BigFix
UI Administration Guide

Special notice

Before using this information and the product it supports, read the information in Notices *(on page lvii)*.

Edition notice

This edition applies to BigFix version 11 and to all subsequent releases and modifications until otherwise indicated in new editions.

Chapter 1. Introduction

The BigFix UI Administration Guide explains how to install, configure, start, upgrade, and uninstall the BigFix UI. Use this guide to set up a working BigFix UI environment and keep it current.

About this guide

This guide covers the full administration lifecycle of the BigFix UI. It describes the environment that the product needs, the installation and configuration steps that you complete in the BigFix UI Configuration Manager, the tasks that you perform after deployment, and the procedures to upgrade or remove the product.

BigFix UI Configuration Manager is an interactive, step-by-step tool that sets up the BigFix UI environment. The tool guides you through each setting in the correct order, validates your entries, and lets you move between steps, reset fields, or exit at any time.

Who should read this guide

This guide is for BigFix administrators and IT operators who install, configure, and maintain the BigFix UI. Readers need administrator access to the target Windows server, working knowledge of the BigFix Platform, and familiarity with database and certificate administration.

What this guide contains

System Requirements *(on page 6)*

Lists the supported operating systems, database, browsers, BigFix Platform levels, hardware recommendations, and user privileges.

Installation *(on page 8)*

Explains how to run the BFUI.exe installer and complete each step in the BigFix UI Configuration Manager: administrative credentials, database, HTTPS certificates, application settings, mTLS certificates, external application resources, BES Explorer, and frontend deployment.

Getting into BigFix UI *(on page 38)*

Describes the configuration variables, port mappings, and the steps to start the BigFix UI after deployment.

Upgrade *(on page 53)*

Describes how to move an existing installation to a later version.

Uninstallation *(on page 55)*

Describes how to remove the BigFix UI from the system.

How to use this guide

For a first-time installation, read the chapters in order. Start with the system requirements, then complete the installation and configuration steps, and then start the BigFix UI. If you maintain an existing deployment, go directly to the chapter that covers your task.

Related information

System Requirements *(on page 6)*

Installation *(on page 8)*

Getting into BigFix UI *(on page 38)*

Chapter 2. System Requirements

This topic provides details about the environment requirements for running and configuring the BigFix UI. It outlines the supported hardware and software platforms, and the methods available to initiate the setup.

Software Requirements

To run BigFix UI Configuration Manager, ensure the following software components are installed and available in your environment:

Operating System: Windows Server Standard edition 2019, Windows Server Standard edition 2022, Windows Server Standard edition 2025, and Windows 11 on x86-64 (x64) architecture.

Database: Microsoft SQL Server 2025 is supported. The database can be installed on the same system where BigFix UI runs, or remotely on a different system. Configure the database server as follows:

- Install Microsoft SQL Server with FIPS mode enabled.
- Create a database user with the `sysadmin` and `public` roles.
- Open the port that the database uses. For Microsoft SQL Server, the default port is 1433.
- Set the server authentication to **SQL Server and Windows Authentication mode**.
- Create the database before you run the setup if you plan to connect through a DSN. BigFix UI does not create a database when you supply your own DSN.

BigFix Platform:

- BigFix Platform 11.0.7 or later is required to use the BigFix UI.
- The BES Explorer 11.0.7 or later must be installed and configured in the environment.
- Ensure the firewall on the system where the BES Explorer is installed does not block the port that the BES Explorer uses. The standard default port for the BES Explorer instance is 9383. If you changed the port when you deployed the BES Explorer, open that port instead. For more information, see the BigFix Platform documentation.

Browsers: Microsoft Edge and Google Chrome are supported for accessing the BigFix UI Configuration Manager.

Hardware Recommendations

Use a dedicated virtual machine for the BigFix UI installation. A dedicated infrastructure gives better performance than a shared system.

User Privileges

BigFix UI supports installation only with local administrator privileges. Run the setup with a user that is a member of the local `Administrators` group on the target system.

- A local administrator can install BigFix UI as a Windows service or run it as a process. Start BigFix UI as a Windows service for smooth operation.
- A non-administrator user cannot install BigFix UI. The setup does not display the **Install the application as a Windows service** checkbox, and the **BES UI** service is not installed.



Remember: Install BigFix UI with local administrator privileges and start it as a Windows service.

Related information

Installation *(on page 8)*

Chapter 3. Installation

This topic explains how to install the BigFixUI.exe file to initiate the BigFix UI Setup. It also outlines the steps to configure settings using the BigFix UI Configuration Manager tool.

Downloading the BFUI installer: Log in to My HCL Software and download the BFUI installer.

Stopping a running instance: BigFix UI runs as a single instance. Stop the **BES UI** service, and close any other running BFUI.exe process, before you run `--setup`.

The BFUI.exe installer initializes the setup process for the BigFix UI. The executable must be launched from the command line to initiate the configuration steps. Run the setup with a user that has administrator privileges. If you run the setup as a non-administrator user, it stops with an error. Once executed, the installer guides you through a sequence of setup screens, called steppers, that collect essential information required to establish secure communication between the BigFix UI and your environment.

1. Download and extract the BigFix UI build.

   Unzip the BigFixUI-<build_number>.zip file to a directory of your choice. This directory is referred to as the extracted directory.

   The extracted directory contains the `artifacts` and `cmd` folders.

2. Navigate to the folder containing the BFUI.exe file.

   Go to the `cmd\bfui` folder inside the extracted directory and verify that the `BFUI.exe` file is present.

   For example: `C:\BFUI\BuildExtractDir\cmd\bfui\BFUI.exe`.

3. Run the executable.

   Open a command prompt as an administrator (right-click Command Prompt and select Run as administrator), navigate to the folder that contains `BFUI.exe`, and then enter one of the following commands.

   - To install BigFix UI in the default directory, enter:

     BFUI.exe --setup

     The default installation directory is `C:\Program Files (x86)\BigFix Enterprise\BES UI`.

     The command prompts you to confirm the default setting. Enter `yes` to install in the default directory. Enter `no` to cancel the command and rerun it with the `--folder` parameter.

   - To install BigFix UI in a custom directory, add the `--folder` parameter followed by the full path:

     BFUI.exe --setup --folder "<path>"

     For example:

BFUI.exe --setup --folder "D:\BigFix\BES UI"

Enclose the path in quotation marks if it contains spaces.

Restriction: The custom installation directory cannot be a subfolder of the extracted directory. The setup rejects such a path as invalid. For example, BFUI.exe --setup --folder "<ExtractedDir>\cmd\bfui" fails.

The command extracts the required artifacts to the target location and opens the BigFix UI Configuration Manager in the default web browser.

Restriction: Install BigFix UI in an existing folder path that contains no special characters. A path that does not exist, or that contains special characters such as `#`, `@`, or `$`, can leave the installation incomplete.

Note: Setup records the installation path in the Windows registry under `HKEY_LOCAL_MACHINE\SOFTWARE\BigFix\BES UI` in the `InstallFolder` value.

Important:

BigFix UI must be the only running instance when you run BFUI.exe. The setup mode does not start while the **BES UI** service runs. Stop the service first, then run the command again to open the BigFix UI Configuration Manager.

If BigFix UI detects that another instance is already running, it stops on its own.

The `--upgrade` flag is the only exception. It detects a running service and asks you to confirm the stop. For details, see Upgrade *(on page 53)*.

Note:

In addition to `--setup`, the BFUI.exe command-line utility supports the following flags. Use these flags to configure mutual TLS (mTLS) certificates and connection settings for external application resources.

Flag	Help text	Description
`--configure-dex`	Configure DEX resource.	Configures mTLS and connection settings for the Digital Employee Experience (DEX) resource.

Flag	Help text	Description
`--configure-ivr`	Configure IVR resource.	Configures mTLS and connection settings for the Insights for Vulnerability Remediation (IVR) resource.
`--configure-wfb`	Configure Workflow Builder resource.	Configures mTLS and connection settings for the Workflow Builder (WFB) resource.

Run these flags independently of `--setup`. For the detailed procedure, see Configure External Application Resources *(on page 31)*.

To list all supported flags, run `BFUI.exe --help`.

4. After execution, the browser opens the Legal Notes page.



5. Click **Launch BigFix UI Configuration Manager**.
6. The BigFix UI Configuration stepper process starts, beginning with the Application instance credential page.

   After you set the application instance credential, a progress indicator shows the following seven configuration steps:

   a. Administrative Credentials
   b. Database Configurations
   c. HTTPS Certificates Configuration
   d. Application Configurations
   e. MTLS Certificate Configurations
   f. BES Explorer Configurations
   g. Frontend

   Each step provides **Previous**, **Reset**, and **Next** controls. **Next** remains disabled until you complete the mandatory fields for that step.

   If a step is already configured, it opens in a read-only summary that lists the current values. To edit the values, click the **here** link in the *"Click here to change your ... configuration"* text.

Changing the configuration later

To change a configuration after the setup completes, for example the database, the HTTPS certificate, the application settings, or the mTLS certificate, run BFUI.exe --setup again. The command reopens the BigFix UI Configuration Manager, where each completed step shows a read-only summary that you can edit.

Stop the **BES UI** service before you rerun the command. The Configuration Manager does not launch while the service runs.

1. Stop the **BES UI** service from the Windows Services console, or run `BFUI.exe --stop`.
2. Run BFUI.exe --setup from the `cmd\bfui` folder.
3. Update the required settings in the BigFix UI Configuration Manager.
4. Start the **BES UI** service again. See Starting BigFix UI *(on page 52)*.

Note: To change the settings for an external application resource, use the dedicated flag instead of `--setup`. See Configure External Application Resources *(on page 31)*.

Related information

Configure Administrative Credentials *(on page 11)*

Configure Database Settings *(on page 14)*

Configure HTTPS Certificates *(on page 18)*

Configure Application Settings *(on page 22)*

Configure mTLS Certificate *(on page 28)*

Configure External Application Resources *(on page 31)*

Configure BES Explorer *(on page 32)*

Deploy Frontend *(on page 35)*

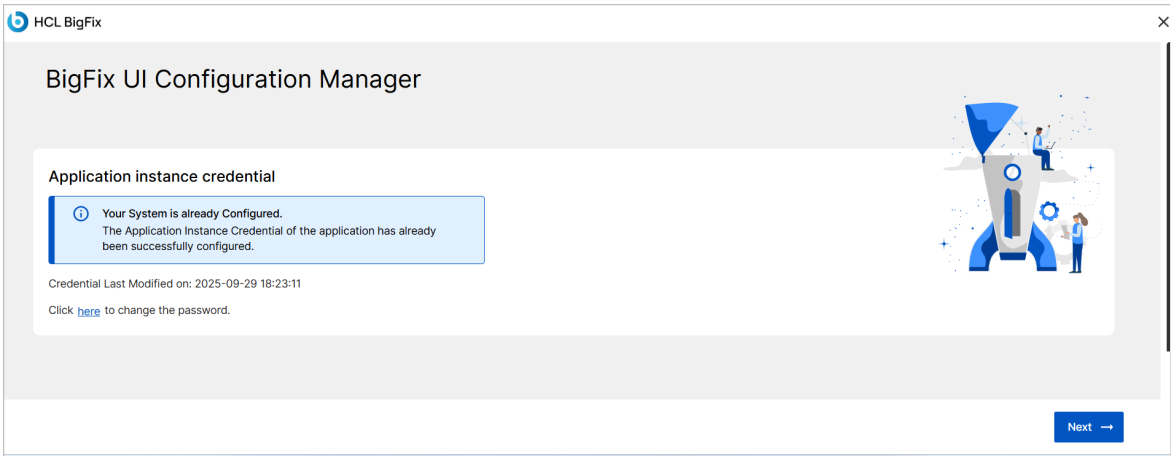
Configure Administrative Credentials

Create the application instance credential that you use to set up and configure the application. This credential is a local application identity that secures the configuration mode. Use it only for initial access to the BigFix UI and for configuration.

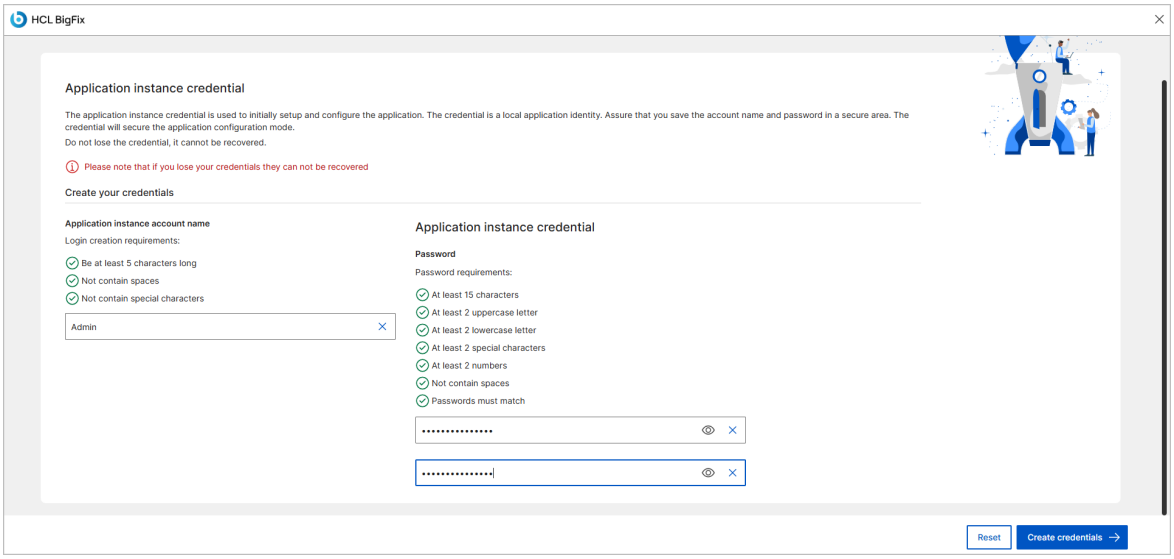
This task allows you to initialize or update administrative credentials.

1. Click **Launch BigFix UI Configuration Manager** button from the Legal Notes page.

   If the credentials already exist, the **Application instance credential** screen shows a summary that confirms the configuration and lists the date on which the credential was last modified. To change the password, click the **here** link in the *"Click here to change the password"* text.



   If no credentials exist, the **Application instance credential** screen appears with required fields to create a new account.



2. Enter the application instance account name.

   In the **Application instance account name** field, type a user name that meets the following criteria:

   - At least 5 characters long.
   - Does not contain spaces or special characters.

Important: You can set the application instance account name only during the initial setup. You cannot change it later. You can change only the password. Choose the account name carefully the first time.

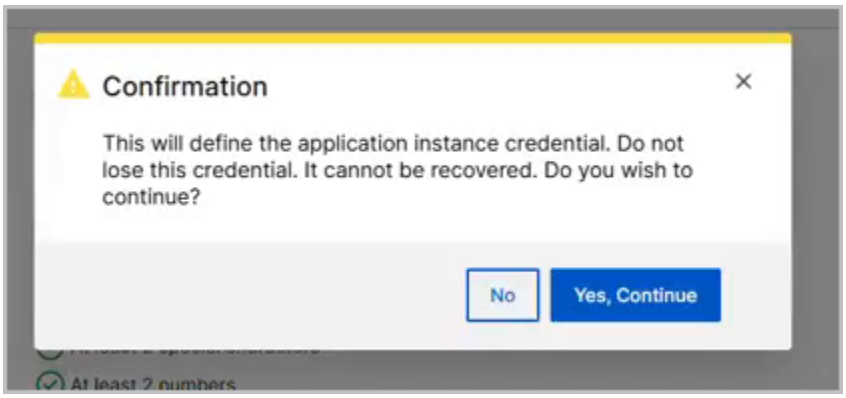
3. Enter and confirm the password.

   In the **Password** and **Confirm password** fields, enter a password that meets all of the following criteria:
   - At least 15 characters or more.
   - At least 1 lowercase letter.
   - At least 1 uppercase letter.
   - At least 1 number.
   - At least 1 special character.
   - No spaces.
   - Password must match in both fields.

   The screen marks each requirement with a green check mark when the password you type meets it.

4. Click the **Create Credentials** button.

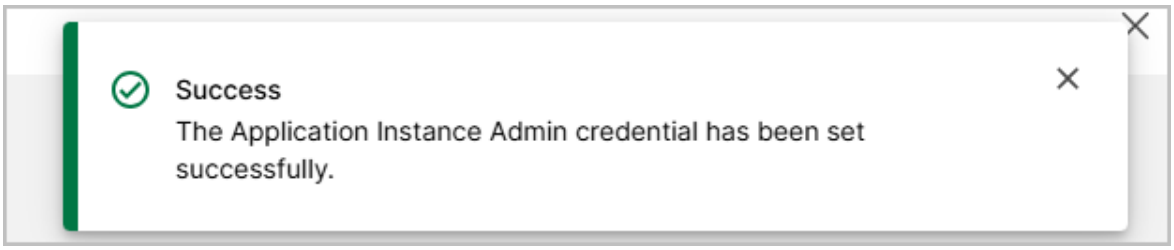
   A confirmation dialog appears.



5. Confirm the credentials.

   In the confirmation dialog, click **Yes, Continue** to define the application instance credentials.

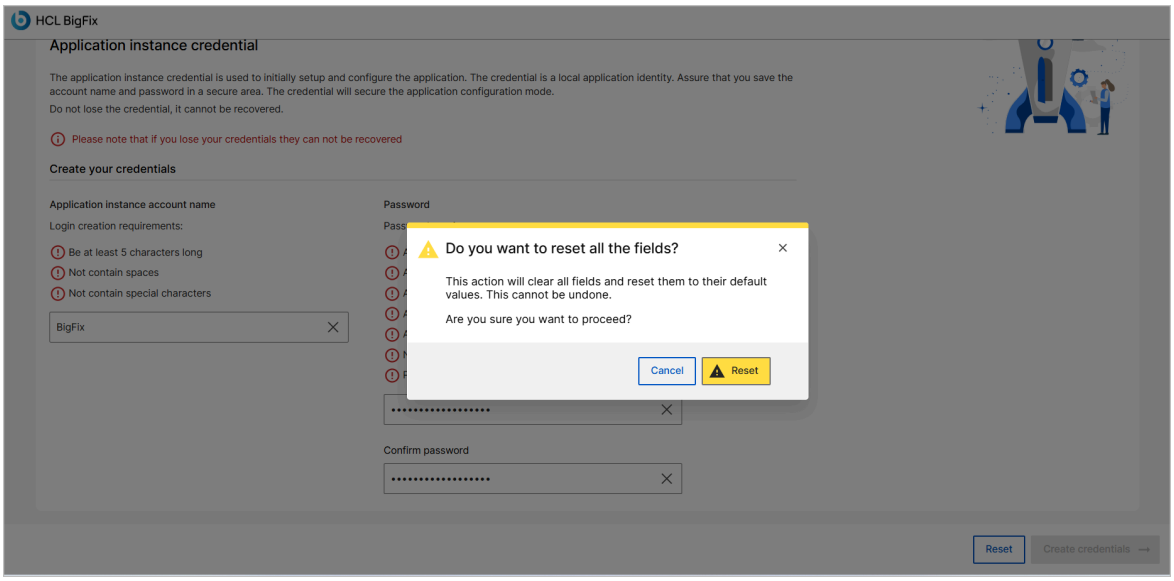
   If successful, a success message is displayed, and you are directed to the next configuration step.



Note: Store the application instance credentials securely. These credentials cannot be recovered once lost.

If there is an error, an error message is displayed and you remain on the same screen.

6. If there is an error, reset the entered values.



To reset the entered details on the **Application Instance credential** screen, click **Reset** button on the confirmation screen.

Click **Cancel** button to return without making changes.

7. After the credentials are successfully configured, click the **Next** button to proceed to the Database Configurations.

8. If you want to exit the setup process, click the X icon in the top-right corner of the screen.

Related information

Configure Database Settings *(on page 14)*

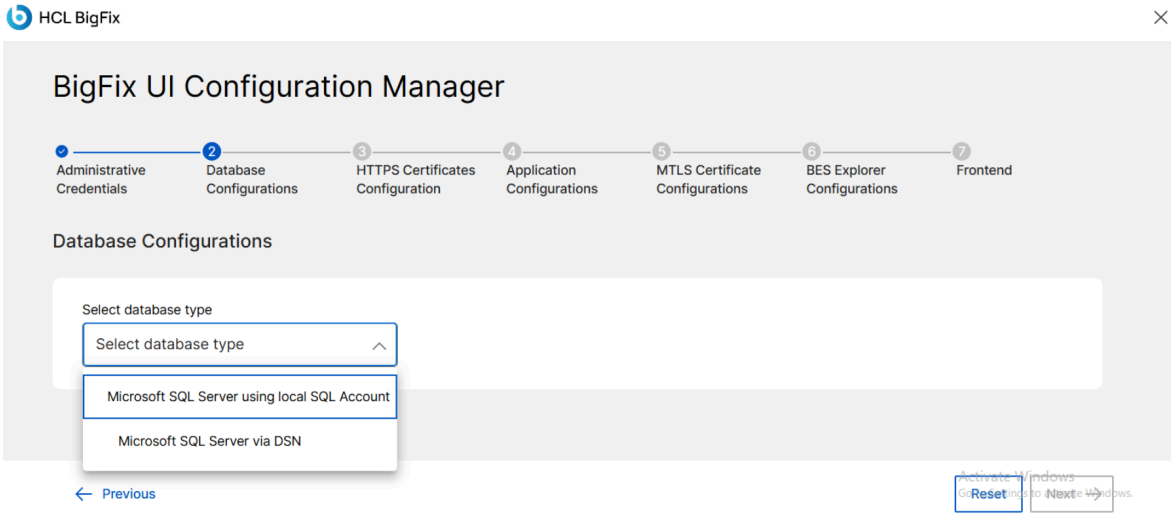
Configure Database Settings

This task explains how to select a supported database type and provide the required connection details to configure the database in the BigFix UI Configuration Manager.

Use this task to configure the database settings required for the BigFix UI Configuration Manager. You can select a supported database type and provide connection details to proceed with the application setup.

1. Ensure you are on the **Database Configurations** screen, which is step 2 of 7 in the BigFix UI Configuration Manager.

   The screen prompts you to indicate an existing database or create a new one for the BigFix UI application. The user that you specify in this step must have authority to create a database if you want to create a new one with the **Create Database (if it does not already exist)** checkbox.



2. If the database is already configured, review the **Current Database Configurations** summary.

   The summary lists the following read-only values:

   - Database host
   - Database name
   - Database username
   - Database port
   - Database type
   - Database auth type
   - Database SSL mode
   - DSN

   To change these values, click the **here** link in the *"Click here to change your database configuration"* text. To keep the current configuration, click **Next**.

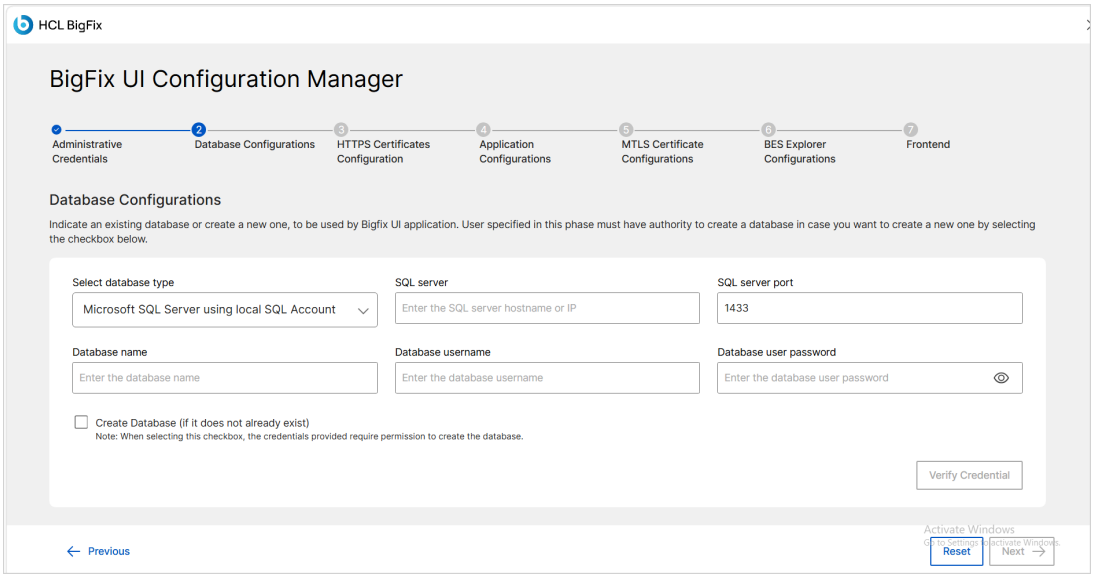
3. In the **Select database type** dropdown, click to view the available options, and then select the database type:
   - **Microsoft SQL Server using local SQL Account**
   - **Microsoft SQL Server via DSN**

   Note: Only Microsoft SQL Server is supported for database configuration. Select **Microsoft SQL Server using local SQL Account** to connect with a local SQL account. Select **Microsoft SQL Server via DSN** to connect with your own DSN string, which also supports Windows authentication.

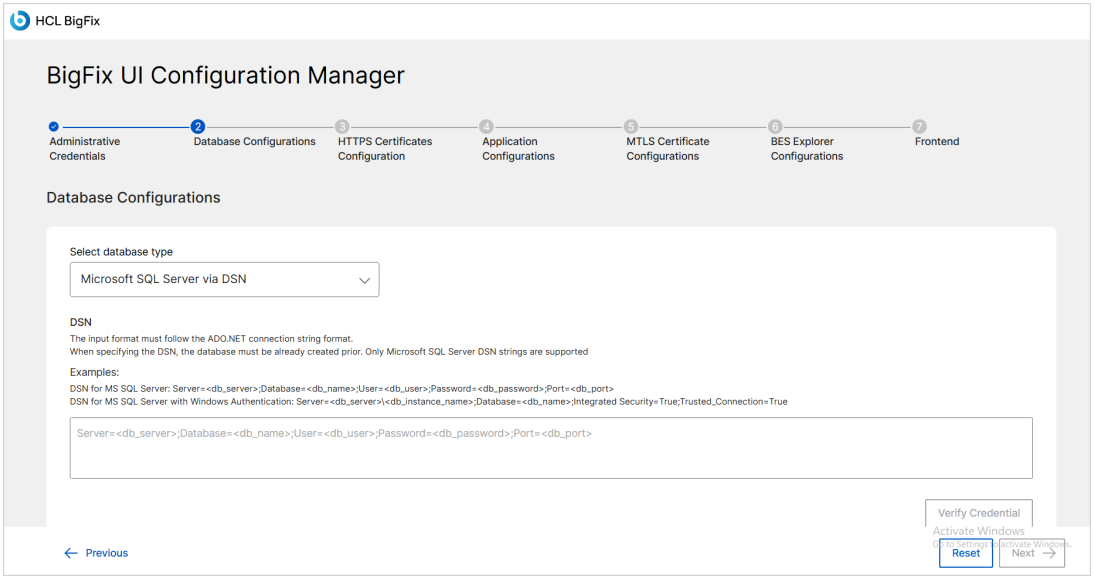
4. Review the database configuration fields that appear.

   After selecting a database type, relevant configuration fields appear. For example:

   - **Microsoft SQL Server using local SQL Account** displays fields such as:
     - SQL server
     - SQL server port (the default value is 1433)
     - Database name
     - Database username
     - Database user password
     - Create Database (if it does not already exist) checkbox.



   - **Microsoft SQL Server via DSN** shows the **DSN** input field.



Important: When you select **Microsoft SQL Server via DSN**, create the database before you run the setup. The DSN string must follow the ADO.NET connection string format. Only Microsoft SQL Server DSN strings are supported.

The screen shows the supported DSN formats. Use one of the following:

- SQL Server authentication:

```
Server=<db_server>;Database=<db_name>;User=<db_user>;Password=<db_password>;Port=<db_port>
```

- Windows (Microsoft Integrated) authentication:

```
Server=<db_server>\<db_instance_name>;Database=<db_name>;Integrated
 Security=True;Trusted_Connection=True
```

5. If the selected database does not exist, Select the **Create Database (if it does not already exist)** checkbox .

Note: When selecting this checkbox, the credentials provided require permission to create the database.

6. Enter the database configuration details.

   Fill in the fields that appear based on your database type selection.

Note: All required fields must be completed before the **Verify Credential** and **Next** buttons are enabled.

7. Once all mandatory fields are completed, click **Verify Credential** to validate the database connection.

   If verification is successful, a success message such as "Credentials verified successfully." is displayed.

   If verification fails, an error message is shown (e.g., "Invalid credentials or connection timeout. Please check your inputs.")

   Problematic fields are highlighted if possible.

8. If there is an error, click **Reset** button to clear all the entered information.

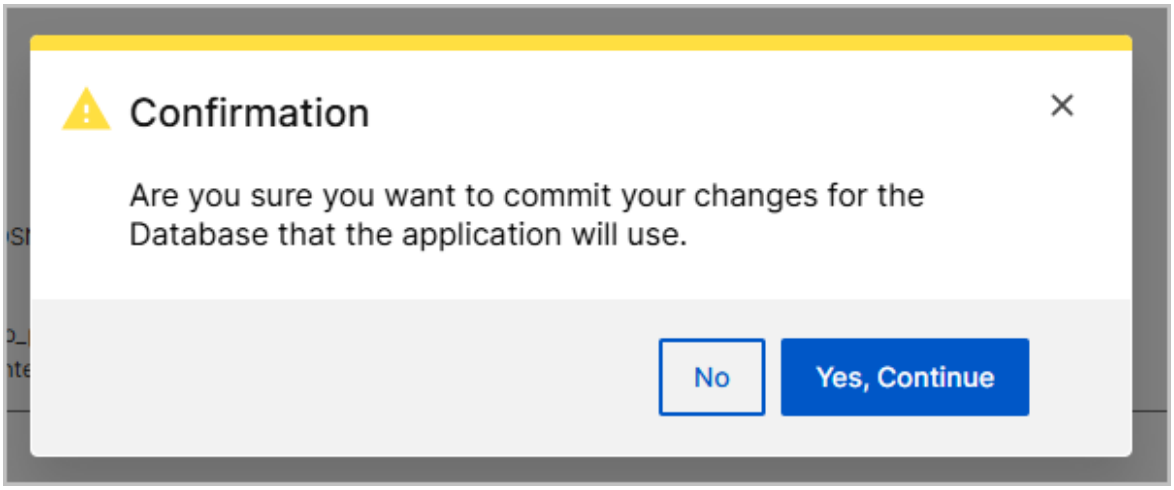
   When you click the **Reset** button, a confirmation dialog appears with the options:
   - **Reset**: Clears all data entered in the form.
   - **Cancel**: Closes the dialog and keeps the entered data.

   If you want to exit the setup process, click the **X** icon in the top-right corner of the screen.

9. Confirm the database configuration.

A **Confirmation** dialog appears with the message *"Are you sure you want to commit your changes for the Database that the application will use."*

- Click **Yes, Continue** to commit the database configuration.
- Click **No** to close the dialog and return to the form without committing the changes.



10. After the successful database configurations, click the **Next** button to proceed to the **HTTPS Certificates Configuration** screen, which is step 3 of 7 in the BigFix UI Configuration Manager.

Related information

Configure HTTPS Certificates *(on page 18)*

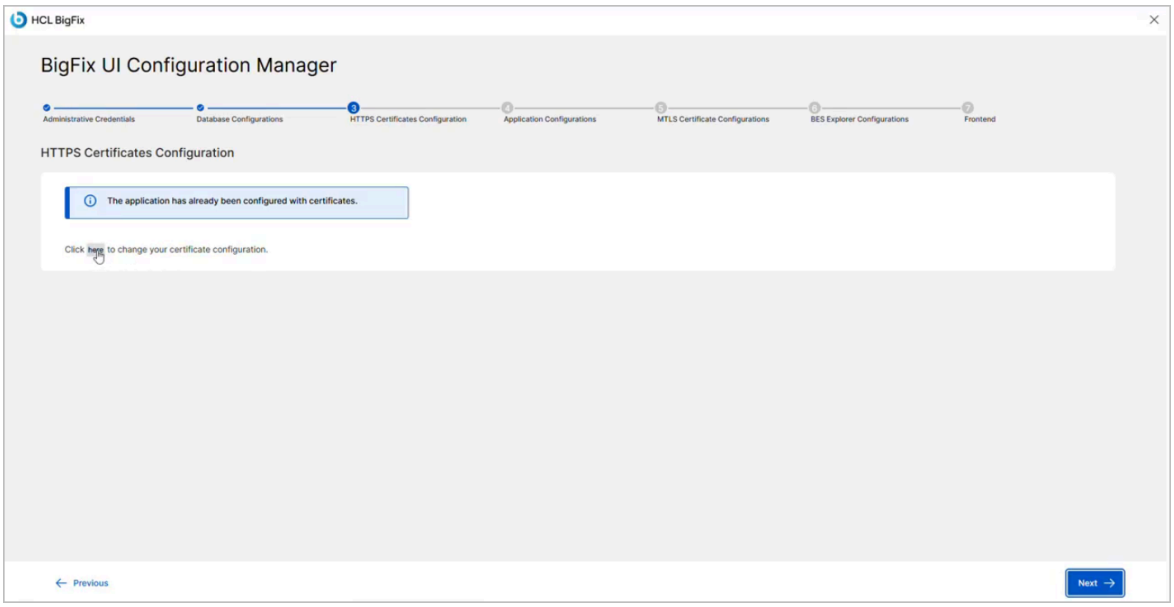
Configure HTTPS Certificates

This task explains how to set up or update the HTTPS certificate and key required for secure communication in the BigFix UI Configuration Manager.

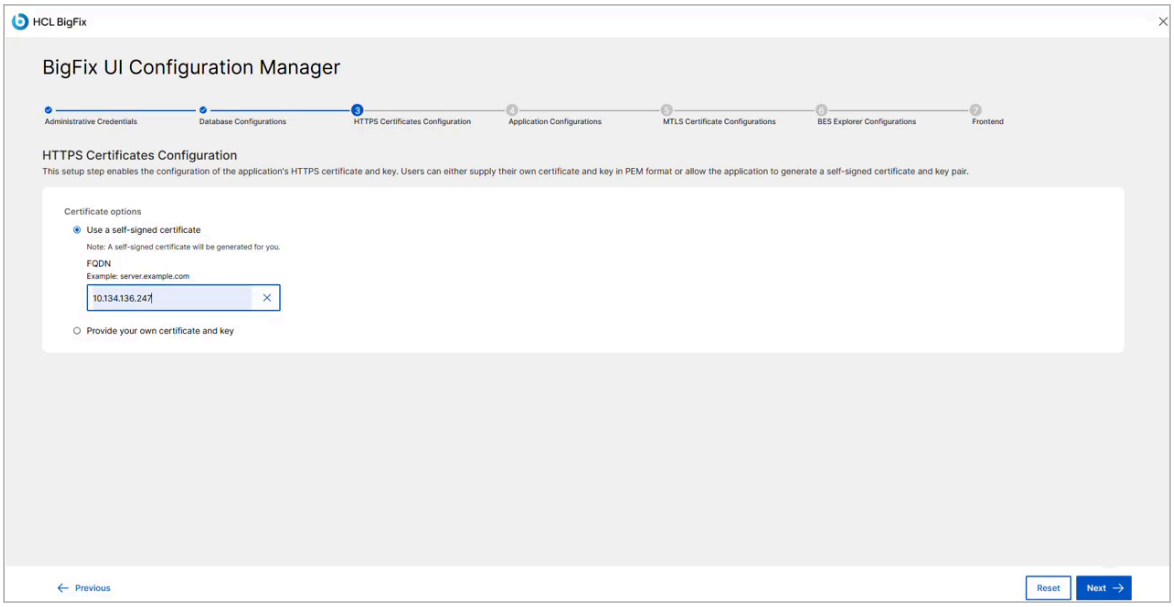
Use this task to configure certificate settings for the BigFix UI Configuration Manager. You can use a self-signed certificate generated by the system or provide your own certificate and key in PEM format.

1. After completing database configuration, click **Next** to open the **HTTPS Certificates Configuration** screen.
2. If the system detects existing certificate settings, a message is displayed indicating that the application is already configured with certificates.

   Click the provided link to change your certificate configuration, if needed.



3. If there is no existing certificate, a Certificate Configuration setup screen displayed.



Under **Certificate options**, select one of the following:

- **Use a self-signed certificate**: The system generates the certificate automatically. Enter the **FQDN** for the certificate, for example server.example.com. You can also enter an IP address.
- **Provide your own certificate and key**: You must manually enter the required details.

Enter a fully qualified domain name that resolves to the host on which you install BigFix UI, for example `mylocal.domain.com`. The certificate is issued for this name, so browsers must be able to resolve it to the BigFix UI host.

Tip:

If the FQDN is not registered in DNS, for example on a test or development system, map the name locally. Edit the following file as an administrator:

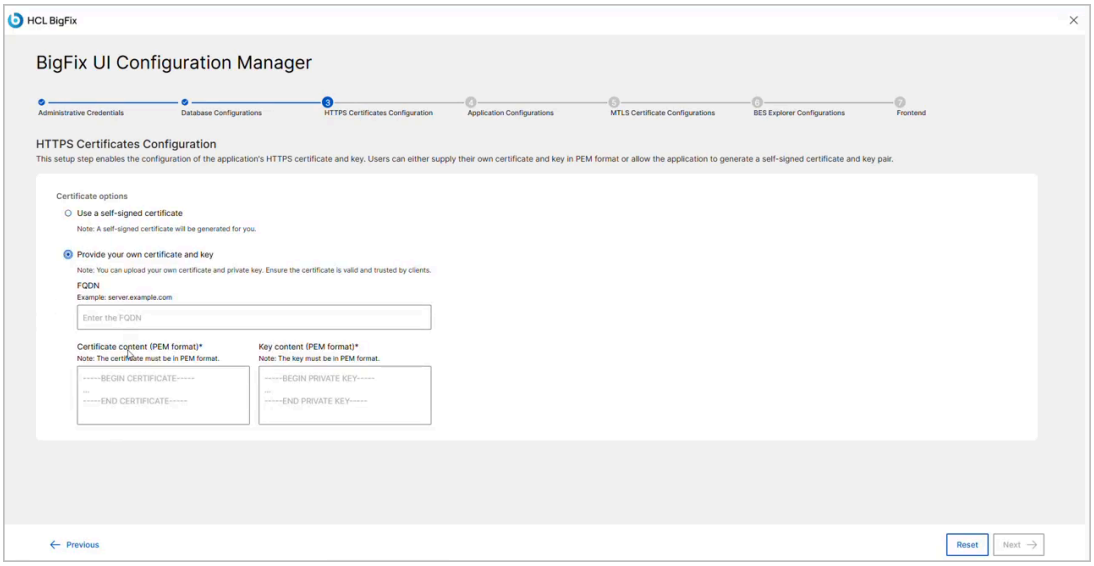
`C:\Windows\System32\drivers\etc\hosts`

Add an entry that maps the FQDN to the host. For a local development system, use the loopback address:

```
127.0.0.1    mylocal.domain.com
```

Save the file, then enter the same FQDN in the **FQDN** field. Use the loopback address only for development. For a production or shared system, register the FQDN in DNS or map it to the reachable IP address of the host.

4. When **Provide your own certificate and key** is selected, enter:
   - **FQDN** – Fully qualified domain name for the certificate
   - **Certificate content (PEM format)** – Paste or upload the PEM certificate text.
   - **Key content (PEM format)** – Paste or upload the PEM key text.

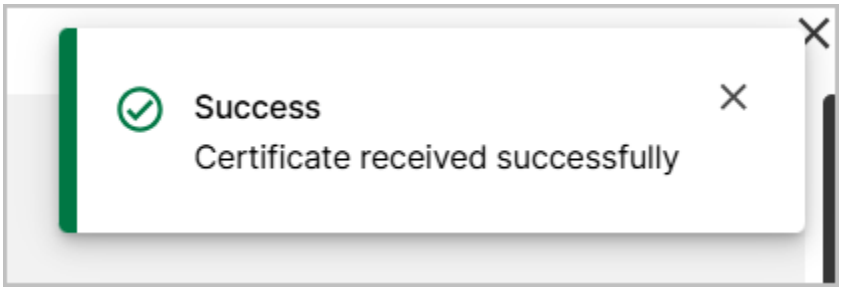


Note:

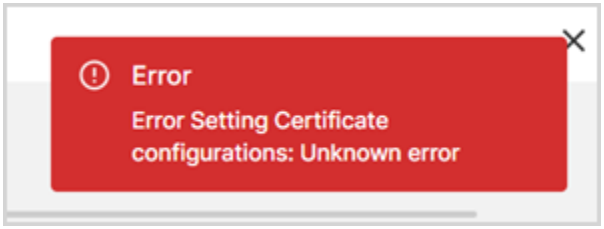
- All mandatory fields are marked with an asterisk (*).
- Certificates and keys must be in PEM format.
- A self-signed certificate option is recommended only for testing or non-production environments.

5. After filling in all mandatory fields, click **Next** button.

   If the certificate is accepted, a success message such as "Certificate received successfully" appears, and you proceed to the next setup step.

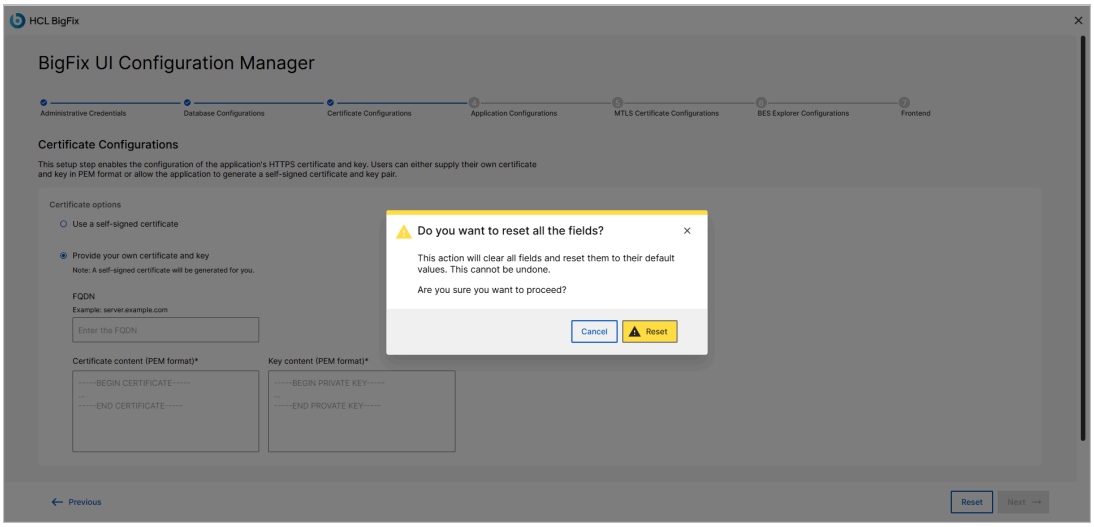


   If the certificate is invalid, an error message is displayed, and you remain on the same screen until the issue is resolved.



6. If there is an error, click **Reset** button to clear all the entered information. A confirmation dialog appears with the options:

- **Reset** - Clears all fields and reverts them to default values.
- **Cancel** - Closes the dialog without resetting fields.



7. If you want to exit the setup process, click the **X** icon in the top-right corner of the screen. An exit confirmation dialog displayed.
   - Click **Okay**, Exit to terminate the setup and close the application window.
   - Click **Cancel** to remain on the current screen.

Related information

Configure Application Settings *(on page 22)*

Configure Application Settings

Use this step to define application-level settings such as communication protocol, web server port, CORS origins, and security mode for the BigFix UI.

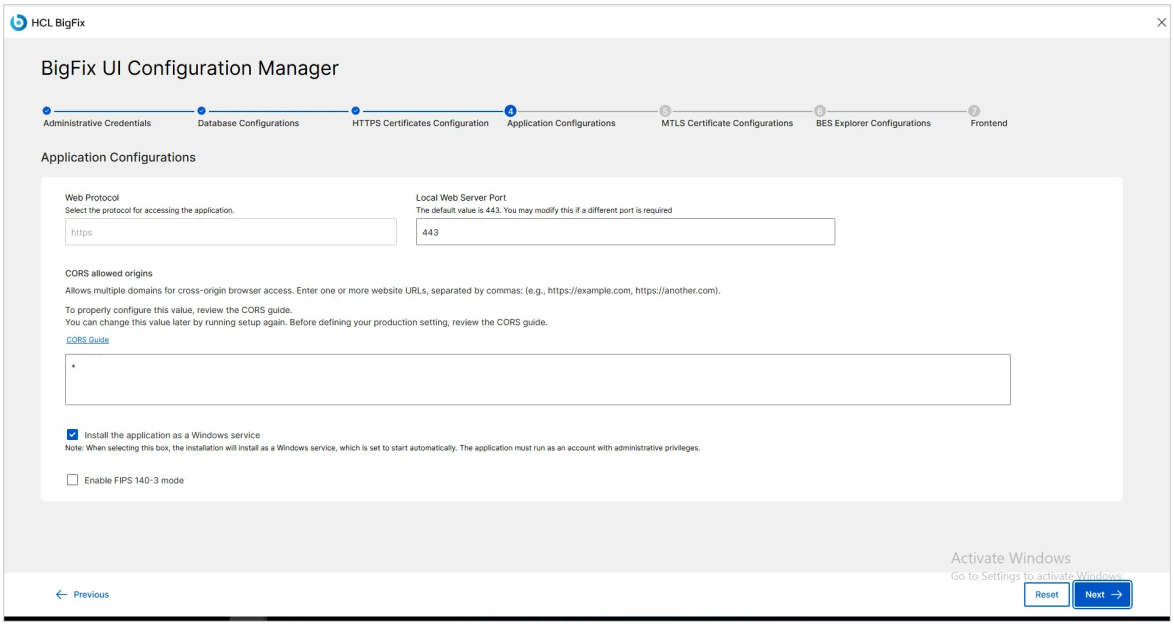
The Application Configuration page defines how BigFix UI communicates with your environment. Here you can set the communication protocol (such as HTTPS), specify the web server port, list trusted domains for browser access, enable FIPS 140-3 mode for stronger security, and optionally install the application as a Windows service. These settings ensure that the UI runs securely and consistently based on your organization's requirements.

1. After completing the HTTPS certificates configuration, click **Next** button to open the **Application Configurations** screen.
2. If the system detects existing application settings, a message is displayed indicating that the application settings are already configured.

   Click the provided link to change your application settings, if needed.



3. If no application settings exist, the **Application Configurations** setup screen is displayed.



4. Review the **Web Protocol** field.

   The field shows `https` and is read-only. BigFix UI always uses HTTPS for access.

5. In the **Local Web Server Port** field, review or enter a port number.
   - The default port is 443.
   - You can provide a custom port if required.

6. In the **CORS allowed origins** field, enter the wildcard `*` or one or more website URLs that are permitted to access this application.

   The field accepts one of the following values:

   - `*` — all origins are allowed. This is the default value.
   - A comma-separated list of origins, for example `https://192.168.0.7,https://bfui.myserver.com:8443`.

   Each origin in the list must follow these rules:

   - The origin must start with `http://` or `https://`.
   - The origin must not include a path, such as `/ui`.
   - The origin must not end with a trailing slash.

- The list must not contain extra spaces or empty entries, such as `,,`.
- If an origin uses a port other than 443, the origin must include that port.

Important: By default, this field is set to `*` (all origins) to ensure access during initial setup. Use `*` for development only. In production, set a comma-separated list of trusted origins instead, according to your organization's security policy. For detailed guidance, see CORS allowed origins configuration *(on page 25)*.

Note: The setup process uses this value to build the BigFix UI URL that it suggests when the setup finishes. For details, see How the suggested BigFix UI URL is built *(on page 27)*.

To open the reference guidance for this field, click the **CORS Guide** link. You can change this value later by running the setup again.

7. Keep the **Install the application as a Windows service** checkbox selected.

   This checkbox is selected by default. When selected, the installation registers the application as a Windows service that starts automatically. The application must run under an account with administrative privileges.

   Note: The setup requires administrative rights. If you run BFUI.exe as a non-administrator user, the setup stops with an error and does not open the Configuration Manager. Run the command prompt as an administrator. See Installation *(on page 8)*.

   Remember: Installing the application as a Windows service is recommended. Clear this checkbox only when you run the application manually from the command line, for example for development or debugging.

   Note: This checkbox will automatically disappear once the service is successfully installed. If the service is uninstalled, the checkbox will reappear. This behavior is intentional and is designed to reflect the current installation status of the service.

8. Review the **Enable FIPS 140-3 mode** checkbox.

   This checkbox is cleared by default. Select it only when your environment requires FIPS 140-3 compliant cryptography.

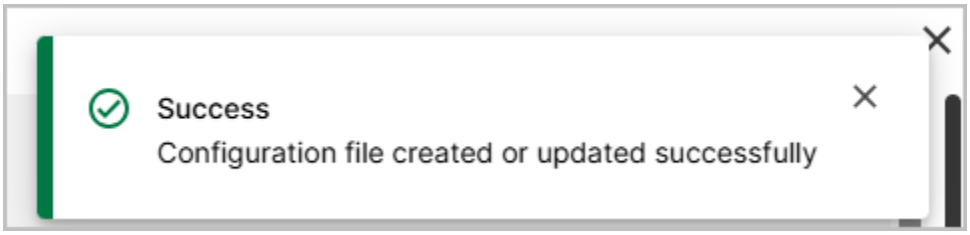
9. Review all settings and click **Next** to save the configuration and proceed to the next step in the setup process.

   If the **CORS allowed origins** field contains the wildcard value `*`, the **Confirm less restrictive CORS settings** dialog appears. The dialog warns that a wildcard lets any domain make cross-origin requests to your environment.

- Click **Yes, confirm** to proceed with the wildcard value.
- Click **Cancel** to return to the form and enter trusted domains.

If you enter specific URLs in the **CORS allowed origins** field, this dialog does not appear.

If the application configuration is accepted, a success message is displayed.



Related information

Configure mTLS Certificate *(on page 28)*

CORS allowed origins configuration *(on page 25)*

CORS allowed origins configuration

CORS (cross-origin resource sharing) controls which browser origins can interact with BigFix UI. Review this information before you set the **CORS Allowed Origins** value for a production deployment.

Every request — loading data, saving changes, authenticating — passes through the CORS control. If an origin is not allowed, the browser blocks the request silently: the page loads, but nothing works. CORS is both an operational and a security decision. If the configuration is too restrictive, teams cannot work; if it is too permissive, session details risk exposure on untrusted networks. The right choice depends on how your teams access the application, your operating environment, and your security policies.

Important: BigFix UI ships with CORS configured to allow all origins (`*`). This default ensures access during initial setup and must not remain after deployment. Set the allowed origins according to your organization's security policy before production use.

How CORS works

When a user opens the application, the browser records the origin — the scheme, host, and port from the address bar. Every request includes that origin in a header. The server responds with its list of trusted origins; if the user's origin is absent, the browser blocks the response before the page can read it.

Think of a CORS origin from the browser's perspective, not the server's. For example, `https://app.company.com`, `http://192.168.1.100`, `https://bfuiServer`, and `https://localhost` are different origins even if they point to the same server. Each access method is a distinct origin and must be explicitly allowed.

Access methods and valid configuration values

The value that you enter in the CORS allow-list must exactly match the origin that the browser sends, including scheme, host, and port. The following table shows the correct format for each access method.

Table 1. Valid configuration values per access method

Access method	Example	Valid configuration value
Domain name	`https://app.company.com`	`https://app.company.com`
Domain name (HTTP)	`http://app.company.com`	`http://app.company.com`
IP address	`https://192.168.1.100`	`https://192.168.1.100`
IP address (with port)	`https://192.168.1.100:8080`	`https://192.168.1.100:8080`
NetBIOS / hostname	`https://myserver.local`	`https://myserver.local`
Localhost	`https://localhost:3000`	`https://localhost:3000`
All origins (wildcard)	any	`*`

Keep the following rules in mind:

- **Port matters.** `http://192.168.1.100` and `http://192.168.1.100:8080` are different origins. If an origin uses a port other than 443, include that port in the origin.
- **HTTP and HTTPS are different origins.** `http://app.company.com` and `https://app.company.com` must be listed separately if both are in use.
- **No trailing slash.** `https://app.company.com/` is not a valid origin value — do not include a path.

Accepted values and syntax rules

The **CORS allowed origins** field accepts one of the following values:

- `*` — all origins are allowed. This is the default value.
- A comma-separated list of origins, for example `https://192.168.0.7,https://bfui.myserver.com:8443`.

Each origin in the list must follow these rules:

- The origin must start with `http://` or `https://`.
- The origin must not include a path, such as `/ui`.
- The origin must not end with a trailing slash.
- The list must not contain extra spaces or empty entries, such as `,,`.
- If an origin uses a port other than 443, the origin must include that port.



Important: Use `*` for development only. In production, set a comma-separated list of trusted origins instead.

How the suggested BigFix UI URL is built

When the setup process finishes, it suggests a URL for accessing BigFix UI. The setup process derives that URL from the **CORS allowed origins** value:

- If the value is `*` (the default), the suggested URL is `https://127.0.0.1/ui`. The port is added only if it is not 443, for example `https://127.0.0.1:8443/ui`.
- If the value is a custom list, the suggested URL is the first origin in the list plus `/ui`. For example, `https://192.168.0.7,https://bfui.myserver.com` gives `https://192.168.0.7/ui`.

Because the suggested URL always matches a configured origin, browsing to it does not cause a CORS violation.

Considerations before deciding

- **Actual access patterns:** How do your teams reach the application?
- **Network trust boundary:** If IP or hostname access is needed, is it confined to a trusted internal network? The security value depends on network controls. In general, avoid IP addresses in the CORS configuration.
- **DNS reliability:** Using an IP address or hostname because DNS is unreliable is a network issue, not a CORS issue.
- **Environment lifecycle:** Development and staging environments might need temporary origins; these must not persist in production.

Configuration options

Table 2. CORS configuration options

Option	Configuration	Acceptance
Domain name only (recommended starting point)	`Allow: https://www.company.com`	Users must access the application via the domain. If DNS fails or users use an IP address, the application appears broken. Ensure DNS reliability before choosing this option.
Domain + IP + hostname	`Allow: https://www.company.com, http://192.168.1.100, http://my-server.local`	Broader access for continuity.
Wildcard (`*`)	`Allow: *` (any origin)	Any website can send requests using an authenticated user's session. Generally not recommended. BigFix UI defaults to this value temporarily to prevent service interruption during setup. Administrators must configure the allow-list per security policy.

Impact of incorrect configuration

Too restrictive: The application loads, but requests fail silently. Users see blank screens, missing data, or unresponsive actions. No error messages appear.

Too permissive: A malicious website can send requests using an authenticated user's session — reading, modifying, or deleting data without the user's knowledge. This risk applies only where users authenticate.

Recommended configuration

A common best practice is to align the CORS configuration with the server certificate. Have a trusted third-party certificate authority (CA) issue a signed certificate for the application server, bound to a DNS hostname, and use that certificate for TLS communications.

In an ideal configuration, the CORS allow-list aligns exactly with the DNS hostnames that are covered by the issued certificate. This alignment matters because CORS controls which origins the browser trusts for cross-origin requests — restricting it to the certificate's hostname prevents access through entry points that the certificate does not cover.

This is also why IP addresses should generally not appear as CORS origins. Most CAs do not issue a certificate bound to an IP address, so no equivalent authentication is available for that entry point. Aligning CORS with the certificate's hostname ensures that browser access is only possible through an entry point that the presented certificate can authenticate.

This guidance rests on two assumptions that hold in most environments but should be verified rather than presumed:

- **DNS reliability** — hostname resolution within the network cannot be trivially spoofed or hijacked (for example, via a compromised internal DNS server or hosts-file tampering).
- **Client trust store integrity** — clients hold a valid certificate chain to authenticate the presented certificate.

Related information

Configure Application Settings *(on page 22)*

Configuration Variables *(on page 38)*

Configure mTLS Certificate

Use this step to configure mutual TLS (mTLS) certificates for secure internal and external communications.

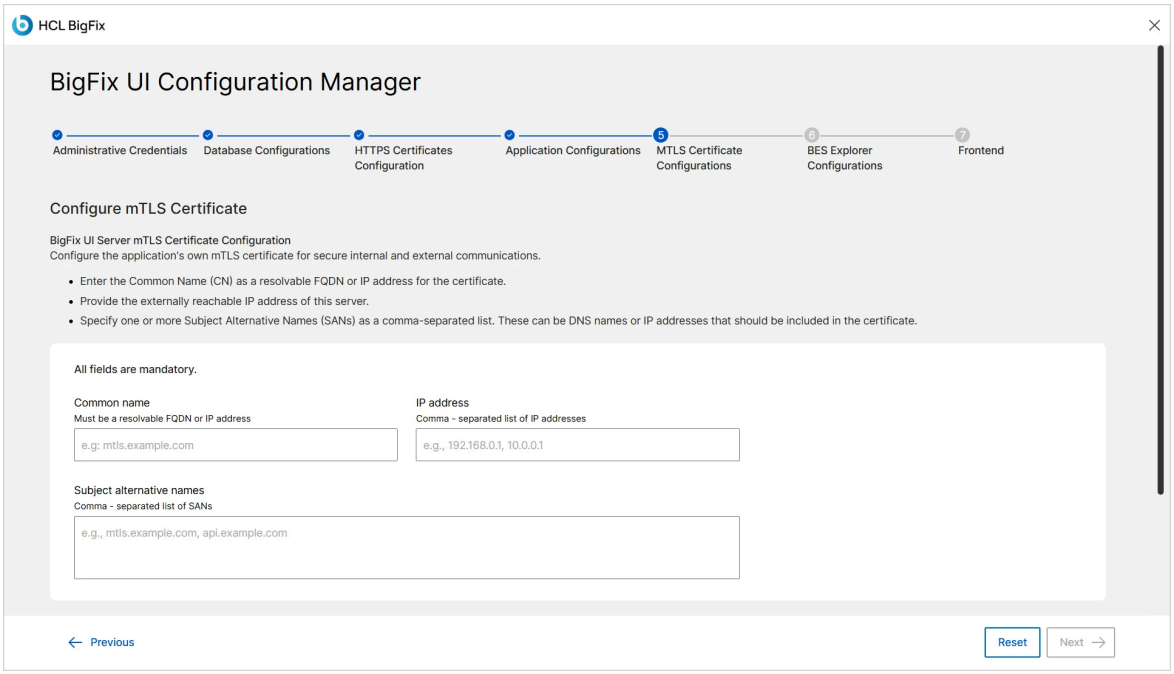
The **Configure mTLS Certificate** step ensures encrypted and trusted communication between the BigFix UI application and its endpoints. The screen shows the subtitle **BigFix UI Server mTLS Certificate Configuration** and the description *"Configure the application's own mTLS certificate for secure internal and external communications."*

The screen lists the following guidance:

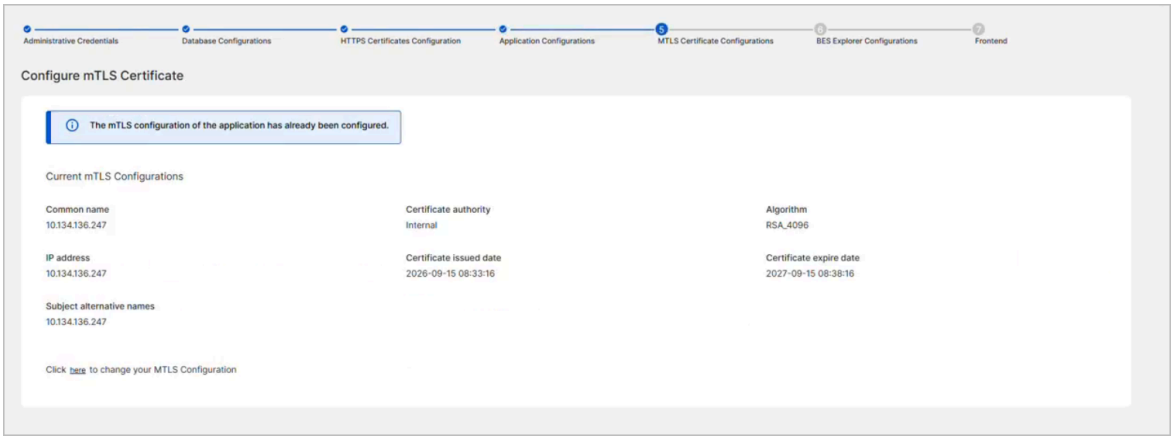
- Enter the Common Name (CN) as a resolvable FQDN or IP address for the certificate.
- Provide the externally reachable IP address of this server.
- Specify one or more Subject Alternative Names (SANs) as a comma-separated list. These can be DNS names or IP addresses that should be included in the certificate.

All fields are mandatory.

1. After completing Application Configuration, click **Next** button to open the **Configure mTLS Certificate** screen.



2. If the mTLS certificate is already configured, review the **Current mTLS Configurations** summary.



The summary lists the following read-only values:

- Common name
- Certificate authority

- Algorithm
- IP address
- Certificate issued date
- Certificate expire date
- Subject alternative names

To change these values, click the **here** link in the *"Click here to change your MTLS Configuration"* text. To keep the current configuration, click **Next**.

Note: The internal CA issues the certificate with a validity period of one year from the issued date.

3. In the **Common name** field, enter a resolvable FQDN or IP address for the certificate.

   The field helper text reads *"Must be a resolvable FQDN or IP address"*.

   Example: mtls.example.com.

4. In the **IP address** field, provide the externally reachable IP address of this server.

   The field helper text reads *"Comma - separated list of IP addresses"*. Separate multiple IP addresses with commas.

   Example: 192.168.0.1, 10.0.0.1.

5. In the **Subject alternative names** (SANs) field, specify one or more DNS names or IP addresses.

   The field helper text reads *"Comma - separated list of SANs"*. Separate multiple SANs with commas.

   Example: mtls.example.com, api.example.com.

6. Review all entries carefully. After saving, the application generates and uses this certificate for mTLS-secured endpoints.

7. Click **Next** to continue to the BES Explorer Configurations step.

   If the mTLS configuration is accepted, a success message is displayed.



Related information

Configure BES Explorer *(on page 32)*

Configure External Application Resources *(on page 31)*

Configure External Application Resources

Use this task to configure mutual TLS (mTLS) certificates and connection settings for the external application resources that integrate with BigFix UI: Digital Employee Experience (DEX), Insights for Vulnerability Remediation (IVR), and Workflow Builder (WFB).

- The initial BigFix UI setup (`BFUI.exe --setup`) completed successfully.
- You have administrative privileges on the host system.
- The **BES UI** service is stopped. BigFix UI must be the only running instance when you run BFUI.exe.
- The required TLS/mTLS client certificates, private keys, and trusted CA certificates are available for each external resource.

The BFUI.exe setup command-line utility provides a dedicated flag for each external application resource. Run these flags independently of `--setup`, at any time after the initial setup completes. To list all supported flags, run `BFUI.exe --help`.

Important: Every BFUI.exe launch mode requires BigFix UI to be the only running instance. Stop the **BES UI** service before you run these commands. If BigFix UI detects that another instance is already running, it stops on its own.

1. Navigate to the `cmd\bfui` folder that contains the BFUI.exe file.
2. Open a command prompt with elevated administrative permissions.
3. Run the command for the external application resource that you want to configure.

Resource	Command
Digital Employee Experience (DEX)	`BFUI.exe --configure-dex`
Insights for Vulnerability Remediation (IVR)	`BFUI.exe --configure-ivr`
Workflow Builder (WFB)	`BFUI.exe --configure-wfb`

4. Follow the interactive prompts to provide the required certificate paths, key files, and external service endpoint details.
5. Verify that the configuration completes successfully before you start or restart the BigFix UI service.

BigFix UI uses the configured certificates to establish mTLS-secured communication with each external application resource.

Note: Configure each external resource separately. Run these commands individually at any time after the initial setup completes.

Related information

Configure mTLS Certificate *(on page 28)*

Configure BES Explorer *(on page 32)*

Configure BES Explorer

Use this step to configure the BES Explorer instance for secure communication with the BigFix UI application.

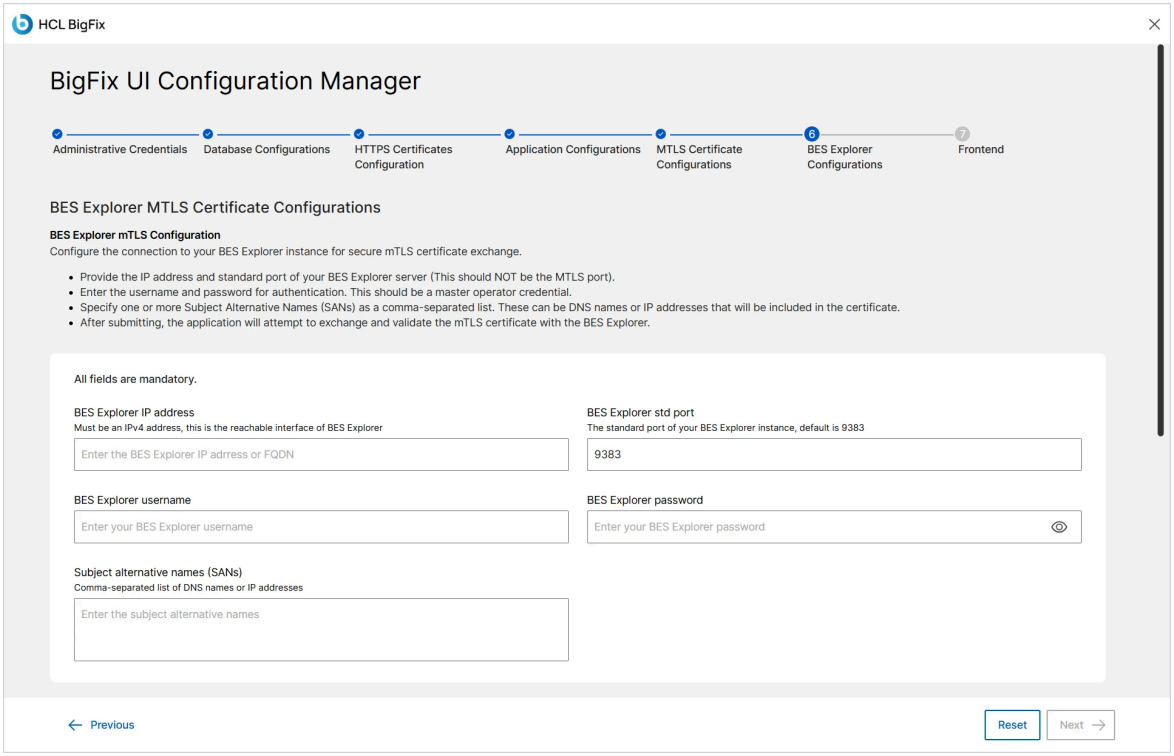
The **BES Explorer MTLS Certificate Configurations** step ensures that the BigFix UI can connect to the BES Explorer instance with valid credentials, host details, and port settings. The screen shows the subtitle **BES Explorer mTLS Configuration** and the description *"Configure the connection to your BES Explorer instance for secure mTLS certificate exchange."*

The screen lists the following guidance:

- Provide the IP address and standard port of your BES Explorer server (This should NOT be the MTLS port).
- Enter the username and password for authentication. This should be a master operator credential.
- Specify one or more Subject Alternative Names (SANs) as a comma-separated list. These can be DNS names or IP addresses that will be included in the certificate.
- After submitting, the application will attempt to exchange and validate the mTLS certificate with the BES Explorer.

All fields are mandatory.

1. After completing MTLS Certificate Configuration, click **Next** button to open the **BES Explorer MTLS Certificate Configurations** screen.



2. If the BES Explorer mTLS certificate is already configured, review the **Current BES Explorer mTLS Configurations** summary.

   The summary lists the following read-only values:

   - ID
   - Alias
   - Type
   - Std port
   - mTLS port
   - Masthead serial number
   - Target
   - Ip address
   - Subject alternative names (SANs)
   - Created at
   - Gather url
   - Certificate exchange API
   - Issue chain hash

   To validate the existing certificate, click **Test mTLS certificate**.

To change these values, click the **here** link in the *"Click here to change your BES Explorer mTLS Certificate Configuration"* text. To keep the current configuration, click **Next**.

3. In the **BES Explorer IP address** field, enter the reachable interface of the BES Explorer instance.

   The field helper text reads *"Must be an IPv4 address, this is the reachable interface of BES Explorer"*. Enter an IPv4 address or an FQDN.

4. In the **BES Explorer std port** field, specify the standard port number for the BES Explorer instance.

   The field helper text reads *"The standard port of your BES Explorer instance, default is 9383"*. The default port is 9383.

   Important: Enter the standard port. Do not enter the mTLS port.

   Note:

   This port is the default port used for the initial certificate exchange between BigFix and BES Explorer. After the certificate exchange is complete, communication with the BES Explorer uses port 9387.

   If a custom port is specified for the initial certificate exchange, the port used for subsequent communication is the custom port + 4.

5. In the **BES Explorer username** field, type the BigFix console operator username.

   Note: Use a master operator credential.

6. In the **BES Explorer password** field, type the password for the BigFix console operator.

   To verify the value that you typed, click the eye icon at the end of the field.

7. In the **Subject alternative names (SANs)** field, enter one or more DNS names or IP addresses.

   The field helper text reads *"Comma-separated list of DNS names or IP addresses"*.

   - Separate multiple values with commas.
   - These values are included in the certificate.

8. Verify all values, then click **Next** to proceed to the Frontend Deployment step.

   After you submit the values, the application attempts to exchange and validate the mTLS certificate with the BES Explorer.

Related information

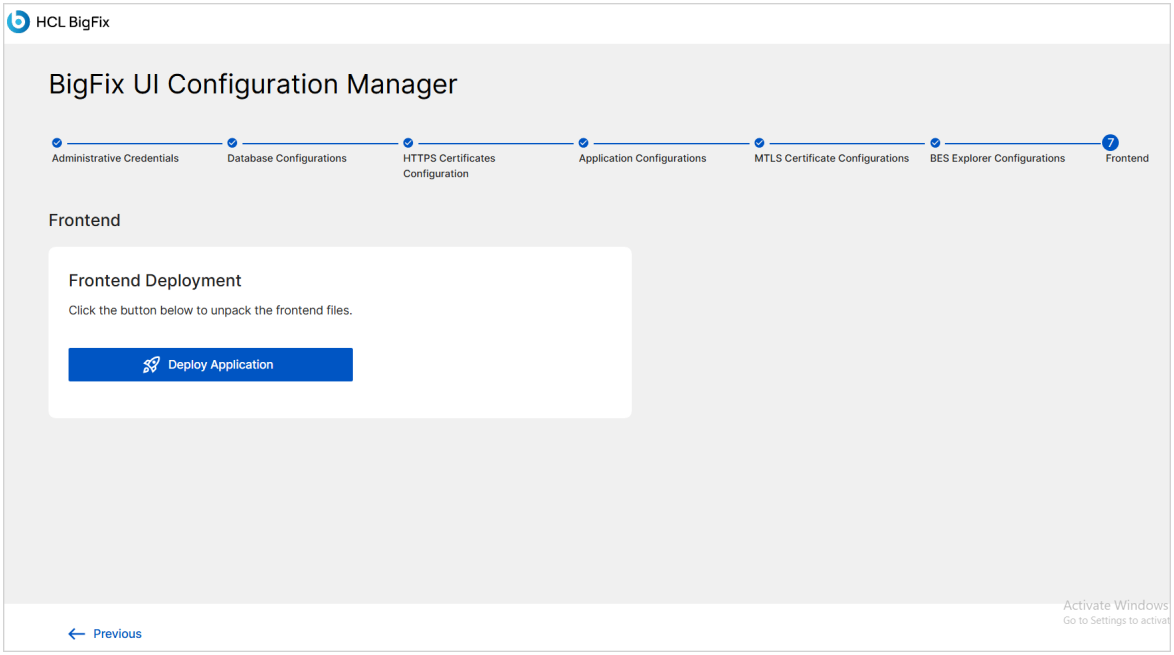
Deploy Frontend *(on page 35)*

Deploy Frontend

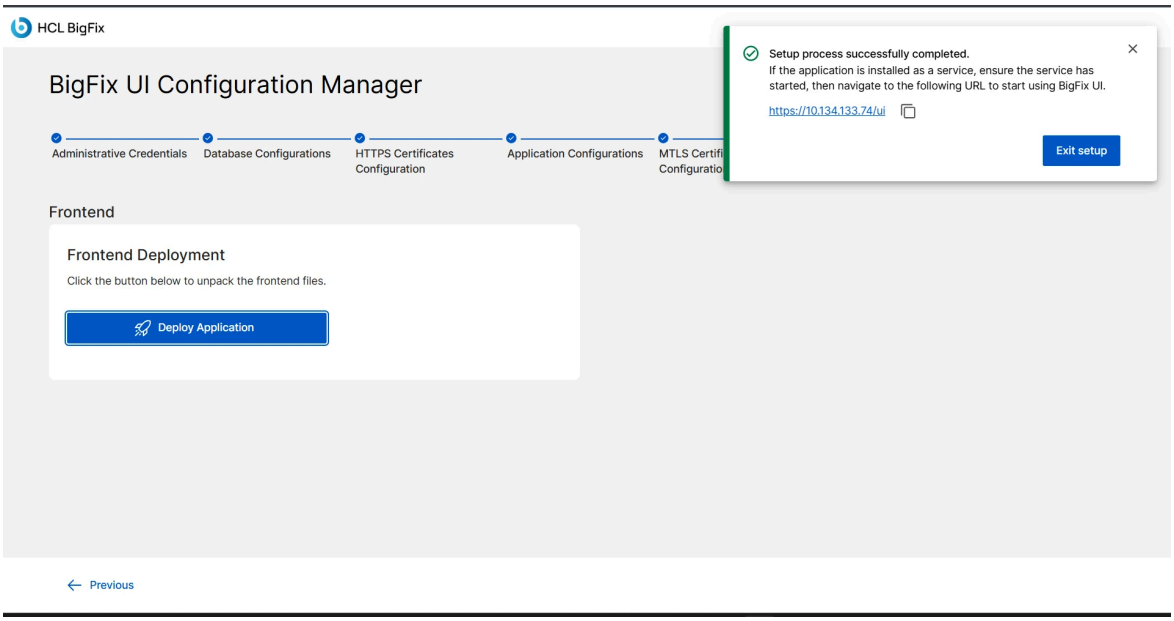
The frontend deployment is the final step in the BigFix UI Configuration Manager setup process. In this step, the system unpacks and installs the frontend files required for the application's user interface. Deploying the frontend ensures that all previously configured components; administrative credentials, database settings, and certificates are integrated and accessible through the application interface. Completing this step finalizes the setup and makes the BigFix UI ready for use.

This task explains how to deploy the frontend component in the BigFix UI Configuration Manager to complete the setup process.

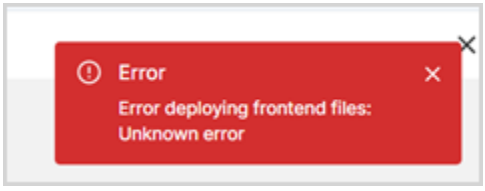
1. After completing BES Explorer configurations, click **Next** button to open the **Frontend** screen.
2. Click the **Deploy frontend** button to begin unpacking the frontend files.



If the deployment is successful, a message such as "Frontend deployed successfully" appears.



If deployment fails, an error message is displayed ("Error deploying frontend files: Unknown error")



3. In the **Setup process successfully completed** pop-up, review the suggested BigFix UI URL.

   The pop-up confirms that the setup finished and shows the URL to use to start using BigFix UI. If the application is installed as a service, ensure that the service has started before you browse to the URL.

   Click the copy icon next to the URL to copy it to the clipboard.

4. Click **Exit setup** to close the BigFix UI Configuration Manager.

How the suggested URL is built

The setup process derives the suggested URL from the **CORS allowed origins** value that you set in the Configure Application Settings *(on page 22)* step:

- If **CORS allowed origins** is `*` (the default), the URL is `https://127.0.0.1/ui`. The port is added only if it is not 443, for example `https://127.0.0.1:8443/ui`.
- If **CORS allowed origins** has a custom value, the URL is the first origin in the list plus `/ui`. For example, `https://192.168.0.7,https://bfui.myserver.com` gives `https://192.168.0.7/ui`.

Important: Use `*` for development only. In production, set a comma-separated list of trusted origins instead.

Related information

CORS allowed origins configuration *(on page 25)*

Chapter 4. Getting into BigFix UI

This section provides an overview of the initial setup and startup configurations required to access and use the BigFix UI after deployment. It includes details about configurable variables, port mappings, and the process to start the application.

Before accessing the BigFix UI, administrators must verify environment readiness by reviewing configuration variables, ensuring required ports are available, and starting the UI service.

Configuration Variables

This topic explains the purpose and structure of the AppConfig.yaml file used by BigFix UI. This configuration file enables administrators to manage application behavior and environment-level settings.

The BigFix UI supports a set of configuration variables that administrators can enable or disable to adjust functionality. These variables let you fine-tune the environment, enable advanced reporting features, or provide additional controls to end users.

Configuration variables are typically set in the application configuration file or as part of the environment setup. Once applied, the changes take effect immediately and influence how the application behaves.

The file follows YAML (YAML Ain't Markup Language) syntax, which uses indentation to define hierarchy. Each setting is represented as a key–value pair that determines specific application behavior. Maintaining correct indentation and data types (for example, Boolean, string, or integer) is essential for proper operation.

Purpose

The `AppConfig.yaml` file lets administrators enable, disable, or customize BigFix UI features without modifying the underlying code or restarting the entire environment. These variables help configure system behavior, report management, and feature access dynamically.

Configuration Variables Reference

Table 3. Settings

Key	Property Name	Data Type	Default Value	Description
Settings	`AllowCustomJsonDefinitionReportsView`	bool	`false`	Shows the **View report** tab and the **View report** button, where you preview the output of a scripted report. Works together with `AllowCustomJsonDefinitionReportsExecution`. If either set-

Table 3. Settings (continued)

Key	Property Name	Data Type	Default Value	Description
				ting is `false`, the tab and the button are hidden. If this flag is `false` then "reports created from scripted reports app" will be excluded from report list in all reports page.
Settings	`AllowCustomJsonDefinitionReportsCreation`	bool	`false`	Lets you save or edit scripted reports. If this setting is `false`, the **Save Report** button stays disabled.
Settings	`AllowCustomJsonDefinitionReportsExecution`	bool	`false`	If the flag is `true`, Shows the **View report** tab and the **View report** button, where you preview the output of a scripted report. Works together with `AllowCustomJsonDefinitionReportsView`. If either setting is `false`, the tab and the button are hidden.
Settings	`AllowUseQna`	bool	`true`	Shows the **QnA** tab, where you evaluate Relevance expressions. If this setting is `false`, the **QnA** tab is hidden.
Settings	`LogVerbose`	bool	`false`	Enable to print debug logs; disable to hide them.

Table 3. Settings (continued)

Key	Property Name	Data Type	Default Value	Description
Settings	`LogAuditToSeparate-File`	bool	`true`	Enable to print Audit logs in a dedicated log file separate from main Application logs; disable to hide them.
Settings	`LogConsoleOnlyMsg`	bool	`false`	Enable to log only to console; disable to log to console and file

Table 4. AppConfig

Key	Property Name	Data Type	Default Value	Description
AppConfigType	`ByPassCertificateV-erification`	bool	`true`	Enable to skip SSL/TLS certificate checks; disable to enforce verification.
AppConfigType	`WebServer`	string	`127.0.0.1`	Web server host/address to bind (e.g., 0.0.0.0, localhost).
AppConfigType	`CorsAllowedOrigins`	string	`*`	List of allowed origins for CORS requests. The default value `*` (all origins) is intended for initial setup only; set the allowed origins according to your organization's security policy before production use. For guidance, see CORS allowed origins configuration *(on page 25)*.
AppConfigType	`FIPSModeEnabled`	string	`"off"`	Enables FIPS-compliant Cryptography;Im-

Table 4. AppConfig (continued)

Key	Property Name	Data Type	Default Value	Description
				portant for deployment that must meet US government Standards
AppConfigType	`DBConnectionMaxRetries`	int	`100`	Maximum retries for database connection on failure.
AppConfigType	`DBConnectionBackoffInterval`	int	`2`	0, 1, 2 correspond to immediate, short, and long backoff intervals between DB retries
AppConfigType	`DBConnectionMaxBackoffInterval`	int	`4`	sets the max DB retry backoff.
AppConfigType	`DbSchemaName`	string	`bfui`	Name of the database schema the application will use.
AppConfigType	`LogMaxSizeMb`	int	`10`	Maximum size (in MB) of a log file before it is rotated.
AppConfigType	`LogMaxBackups`	int	`100`	Maximum number of rotated log files to keep.
AppConfigType	`LogMaxAgeDays`	int	`15`	Maximum number of days to retain old log files before deletion.
AppConfigType	`LogCompression`	bool	`true`	Enable to compress rotated log files; disable to keep them uncompressed.
AppConfigType	`LogFormatJSON`	bool	`false`	Enable to output logs in JSON format; disable for plain text logs.

Table 4. AppConfig (continued)

Key	Property Name	Data Type	Default Value	Description
AppConfigType	`AccessTokenDuration-InMins`	int	`15`	Duration in minutes before an access token expires.
AppConfigType	`RefreshTokenDura-tionInMins`	int	`30`	Duration in minutes before a refresh token expires.
AppConfigType	`EnableRenewalRe-freshToken`	bool	`true`	Controls automatic renewal of refresh tokens when users request a new access token near expiration.
AppConfigType	`AutoRenewalRefresh-TokenTimeInMins`	int	`10`	Time, in minutes, before refresh token expiration at which the token is automatically renewed. Applies when `EnableRenewal-RefreshToken` is set to `true`.
AppConfigType	`DisableJobLogs`	bool	`false`	Enable to turn off job execution logging and reduce log volume; disable when troubleshooting job-related issues.
AppConfigType	`EnableDynamicPortS-electionIfInUse`	bool	`false`	Enable to allow the NATS JetStream server and other microservices to automatically find and bind to an available port if the configured port is already in use. **Note:** If the default port is busy or oc-

Table 4. AppConfig (continued)

Key	Property Name	Data Type	Default Value	Description
				cupied in your environment, you can enable dynamic port selection. When active, the system will automatically allocate and start on a new available port within the 8,000 to 9,000 port series.
AppConfigType	`EmailMaxIndividualRecipients`	int	`50`	Maximum number of individual recipients that a user can add to an email notification, such as the **Email to** list of a scheduled report.

Table 5. AsyncConfig

Key	Property Name	Data Type	Default Value	Description
AsyncConfig	`AsyncReportExpirySeconds`	int	`900`	Time, in seconds, after which a completed asynchronous report expires and is removed from the cache.
AsyncConfig	`AsyncReportWorkerCount`	int	`8`	Number of workers that process asynchronous report requests in parallel.

Table 5. AsyncConfig (continued)

Key	Property Name	Data Type	Default Value	Description
AsyncConfig	`AsyncReportWorker-TimeoutSeconds`	int	`1800`	Maximum time, in seconds, that an asynchronous report worker can run before it times out.
AsyncConfig	`AsyncReportsCon-sumerLag`	int	`50`	Maximum number of pending requests (consumer lag) allowed in the asynchronous reports queue.
AsyncConfig	`CachePurgeFrequency-Minutes`	int	`180`	Frequency, in minutes, at which expired asynchronous reports are purged from the cache.
AsyncConfig	`LagMonitorInterval-Seconds`	int	`300`	Interval, in seconds, at which the consumer lag of the asynchronous reports queue is monitored.
AsyncConfig	`MaxAsyncReportsPer-User`	int	`4`	Maximum number of asynchronous reports that a single user can run concurrently. This will be applicable only when you do bulk processing, which will be upcoming in the future.

Table 6. SiteGatherSettings

Key	Property Name	Data Type	Default Value	Description
SiteGatherSettings	`FileExtensionsTo-Gather`	string	`".brpt"`	File extensions that the site gather ser-

Table 6. SiteGatherSettings (continued)

Key	Property Name	Data Type	Default Value	Description
				vice retrieves from external sites. The service gathers each matching report definition.
SiteGatherSettings	`PollingIntervalMinutes`	int	`720`	Interval, in minutes, at which the site gather service polls external sites for new or changed content. The service compares file hashes and downloads only the files that changed.
SiteGatherSettings	`SiteToFilter`	string	`"All"`	Sites that the service gathers from. The default value `"All"` gathers from every subscribed external site.

Table 7. LockBoxConfig

Key	Property Name	Data Type	Default Value	Description
LockBoxConfigType	`Port`	int	`9010`	Port number used by the LockBox service.
LockBoxConfigType	`WebServer`	string	`127.0.0.1`	Web server host/address to bind (e.g., 0.0.0.0, localhost).
LockBoxConfigType	`IntegrityAuditFrequencyInMinutes`	int	`1440`	Frequency, in minutes, of automated integrity audits that check all protected tables for data tampering.

Table 8. ColMgrConfigs

Key	Property Name	Data Type	Default Value	Description
ColMgrConfigs	`ContextTimeoutSeconds`	int	`60`	Maximum time, in seconds, allowed for a Collection Manager request context before it is cancelled.

Table 9. ColMgrSettings

Key	Property Name	Data Type	Default Value	Description
ColMgrSettings	`EnableLegacyScriptedReportsCreateAndModify`	bool	`false`	Allows creating and modifying legacy scripted reports in the Collection Manager.
ColMgrSettings	`RelevanceQueryMaxProperties`	int	`20`	Maximum number of properties allowed in a single relevance query.
ColMgrSettings	`RelevanceQueryMaxPropertyLength`	int	`500`	Maximum length, in characters, of each property in a relevance query.

Table 10. AuthConfigs

Key	Property Name	Data Type	Default Value	Description
AuthConfigs	`port`	int	`9012`	Handles **authentication**, token generation, and validation.
AuthConfigs	`BESUserCompositionPollingIntervalInMin`	int	`5`	Interval, in minutes, at which BigFix UI polls the BES server for changes to BES user composition.

Table 11. NotifConfigs

Key	Property Name	Data Type	Default Value	Description
NotifConfigs	`Port`	int	`9224`	Notification service starts on this port and enables notification center.
NotifConfigs	`KeepaliveIntervalSeconds`	int	`15`	Interval, in seconds, at which keepalive messages are sent to maintain open notification connections.
NotifConfigs	`NotifSoftDelIntervalMinutes`	int	`240`	Interval, in minutes, at which notifications are soft deleted (marked as deleted but retained in the database).
NotifConfigs	`NotifHardDelIntervalMinutes`	int	`1440`	Interval, in minutes, at which soft-deleted notifications are permanently removed from the database.

Table 12. DataService

Key	Property Name	Data Type	Default Value	Description
DataServiceConfig	`port`	int	`9019`	**Data Service** — is the central data layer for managing and serving data to subsystems like FixletStreams, PLA, and CyberFocus.

Table 13. OSConfigs

Key	Property Name	Data Type	Default Value	Description
OSConfigs	`port`	int	`9020`	**OsConfigs - Orchestration Service**, Orchestration service

Table 13. OSConfigs (continued)

Key	Property Name	Data Type	Default Value	Description
				orchestrates fixlet-streams by scheduling and triggering them at the defined times.
OSConfigs	`MaxConcurrentOrchEngineWorkers`	int	`10`	Maximum number of concurrent orchestration engine workers that process queued jobs.
OSConfigs	`OrchEngineTickerIntervalSeconds`	int	`60`	Polling interval, in seconds, for the orchestration engine queue scan.
OSConfigs	`OrchWorkerTimeoutInMinutes`	int	`5`	Maximum runtime, in minutes, for a single orchestration worker before its context is cancelled.
OSConfigs	`ScheduledReportAccessTokenExpirationHours`	int	`720`	Lifetime, in hours, of scheduled report access tokens. Tokens embedded in scheduled report email notifications become invalid after this duration. The default value is 720 hours (30 days).
OSConfigs	`ScheduledReportsQuotaPerUser`	int64	`10485760`	Per-user storage quota, in bytes, for scheduled report archives that the orchestration service generates. The default value is 10485760 bytes (10 MB).

Table 14. FileServerConfig

Key	Property Name	Data Type	Default Value	Description
FileServerConfig	`MaxUploadSizeBytes`	int64	`5368709120`	Maximum allowed file upload size, in bytes. The default value is 5368709120 bytes (5 GB).
FileServerConfig	`AllowedExtensions`	string	`*`	Comma-separated list of file extensions that the file server accepts. Use `*` to allow all file types.
FileServerConfig	`StoragePath`	string	None	Storage path for uploaded files. Specify an absolute path to the directory where the file server stores files.
FileServerConfig	`IPACL`	string	`127.0.0.1`	List of IP addresses that can access the file server download endpoint. Specify the IP addresses of the root servers that are permitted to download files.

Table 15. Miscellaneous Configs

Key	Property Name	Data type	Default value	Description
LbUnsecureConfigs	`port`	int	`9011`	Handles unsecured load balancer service that routes HTTP traffic between internal components.
RnsConfigs	`port`	int	`9013`	**Relay Notification Service** — manages event propagation and notifications.

Table 15. Miscellaneous Configs (continued)

Key	Property Name	Data type	Default value	Description
sqlRetConfigs	`port`	int	`9016`	**SQL Return Service** — handles SQL-based queries, fixlet streams, or analytics results.
ExpRetConfigs	`port`	int	`9018`	**Export/Reports Return service** — processes and serves reporting data.
DexRetSettings	`MaxConcurrentCalls`	int	`8`	**DEX Retriever Service** — maximum number of concurrent calls that the DEX Retriever service can process.
IvrRetConfigs	`Port`	int	`9078`	**IVR Retriever Service** — port number reserved for the IVR Retriever service.
ScmRetConfigs	`Port`	int	`9079`	**SCM Retriever Service** — port number reserved for the SCM Retriever service.
SwdRetConfigs	`Port`	int	`9080`	**SWD Retriever Service** — port number reserved for the SWD Retriever service.
BfiRetConfigs	`Port`	int	`9082`	**BFI Retriever Service** — port number reserved for the BFI Retriever service.
WkflRetConfigs	`Port`	int	`9083`	**Workflow Builder Retriever Service** — port number reserved for the Workflow Builder Retriever service.

Table 15. Miscellaneous Configs (continued)

Key	Property Name	Data type	Default value	Description
PatchRetConfigs	`Port`	int	`9084`	**Patch Retriever Service** — port number reserved for the Patch Retriever service.

Editing Configuration Variables

1. Navigate to the `data\config\` folder in your BigFix UI installation directory.

   For a default installation, this is `C:\Program Files (x86)\BigFix Enterprise\BES UI\data\config\`.

   If you specified a custom directory during setup, this is `<custom_install_path>\data\config\`.

2. Open the `AppConfig.yaml` file in a text editor such as Notepad++ or VS Code.
3. Locate the relevant section and modify the desired property.
4. Save the file.
5. Restart the BigFix UI service for the changes to take effect.

   Note:

   When you modify variables under the `Settings` section in the `AppConfig.yaml` file, restarting the BigFix UI service is not required. The changes take effect immediately after the file is saved.

Note:

- Use lowercase Boolean values (true or false).
- Verify correct YAML indentation; syntax errors can prevent startup.
- Back up the configuration file before editing.
- Apply changes before starting the application.

Related information

Starting BigFix UI *(on page 52)*

Starting BigFix UI

This topic explains how to launch the BigFix UI after completing deployment and configuration.

Starting BigFix UI

1. Start the BigFix UI
   - Open a command prompt and run the following command:

     bfui.exe --start

   - If you selected the Windows service option during setup, the service will start automatically when the system boots.

   Note: Installing BigFix UI as a Windows service is recommended. If you installed the service during setup, it starts automatically when the system starts. Otherwise, start BigFix UI manually from the command line.

2. Access the BigFix UI
   - Once the service is running, open a supported browser (Microsoft Edge or Google Chrome) and navigate to the BigFix UI URL **https://<host>[:port]/ui**.
     - **Host** refers to the system where the BigFix UI is installed.

       This can be localhost (if accessing from the same system) or a remote server hostname or IP address (if accessing the UI from another machine).

     - **Port** refers to the Main Application Service port defined in the configuration file.

       The default port is 443.

   - The login page is displayed.
3. First Login: Log in using the credentials you created during the setup process. This login will only be used for configuring the first administrator; it is not meant to be used for running other use cases in the UI.
4. Create an Administrator User
   - Navigate to the **Identity & Access** section.
   - Create a new user and assign them the role of Master Operator.
   - This administrator account will be used for all future operations.

   For details about creating users and assigning roles, see User & Role Management in the BigFix UI User's Guide.

5. After the administrator user is created, the BigFix UI is ready for use. For details on navigation, dashboards, and features, see the BigFix UI User's Guide.

External Reference

For details on how to use the BigFix UI once it has started, see the BigFix UI User's Guide.

Chapter 5. Upgrade

This topic explains how to upgrade an existing BigFix UI installation to a newer version with the BFUI.exe --upgrade command.

Before you start the upgrade, complete these tasks:

- Download the new version of the BigFix UI build and extract it to a directory.
- Check the version of both builds. Run BFUI.exe --version from the `cmd\bfui` folder of each build. The command reports the BFUI version, the database manual patch version, and the embedded frontend versions.
- Confirm that the new version is higher than the installed version.
- Take a full backup of the existing database.
- Run the upgrade with a user that has administrator privileges. Without administrator privileges, the command fails with the message *"Failed to determine administrator privileges"*.

The upgrade command copies the new executable and service manager to the installed location, migrates the database, upgrades the `AppConfig.yaml` file, and extracts the new frontend. Run the command from the new version.

1. Open a command prompt as an administrator and navigate to the `cmd\bfui` folder of the new extracted build.
2. Run the upgrade command.

   ```
   BFUI.exe --upgrade
   ```

   The command detects the existing installation automatically.
3. Confirm the upgrade.

   The command displays the target version and asks whether to proceed. Enter `yes` to continue, or `no` to cancel the upgrade.

   Note: If the target version is not newer than the installed version, the command displays a warning and asks for confirmation. Enter `no` to cancel. Downgrading is not recommended.
4. Enter the instance admin username and password when prompted.

   These are the application instance credentials that you created during the setup. If the credentials are not correct, the command reports *"Authentication failed"* and stops.
5. Choose whether to back up the local application state when prompted.

Before it changes anything, the command copies the local application state (the `cmd`, `data`, and `wwwroot` folders) to a backup folder. The command displays the backup location, the space required, and the space free on the install volume.

Enter `yes` to take the backup. If you already backed up the application, or if the volume cannot accommodate the space shown, type `I understand the risk` to skip the backup. If you skip the backup, you accept the risk.



Note: The backup does not include the application database. Back up the database separately before you continue.

6. Start the **BES UI** service when prompted.

   The command stops the **BES UI** service before the upgrade starts. After the upgrade completes, the command asks whether to start the service.

   Enter `yes` to start the service. The service comes up a few seconds after the program exits. Enter `no` to leave the service in the stopped state and start it manually later.

The command reports the progress of the upgrade step by step, in the following order:

1. Validating the installation folder
2. Backing up the installed version
3. Verifying database permissions
4. Running the integrity audit with the previous version
5. Capturing the schema snapshot with the previous version
6. Analyzing upgrade impact
7. Running database migrations
8. Installing binaries and configuration
9. Extracting the frontend

Under **Summary**, the command reports that the upgrade to version 1.x.x.xxx completed, and the time that the upgrade took. For example, *"Upgrade to version 1.0.4.542 completed in 2m6s."* The summary also lists the backup location, the detail log, and the transcript log.

Verify that the **BES UI** service is running and that you can access the BigFix UI.

Related information

Uninstallation *(on page 55)*

Chapter 6. Uninstallation

This topic explains how to uninstall the BigFix UI application with the BFUI.exe --reset command.

The BFUI.exe command-line utility provides the `--reset` flag. Use `--reset` to remove the application and return the installation to a clean state.

Uninstall BigFix UI

The uninstall operation removes the BigFix UI application from the system. It performs the following actions:

- Stops and removes the **BES UI** service.
- Deletes the data folders, such as `certs`, `cache`, `configs`, `db`, `wwwroot`, and `pubCache`, except the `logs` folder and its content.
- Deletes the BigFix UI registry keys.

To uninstall BigFix UI, perform these steps:

1. Navigate to the `cmd\bfui` folder in your BigFix UI installation directory.

   For a default installation, this is `C:\Program Files (x86)\BigFix Enterprise\BES UI\cmd\bfui`.

   If you specified a custom directory during setup, this is `<custom_install_path>\cmd\bfui`.

2. Open a command prompt as an administrator and run the following command:

   BFUI.exe --reset

3. When the command prompts for confirmation, enter `yes`.

4. Delete the remaining application files from the BigFix UI installation directory.

5. Access your Microsoft SQL Server database with administrative credentials and drop the BigFix UI database.

Note: After the uninstall completes, the `InstallFolder` value is removed from the `HKEY_LOCAL_MACHINE\SOFTWARE\BigFix\BES UI` registry key.

Note: The `--reset` flag does not delete the whole `fileServer` path, because the setup cannot determine which part of that folder tree belongs to the application. Review the `fileServer` folder and delete the application content manually.

Related information

Upgrade *(on page 53)*

Appendix A. Support

For more information about this product, see the following resources:

- BigFix Support Portal
- BigFix Developer
- BigFix Playlist on YouTube
- BigFix Tech Advisors channel on YouTube
- BigFix Forum

Notices

This information was developed for products and services offered in the US.

HCL may not offer the products, services, or features discussed in this document in other countries. Consult your local HCL representative for information on the products and services currently available in your area. Any reference to an HCL product, program, or service is not intended to state or imply that only that HCL product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any HCL intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-HCL product, program, or service.

HCL may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

For license inquiries regarding double-byte character set (DBCS) information, contact the HCL Intellectual Property Department in your country or send inquiries, in writing, to:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

HCL TECHNOLOGIES LTD. PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. HCL may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-HCL websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this HCL product and use of those websites is at your own risk.

HCL may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by HCL under terms of the HCL Customer Agreement, HCL International Program License Agreement or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-HCL products was obtained from the suppliers of those products, their published announcements or other publicly available sources. HCL has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-HCL products. Questions on the capabilities of non-HCL products should be addressed to the suppliers of those products.

Statements regarding HCL's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to HCL, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. HCL, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS," without warranty of any kind. HCL shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:
© (your company name) (year).
Portions of this code are derived from HCL Ltd. Sample Programs.

Trademarks

HCL Technologies Ltd. and HCL Technologies Ltd. logo, and hcl.com are trademarks or registered trademarks of HCL Technologies Ltd., registered in many jurisdictions worldwide.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other product and service names might be trademarks of HCL or other companies.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the HCL website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of HCL.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of HCL.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

HCL reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by HCL, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

HCL MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.