

BOMGAR™

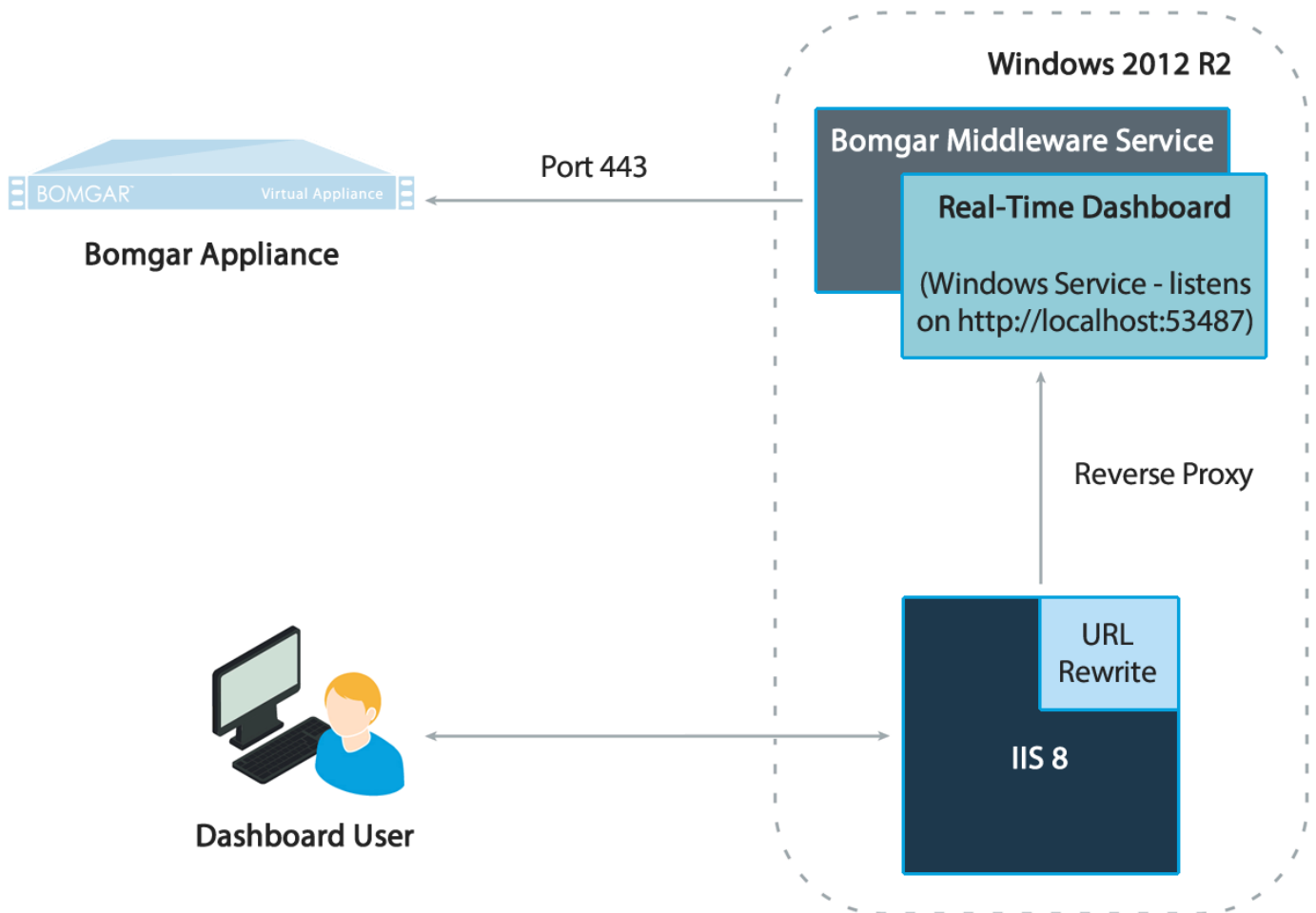
**Real-Time Dashboard Integration
Bomgar Remote Support**

Table of Contents

Real-Time Dashboard Installation and Configuration	3
Prerequisites for the Real-Time Dashboard Integration	4
Applicable Versions	4
Network Considerations	4
Prerequisite Installation and Configuration	4
Microsoft IIS Setup for Real-Time Dashboard Integration	5
Install Websocket Support into IIS	5
Download Web Platform Installer 5.0	5
Set Up Reverse Proxy	5
Create a Self-Signed Certificate	5
Create an SSL binding	6
Configure Bomgar for the Real-Time Dashboard Integration	7
Verify That the API Is Enabled	7
Create an API Service Account	7
Configure the Real-Time Dashboard Plugin for Integration with Bomgar Remote Support	8
Bomgar Appliance	8
Real-Time Dashboard Instance	9

Real-Time Dashboard Installation and Configuration

This document describes how to install and configure the Bomgar Real-Time Dashboard. The Real-Time Dashboard is a web-based dashboard which shows the current state of your Bomgar Appliance in real time. This diagram illustrates how the dashboard operates:



When the end-user accesses the Real-Time Dashboard from the web, the request is handled by Internet Information Services (IIS), which acts as a reverse proxy to the dashboard application. The dashboard application runs as a Windows Service on a local port. The dashboard communicates with the Bomgar Appliance over port 443.

Prerequisites for the Real-Time Dashboard Integration

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

Applicable Versions

- Bomgar Remote Support: 17.x and newer

Network Considerations

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

Outbound From	Inbound To	TCP Port #	Purpose
Bomgar Middleware Engine Server	Bomgar Appliance	443	Bomgar API calls.

Prerequisite Installation and Configuration

- The Real-Time Dashboard is a Bomgar Middleware Engine plugin. To install the Bomgar Middleware Engine, follow the instructions in the [Bomgar Middleware Engine Configuration](http://www.bomgar.com/docs/integrations/middleware-engine) document at www.bomgar.com/docs/integrations/middleware-engine.
- The host computer must be a fully patched Windows Server 2012 R2 or later.
- Microsoft IIS must be setup as described in "[Microsoft IIS Setup for Real-Time Dashboard Integration](#)" on page 5.
- For best results, the Bomgar Appliance and the server hosting the Bomgar Middleware Engine should have synchronized clocks. Clock drift between the servers could cause inaccurate time durations to appear on the dashboard.
- CA-signed SSL certificates are recommended for production deployments. While self-signed certificates can be used for product staging and test deployments, a self-signed certificate is NOT recommended for production use.

Microsoft IIS Setup for Real-Time Dashboard Integration

To use the Real-Time Dashboard, Microsoft Internet Information Services (IIS) must be installed. The steps below describe how to set up IIS to act as a reverse proxy to the Real-Time Dashboard, with a minimal amount of configuration.

Install Websocket Support into IIS

While websocket support is not strictly required, a websocket connection is preferred for its efficiency. If enabled in IIS and supported on the user's browser, a websocket connection is used for the dashboard.

1. From the **Server Manager** dashboard, click on **Add Roles and Features**.
2. In the wizard, click through the first few screens until you get to **Server Roles**.
3. Select **Web Server (IIS)**. If prompted to add extensions, click **OK**.
4. Leave the **Features** options at the default.
5. Under **Web Server Role (IIS) > Role Services**, expand the **Application Development** section and select **WebSocket Protocol**.
6. After completing the installation, restart IIS services.

Download Web Platform Installer 5.0

1. Download and install the **Microsoft Web Platform Installer 5.0** from www.microsoft.com/web/downloads/platform.aspx.
2. Using the Web Platform Installer, install **URL Rewrite 2.0** and **Application Request Routing 3.0** into IIS.
3. After this installation, restart IIS services or restart the server.

Set Up Reverse Proxy

These instructions assume that the Default Web Site application will be used. However, it is possible to create a new website at a specific context path, as well.

1. In the **IIS Manager**, go to the **Default Web Site**.
2. Double-click on **URL Rewrite**.
3. Click **Add Rule(s)**.
4. Choose **Reverse Proxy**. If prompted to enable proxy functionality, click **OK**.
5. Enter **localhost:53487** as the server name, leaving the other options as default.
6. Restart the Default Web Site.

Create a Self-Signed Certificate

Note: This step is only for product staging and test deployments. A CA-signed certificate is recommended for production use.

1. In **IIS Manager**, select **Web Server**.
2. Double-click on **Server Certificates**.
3. Under **Actions**, click on **Create Self-Signed**.
4. Enter a name for the certificate (e.g., **Default SSL**).
5. Select **Web Hosting** for the certificate store.
6. Click **OK**.

Create an SSL binding

Note: This step is only for product staging and test deployments. A CA-signed certificate is recommended for production use.

1. In **IIS Manager**, select the **Default Web Page**.
2. Select **SSL** settings.
3. Select **Bindings**.
4. Select type **https**.
5. Select port **443**.
6. Select the SSL certificate (from the example above, **Default SSL**).
7. Click **OK**.
8. Click **Close**.

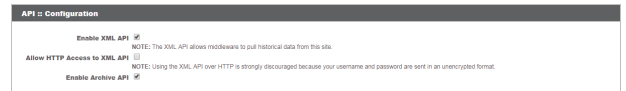
Configure Bomgar for the Real-Time Dashboard Integration

Several configuration changes are necessary on the Bomgar Appliance to implement the Real-Time Dashboard. All of the steps in this section take place in the Bomgar **/login** administrative interface. Access your Bomgar interface by going to the hostname of your Bomgar Appliance followed by **/login** (e.g., <https://support.example.com/login>).

Verify That the API Is Enabled

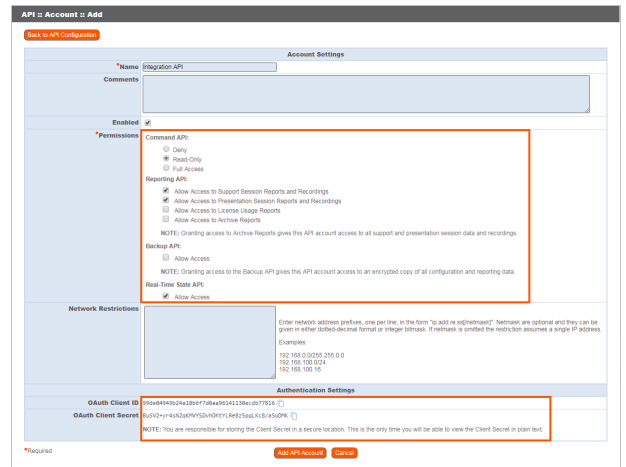
This integration requires the Bomgar XML API to be enabled. This feature is used by the Bomgar Middleware Engine to communicate with the Bomgar APIs.

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



Create an API Service Account

1. Go to **Management > API Configuration** and create a new API account.
2. Under **Permissions**, check **Read Only** to the **Command API**.
3. For the **Reporting API**, check **Allow Access to Support Session Reports and Recordings** and **Allow Access to Presentation Session Reports and Recordings**.
4. For the **Real-Time State API**, check **Allow Access**.
5. Be sure to copy the values for both the **OAuth Client ID** and **OAuth Client Secret** for use in a later step.
6. Click **Add API Account** to create the account.



Configure the Real-Time Dashboard Plugin for Integration with Bomgar Remote Support

Now that you have configured Microsoft IIS and the Bomgar Appliance, the next steps are to deploy and configure the Real-Time Dashboard plugin. For additional information about working with the Bomgar Middleware Engine, please see the [Bomgar Middleware Engine Configuration](#) document at www.bomgar.com/docs/integrations/middleware-engine.

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

Bomgar Appliance

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

1. **Plugin Configuration Name:** Any desired value. Because multiple configurations can be created for a single plugin, allowing different environments to be targeted, provide a descriptive name to indicate how this plugin is to be used.
2. **Appliance Id:** This can be left as **Default** or can be given a custom name.
3. **Bomgar Appliance Host Name:** The hostname of the Bomgar Appliance. Do not include `https://` or other URL elements.
4. **Bomgar Integration API OAuth Client ID:** The Client ID of the OAuth account.
5. **Bomgar Integration API OAuth Client Secret:** The Client Secret of the OAuth account.
6. **Locale Used for Bomgar API Calls:** This value directs the Bomgar Appliance to return session data in the specified language.
7. **Disabled:** Enable or disable this plugin configuration.
8. **Allow Invalid Certificates:** Leave unchecked unless there is a specific need to allow. If enabled, invalid SSL certificates are allowed in calls performed by the plugin. This would allow, for example, self-signed certificates. This is not recommended in production environments.
9. **Use Non-TLS Connections:** Leave unchecked. TLS cannot be disabled for this type of plugin.
10. **Real Time Events Types:** Specify which events the plugin processes when received by the middleware engine. The following should be checked:
 - a. **Real Time Data Unavailable**
 - b. **Real Time Data Available**
 - c. **Real Time State Changed**

The screenshot shows the configuration form for the Bomgar plugin. The fields are as follows:

- Plugin Configuration Name:** QA Environment
- Appliance Id:** default
- Bomgar Appliance Host Name:** support-example.com
- Bomgar Integration API OAuth Client ID:** 6673a20c36398cc77a09404f9d091c0b910
- Bomgar Integration API OAuth Client Secret:** [Redacted]
- Bomgar Integration API User Name:** [Redacted]
- Bomgar Integration API Password:** [Redacted]
- Locale Used for Bomgar API Calls:** English
- Disabled:** Disabled
- Allow Invalid Certificates:** Allow Invalid Certificates
- Use Non-TLS Connections:** Use Non-TLS Connections
- Outbound Event Types:**
 - Support Session End
 - Customer Exit Survey is Completed
 - Representative Survey is Completed
 - Someone Joins a Support Session
- Polling Event Types:**
 - Support Session End
- Polling Interval (in minutes):** [Empty field]

Real-Time Dashboard Instance

The remainder of the plugin configuration provides the necessary settings for communication between the plugin and the Real-Time Dashboard instance. The configuration settings include:

1. **Company API Name:** The value found on the **/login > Status > Information** page of the Bomgar Appliance.
2. **CSAT Polling Interval:** (Optional) The interval in minutes for the plugin to pull customer surveys to obtain a customer satisfaction average. If this functionality is not desired, leave blank or set to **0**.
3. **CSAT Question Name:** (Optional) The **Question Name** of the survey question to use for customer satisfaction levels. This is found by going to **/login > Public Portals > Exit Surveys** and editing the desired question.
4. **Public Sites to Poll for CSAT:** (Optional) A comma-separated list of public site IDs to poll for CSAT responses. If your Bomgar setup has only one public site, you can leave this field blank or enter **1**. Otherwise, go to **/login > Public Portals > Public Sites** and edit each site you wish to include; the site ID appears in the URL. For example, if the URL is **https://support.example.com/login/public_site/5/edit**, then the site ID is **5**.
5. **Allowed Support Teams:** (Optional) A comma-separated list of support team code names to include in the dashboard. This is considered a whitelist of teams. Code names are found by going to **/login > Configuration > Support Teams**. If left blank, all teams are allowed.
6. **Ignored Support Teams:** (Optional) A comma-separated list of support team code names to exclude from the dashboard. This is considered a blacklist of teams. Activity related to these teams is ignored by the real-time state API. Code names are found by going to **/login > Configuration > Support Teams**. If left blank, no teams are ignored.
7. **Valid Session Start Methods:** The types of sessions to show in the dashboard.
8. **Real Time API Idle Timeout:** The connection to the real-time state API is restarted if no data has been received within this number of minutes. If this functionality is not desired, leave blank or set to **0**.
9. **Waiting Session SLAs:** Used to describe service-level agreements for waiting sessions by team. If a session has exceeded the SLA, an exclamation icon appears next to that session.
10. **Real Time Dashboard Web Address:** The local address on which the dashboard runs (e.g., **http://localhost:53487/**).
11. **Format of representative names displayed in the Real Time Dashboard:** The preferred format for displaying names in the Real-Time Dashboard.
12. **Enable session details pop-up:** If enabled, the Real-Time Dashboard contains links in various panels to view details about closed sessions.

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