reports:

- **start_date**, **duration**, and **group_by**
- **start_time**, **duration**, and **group_by**

Parameters for LicenseUsage

start_date=[YYYY-MM-DD]	Specifies that the report should return peak license usage data beginning on or after this date.
start_time=[timestamp]	Specifies that the report should return peak license usage data beginning at or after this time. The time must be a UNIX timestamp (UTC).
duration=[integer]	Length of time from the specified date or time for which you wish to pull reports, or 0 to pull from the specified date to present. If start_date is specified, duration represents days; if start_time is specified, duration represents seconds.
group_by	Specifies whether the data should be grouped by hour , day , or month .

XML Response for LicenseUsage Query

<license_usage>	<p>Contains a <license_time_intervals> element. If no license usage data is returned, this element will contain no license usage elements. If an error occurs during the search, it will contain an <error> element describing the problem.</p> <p>Also contains <start_time> and <end_time> elements displaying the time parameters in the system time and with a timestamp attribute in UTC. A <group_by> element shows whether the data is grouped by hour, day, or month.</p>
-----------------	---

Element Names and Attributes

	<i>//license_usage/license_time_intervals</i>
<license_time_interval>	Contains a <license_time_interval> element for each time at which peak license usage was logged.
	<i>//license_usage/license_time_intervals/license_time_interval</i>
timestamp (attribute)	The timestamp at which peak license usage was logged.
datetime (attribute)	The date and time at which peak license usage was logged.
<license_count>	Contains a <license_count> element for each potential type of license usage. This displays the number of licenses in use at that time.

/license_usage/license_time_intervals/license_time_interval/license_count

license_type (attribute)	The type of license used by the representative.
reason (attribute)	Can be either login or extended_contact , or this attribute may not be present. The login reason indicates that a license was in use due to a user being logged into the representative console. The extended_contact reason indicates that a license was being consumed due to a user being in extended availability mode. If no reason is listed, the license count is the total number of licenses in use.

Query Examples for LicenseUsage

License usage starting May 1 2014 to present, grouped by hour	https://support.example.com/api/reporting?username=test&password=test&generate_report=LicenseUsage&start_date=2014-05-01&duration=0&group_by=hour
License usage during the month of May 2014, grouped by day	https://support.example.com/api/reporting?username=test&password=test&generate_report=LicenseUsage&start_date=2014-05-01&duration=31&group_by=day
License usage starting 8:00 AM May 1 2014 to present, grouped by month	https://support.example.com/api/reporting?username=test&password=test&generate_report=LicenseUsage&start_time=1398931200&duration=0&group_by=month
License usage starting 8:00 AM May 1 2014 to 6:00 PM May 1 2014, grouped by hour	https://support.example.com/api/reporting?username=test&password=test&generate_report=LicenseUsage&start_time=1398931200&duration=36000&group_by=hour

Backup API

The Bomgar backup API is designed to enable you to automatically back up your Bomgar software configuration on a recurring basis. The backup file will include all your configuration settings and logged data except for recordings and some large files from the file store. The backup will only include files from the file store less than 200 KB in size and no more than 50 files total. In the event of a hardware failure, having a backup file will help to speed the disaster recovery process.

Commands are executed by sending a simple HTTP request to the Bomgar Appliance. The request can be sent using any HTTPS-capable socket library or scripting language module, a web browser, or a URL fetcher such as **cURL** or **wget**. Either **GET** or **POST** may be used as the request method.

Note: By default, access to the API is SSL-encrypted; however, you can choose to allow HTTP access by checking the **Allow HTTP Access to XML API** option on the **Management > API Configuration** page of the **/login** administrative interface. **It is highly recommended that HTTP remain disallowed as a security best practice.**

The backup API URL is <https://support.example.com/api/backup>.

Required Parameters for Backup API

username=[string]	The username to use when backing up the support site. Must be an administrator.
password=[string]	The password associated with this username.

Query Example

backup	https://support.example.com/api/backup?username=test&password=test
--------	---

Test Scenario

To get started with this basic API integration, follow the steps below.

1. Log into your Bomgar administrative interface and go to **Management > API Configuration**. Check the box to **Enable XML API**. If you do not have a valid SSL certificate, you may need to enable the option to **Allow HTTP Access to XML API** while you are testing.
2. Create a special Bomgar user account to be used for API commands. Give this user a password that does not need to be reset, and enable all necessary permissions such as the ability to view reports, generate session keys, and participate in the general queue. For simplicity, you can make this user an administrator with all permissions enabled, though this is not required.
3. After saving the user account, edit the account and check the box so that the password does not expire.
4. Create a normal representative user account for yourself. Download the Bomgar representative console and log in.
5. You now can begin testing API commands using your browser. Create the appropriate URLs by copying the samples into a text editor. Modify the parameters as needed for your environment, replacing the hostname, username, password, external key, and so forth.
6. Paste the customized URLs into your browser to test the API commands. The appropriate XML should be returned in the browser.
7. If you receive any errors such as **Document Not Found**, check that the API user has the necessary permissions. Also, make sure that a representative is logged into the site while you are testing.
8. Conduct a support session using the programmatically generated, seven-character session key or corresponding direct download URL. Examine the external key, which is displayed in the representative console on the **Summary** tab of the support session. The key is also visible from the session queue.
9. After the session completes, view the session report from the Bomgar administrative interface. You will notice that if assigned, the external key is displayed for each session.

API Change Log

API Version 1.11.0 for Bomgar 14.1.x

- Use three new commands to view information about your Bomgar Appliances and connected software clients.
 - [API Command: get_appliances](#)
 - [API Command: get_connected_client_list](#)
 - [API Command: get_connected_clients](#)
- Download reports of license usage data.
 - [Download Reports with LicenseUsage](#)
- Command and reporting APIs return XML that declare a namespace.
 - Reporting API: <http://www.bomgar.com/namespaces/API/reporting>
 - Command API: <http://www.bomgar.com/namespaces/API/command>

Note: The above [namespaces](#) are returned XML data and are not functional URLs.

- Specify the language to use for the customer client.
 - [Session Generation API](#)
 - [Start Sessions with Click-To-Chat or Collaborative Browser Sharing](#)

API Version 1.12.0 for Bomgar 14.2.x

- The file extension `.ns` has been deprecated from the API calls. It is still available for backward compatibility.
 - <https://support.example.com/api/command.ns> is now <https://support.example.com/api/command>
 - https://support.example.com/api/client_script.ns is now https://support.example.com/api/client_script
 - https://support.example.com/api/start_session.ns is now https://support.example.com/api/start_session
 - <https://support.example.com/api/reporting.ns> is now <https://support.example.com/api/reporting>
 - <https://support.example.com/api/backup.ns> is now <https://support.example.com/api/backup>
- Use two new API commands to help automate failover.
 - [API Command: check_health](#)
 - [API Command: set_failover_role](#)
- Set custom session attributes when starting or during a session. Customer details, skills, and custom fields can now be added to sessions regardless of session start type (session key, rep selection, issue submission). Additionally, certain fields have been deprecated in favor of a more consistent format. All fields are still available for backward compatibility. Depreciated fields are noted in their respective sections.
 - [API Command: generate_session_key](#)
 - [API Command: set_session_attributes](#)
 - [Session Generation API](#)
 - [Start Sessions with Session Key Acceptance](#)
 - [Start Sessions with Click-To-Chat or Collaborative Browser Sharing](#)
 - [Start Sessions with External Keys \(TicketID\)](#)
- Retrieve custom session attributes during a session.
 - [API Command: get_session_attributes](#)

API Version Reference

The following table shows the relationship between the API and Bomgar versions.

API Version	Bomgar Version
1.12.0	14.2.x, 14.3.1
1.11.0	14.1.x
1.10.0	13.1.x
1.9.0	12.3.x
1.8.0	12.2.x
1.7.1	12.1.4+
1.7.0	12.1.x
1.6.0	11.1.x
1.5.0	10.6.x
1.4.2	10.5.2+
1.4.1	10.5.1
1.4.0	10.5.0
1.3.2	10.4.1+
1.3.0	10.4.0
1.2.2	10.3.4+
1.2.1	10.3.2, 10.3.3
1.2.0	10.3.0, 10.3.1
1.1.1	10.2.5+
1.1.0	10.2.(0-4)
1.0.0	10.1.x

Disclaimers, Licensing Restrictions and Tech Support

Disclaimers

This document is provided for information purposes only. Bomgar Corporation may change the contents hereof without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. Bomgar Corporation specifically disclaims any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. The technologies, functionality, services, and processes described herein are subject to change without notice.

BOMGAR, BOMGAR BOX, mark B, JUMP and UNIFIED REMOTE SUPPORT are trademarks of Bomgar Corporation; other trademarks shown are the property of their respective owners.

Licensing Restrictions

One Bomgar license enables one support representative at a time to troubleshoot an unlimited number of remote computers, whether attended or unattended.* Although multiple accounts may exist on the same license, two or more licenses (one per concurrent support representative) are required to enable multiple support representatives to troubleshoot simultaneously.

*Starter Service accounts are limited to 25 Jumpoints and/or Jump Clients per site. Starter Service accounts do not provide screen or command prompt recordings.

Tech Support

At Bomgar, we are committed to offering the highest quality service by ensuring that our customers have everything they need to operate with maximum productivity. Should you need any assistance, please contact Bomgar Support at www.bomgar.com/support.

Technical support is provided with annual purchase of our maintenance plan.