

OpenText™ FirstClass® Web Services CE 16.2 SP3 (16.2.3)

Release Notes

Product Released: 2022-07-04

1 Introduction

These Release Notes provide an overview of FirstClass® Web Services 16.2 SP3 (16.2.3), including new features, delivery information, and supported platforms.

OpenText recommends that you read these Release Notes in conjunction with the documentation included with the software package. If any conflicts exist, the Release Notes supersede the other documentation.

We also recommend that you check OpenText [My Support](#) for any patches or documentation updates that may have been posted after the initial release of this product.

1.1 Release Notes revision history

Revision date	Sections revised	Description of revisions
2024-06-25	8 Fixed issues 9 Known issues	Updated notes for 16.2 SP3 (16.2.3 build)

2 About FirstClass® Web Services16.2 SP3 (16.2.3)

This section provides an overview of FirstClass® Web Services16.2 SP3 (16.2.3).

The FirstClass Web Client (or any other web-based client) uses HTML 5.0 to communicate with the FirstClass web server. Requests issued to the FirstClass web server are translated into the FirstClass-proprietary protocol FCP and sent to the FirstClass server for processing. The FirstClass server responses are also encapsulated in the FCP protocol, which is then processed by the FirstClass web server and subsequently sent back to the web client as HTML pages, JSON data, CSS style data and/or JavaScript code.

The FirstClass web server also supports WebSocket and/or AJAX connections to facilitate live web client updates with changes that occurred on the FirstClass server (new mail notifications, calendar alarm reminders, container content changes, and so on).

The FirstClass web server provides mechanisms for localization and customization. Basic language localization is done automatically by the FirstClass web server based on the language settings in the web browser **Accept-Language** HTTP header. Additionally, this API provides a JSON mechanism to override browser settings or supply an independent parameter for language localization for each session.

2.1 New features

There are no new features in this release.

2.2 Discontinued and deprecated features

There are no discontinued features in this release.

3 Downloads

Downloads for FirstClass® Web Services are available on [My Support](#).

Downloads for OpenText™ FirstClass® Web Services includes:

- FirstClass Web Services162102N.msi
- fcws-16.2.2.102N-linux-py3.7.tgz

4 Documentation

Documentation for FirstClass® Web Services is available on [My Support](#). The documentation for OpenText™ FirstClass® Web Services includes:

- FirstClass Web Services162102N.msi
- fcws-16.2.2.102N-linux-py3.7.tgz

Following are the required Python packages, which are included with and will be installed, if necessary, by the listed installers:

Package	Python Package Index	pip Installation	Minimum Version	Reason for inclusion
CherryPy	https://pypi.org/project/CherryPy/	pip install CherryPy	18.4.0	A WSGI compliant web server included as a requirement for non-Apache based web server installs.
Mako	https://pypi.org/project/Mako/	pip install Mako	1.1.0	Directly required by the FCWS product.
MarkupSafe	https://pypi.org/project/MarkupSafe/	pip install MarkupSafe	1.1.1	A package required by the Mako package above.
WebOb	https://pypi.org/project/WebOb/	pip install WebOb	1.8.7	Directly required by the FCWS product.
lxml	https://pypi.org/project/lxml/	pip install lxml	4.9	Powerful and Pythonic XML processing library combining libxml2/libxslt with the Element Tree API.

six	https://pypi.org/project/six/	pip install six	1.13.0	Six is a Python 2 and 3 compatibilities libraries.
-----	---	-----------------	--------	--

4.1 Related videos

For additional video content, see the FirstClass® Web Services Videos page on [My Support](#).

5 Supported environments and compatibility

This section provides details about supported platforms, systems, and versions.

5.1 Supported Systems

OpenText™ FirstClass® Web Services 16 requires a 64-bit (x64) operating system.

- A 64-bit version of Python 3.7 must be present on the computer on which you are installing. Further details are listed.
- A WSGI compliant web server. To install and configure, this beta release will install and use a CherryPy web server. An Apache web server can also be used but needs the mod_wsgi module installed or configured along with additional configuration, which will follow this release.
- The FirstClass Server you are signing into must be running version 16.0 or later.
- Memory requirements: A user session may use up to approximately 10MB on the Web Services machine, so based on that, you would currently require approximately 1GB of memory per 100 sessions.
- CPU requirements:
 - PC: 2 GHz or faster processor

Supported Operating Systems

Linux (64 bit)

- Linux Kernel 3.x
- Linux Kernel 4.x
- Linux Kernel 5.x

Windows (64 bit)

- Windows Server 2012 R2 *
- Windows Server 2016*
- Windows Server 2019 *

5.2 OpenText product compatibility

The section provides details about which versions of other OpenText products are compatible with this release of OpenText™ FirstClass® Web Services 16.2 SP3 (16.2.3).

 **Note:** For the latest compatibility information for OpenText products, refer to the Compatibility Matrix (<https://knowledge.opentext.com/go/matrix>) on OpenText My Support.

Product name	Version	Notes
FirstClass® Web Services	16.2 SP3*	FCWS
FirstClass® Client	16.2 SP2*	FCC
FirstClass® GO	16.2 SP3*	FCGO
FirstClass® Sync Services	16.2 SP2*	FCSS
FirstClass® Internet Services	16.2 SP3*	FCIS
FirstClass® Directory Services	16.0*	FCDS
FirstClass® External Integration Services	16.1*	FCEIS
FirstClass® Application Services	4.0 B4534	FCAS
FirstClass® Voice Services	8.3 or later	FCVS
FirstClass® Mobile	16.0*	FC Mobile

*Earlier supported versions may still work with the new server, but new features and fixes will not be available.

5.3 Language support

OpenText™ FirstClass® Web Services is currently localized in the following languages. Future releases may add additional languages.

Component	Languages										
	EN	DA	DE	ES	FI	FR	IT	JA	NL	NO	SV

OpenText™ FirstClass® Web Services 16.2	B	B	B	B	B	B	B	B	B	B	B
---	---	---	---	---	---	---	---	---	---	---	---

UI = user interface only

B = both user interface and online help

6 Installation and upgrade notes

This section provides additional installation and upgrade information, including related or third-party product information and any required critical patches.

6.1 Installation notes

Before you install OpenText™ FirstClass® Web Services, review these additional installation notes and verify related product or third-party product requirements.

New Installations

Windows Installation

Installation Overview

- The attached fcws-16.2.3N-linux-py2.6.tgz file is for installing/updating the FirstClass Web Services on Linux.

Installation Prerequisites (one-time requirements): Python

On Windows systems, Python 3.7.* is required. If you have another version of Python installed, you either must remove it or change the path for Python execution to point to the 3.7 executable after installation.

- The following instructions assume that the install is on your C:\ drive and you are signed in as Administrator- you can adjust accordingly.
- Download and install the Python 3.7.9 [Windows x86-64 executable installer](#).
- Run the installer, verify, and take note of the default installation location, and then click Install Now.
- At the end of the installation, check the box to remove the MAX_PATH length of 260 characters (optional). The option may not appear if it was previously set in an earlier Python installation.

After the installer has finished, you will need to add Python37 and Python37/Scripts to the system's Environment Variables, under System variables:

- Open the Environment Variables form, which is found on the System Properties form, by clicking the button. The System Properties form can be found in Control Panel > System > Advanced system settings.
- On the Environment Variables form, select Path in System variables, and click edit.
 - Ensure you are selecting PATH from the system variables rather than the user variables, which may have been added by Python. This is a known issue with their installer.
- Click new and paste in the first path location, then repeat the process for the second location, using the locations you noted earlier if they differ from this example:
 - C:\Users\Administrator\AppData\Local\Programs\Python\Python37
 - C:\Users\Administrator\AppData\Local\Programs\Python\Python37\Scripts
- After being added, the entries will appear at the bottom of the list. If you have any other python versions (such as 2.7), you will need to use the Move Up button to ensure the Python37 listings come before them so that 3.7 is the default version called by Python.
- Press OK to close the dialog box, OK to close the Environment Variables form, and OK again to close the System Properties form.
- Restart the computer to set the Path.

After the installation, verify the PATH by typing the following in a cmd window:

```
path
```

If it is correct, you should see the entries for Python37 and Python37\scripts listed ahead of any other Python versions.

You can now verify that the correct Python version is being used by typing the following in a cmd window:

```
python --version
```

It should return the installed 3.7 version such as:

```
Python 3.7.9
```

Installation Instructions

- Download and run the 'FirstClass Web Services162300N.msi' installer.
- Type 'Yes' to accept the FirstClass End User License Agreement.
- This will install the necessary files and start the Python based installer script in a command prompt window. Wait for the Python install script to complete.

Note: If upgrading, you can choose to use the default install location or change it to install into the same location you have already installed it.

- A console window will appear that will run the installer script. After completion, you should see the 'Install Complete' message within the console.
- The installation is now complete.

After the installer script has completed, you should have the following structure, and will be able to start Web Services by double-clicking the startfcws.bat file.

Contents:

- Program Files\FirstClass Web Services\fcws-16.2.3.000N-win package directory.
- "startfcws.bat" is a preconfigured startup script for starting the FirstClass Web Services. This script can be customized and does not get overwritten by upgrades.
- "PkgArchives" is used to archive the previously installed package so that any customized config does not get lost. The following user configuration is copied from the archived package during the install:
 - fcws.cfg
 - any user file in the root fcws directory
 - startup command or batch file
 - processes.txt
 - css\custom.css
 - css\ -- any user added css files
 - css\themes\ -- any user added themes
 - images\action_more_ios48_48*.png
 - images\checkbox18_sprite*.png
 - images\sprite_menu*.png
 - images\action_ios*.png
 - images\toolbar_icons32_sprite.png
 - images\brand.png
 - images\icons\ -- any user added icons
 - images\ --- any user added images
 - localized\<lang cde>/user/html/ -- any user added html files.

Note: If this is a new installation, you will need to configure the following items:

1. The 'fcws.cfg' file located in the "C:\Program Files\FirstClass Web Services\fcws-16.2.3.000N-win\fcws" folder.

This file is used to configure the FirstClass servers your instance of Web Services can connect to and the FCP port.

- **FCSPort** - The FCP port over which FCWS will connect to your FirstClass Server
- **ValidFCServers** - A ';' delimited list of <Server Label>, <Server Address> pairs. The label will be displayed to the user at the sign in screen in a drop-down list only if more than one server option exists.

Example: testServer,192.168.1.54; FirstClass,fc.mydomain.com

NOTE 1: In case of one entry in the list, the drop-down list on the sign in screen is suppressed.

NOTE 2: If administrators want to make their system content available through the new API, one of the entries in the list must be **default**.

Example:

ValidFCServers=default,fc.mydomain.com;testServer,192.168.1.54 or simply

ValidFCServers=default,fc.mydomain.com if single server

- **WebSocketEnabled** - This option should currently be set to 1 if running the CherryPy web server, and 0 if running an Apache web server.
- **LoggingLevel** - Current values are: DEBUG,INFO,WARN,ERROR,CRITICAL,FATAL

fcws.cfg sample:

FCSPort=510

ValidFCServers=default,fc.mydomain.com

WebSocketEnabled=1

2. The 'startfcws.bat' file located in the 'C:\Program Files\FirstClass Web Services' folder.
 - This file is currently in place to start the installed web server with the proper configuration.
 - With the default installed file, it should start a web server on port 80. If you then add SSL certificates as mentioned below, then it will start a HTTPS server on port 443.
 - By default, it only tries to start a HTTPS server on port 443 unless no SSL certificate files are found in which case it falls back to starting a HTTP server on port 80.
 - You can specify it starts a HTTP server in addition to the HTTPS server by adding the '-e' as an arg.
 - With the current release, it is configured for starting the CherryPy server through passing parameters to the cherry.pyc file in the FCWS folder.

The startup parameters currently supported are as follows:

Usage: cherry.pyc [options]

Options:

-h, --help

show this help message and exit

-H HOST, --host=HOST

Specify the host name for the CherryPy Server. 0.0.0.0 will serve on all interfaces. (default: 0.0.0.0)

-f FCS_PORT, --fcserverport=FCS_PORT

Specify the FCP port number used to connect to the FirstClass Server.
(default: 510)

-s SSL_PORT, --sslport=SSL_PORT

Specify the SSL / HTTPS port number. (default: 443)

-k SSL_PRIVATE_KEY, --key=SSL_PRIVATE_KEY

Specify the SSL key file name or path. (default: key.pem)

-c SSL_CERTIFICATE, --cert=SSL_CERTIFICATE

Specify the SSL cert file name or path. (default: cert.pem)

-i SSL_CERTIFICATE_CHAIN, --intermediate=SSL_CERTIFICATE_CHAIN

Specify the SSL intermediate certificate file name or path. (default: none)

-t SSL_THREAD_POOL, --sslthreads=SSL_THREAD_POOL

Specify the number of threads for the SSL CherryPy Server. (default: 500)

-e, --enablehttp

Enable the HTTP server. (default: False)

-p PORT, --httpport=PORT

Specify the HTTP port number. (default: 80)

--httpthreads=HTTP_THREAD_POOL

Specify the number of threads for the HTTP CherryPy Server. (default: 100)

(ie.) Modifying the startup line in the 'startfcws.sh' file to either of the following would cause the web server to start up both a HTTPS server on port 8443 and a HTTP server on port 8000.

```
/usr/bin/python3.7 "/opt/FirstClass Web Services/fcws-  
16.2.2.102Nlinux/fcws/cherry.pyc" -H 192.168.1.111 -s 8443 -ep 8000
```

```
/usr/bin/python3.7 "/opt/FirstClass Web Services/fcws-  
16.2.2.102Nlinux/fcws/cherry.pyc" --host=192.168.1.111 --sslport=8443 --enablehttp -  
httpport=8000
```

Note: On pre-Windows 2015 server installations, the path above to the cherry.pyc file is expanded to the actual path through the Python site packages.

Starting & connecting to OpenText™ FirstClass® Web Services:

- For starting FirstClass Web Services, double-click the 'startfcws.bat' file located in the 'C:\Program Files\FirstClass Web Services' folder.
- Pressing CTRL+C will stop the FirstClass Web Services.

Connecting to your FCWS server.

- Open a web browser and type in the URL, which points to the c on which you installed the FirstClass Web Services.
- Example: `http://www.mydomain.com` or if you configured it to start on an alternate port; `http://www.mydomain.com:8080`
If all goes well, a sign-in screen will appear.

How to run FCWS as a Service on Windows:

- Install the latest version of the Python package: pywin32. Type "pip install -U pywin32" into a command line and click return.
- Create the FCWS Service:
 - Open a command prompt and navigate to the fcws directory (C:\Program Files\FirstClass Web Services\fcws-16.2.2.102N-win\fcws)

- To create a windows service, run the utility program `fcws_svc.pyc` as follows:

```
python fcws_svc.py [options] install"
```

options:

`--username domain\username` : The Username the service is to run under

`--password password` : The password for the username"

`--servicename name` : Service name in Registry.

- More options can be found if you run `fcws_svc.py`
- If options are omitted, the service will run under the Local System account with the default name FCWSSvc.

Note: If FCWS is installed in the 'Program Files' directory, the user account must be either Local System or Administrator account to be able to create and write to the log file.

- To create additional instances of the FirstClass Web Services Windows Service, use the install command with the option 'servicename' and give it a new service name.
- After installation, it can be controlled using the 'sc' utility from the command prompt.

`svcName` is the service name established in 'Create FCWS Service'.

To start service: sc start svcName <list of CherryPy command line arguments>"
To stop service: sc stop svcName
To delete service: sc delete svcName

- Alternatively, a list of CherryPy command line arguments can be appended to the 'ImagePath' entry in the service Registry entry. Once appended, the service can be set to start Automatically or with the Windows 'Services' utility.
- All FCWS services instances will be created in
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\<fcws service name>.

Linux Installation

Installation Overview

- The attached fcws-16.2.3.000N-linux-py2.6.tgz file is for installing/updating the FirstClass Web Services on Linux.

Installation Prerequisites (one-time requirements): Python

The OpenText™ FirstClass® Web Services installer must be run as a user with a userid of 'fcadmin' (and any full name, but 'FirstClass Administrator' is recommended). This user must be a member of the 'adm' group. Create this group, if it does not exist.

- After creating this user and adding it to the 'adm' group, sign out and sign in as 'fcadmin'.

Python 3.7.* is required. On Linux, Python3 will in most cases already be present even if 3.7.x is not the default Python3 version.

You can test if Python 3.7.x is installed by typing the following in a terminal window:

```
python3.7 --version
```

If it is present, it should return the installed version as:

```
Python 3.7.9
```

If a Python 3.7 version is returned, install the FirstClass software.

- If a 3.7.x version is not installed, verify that a Python 3.x version is installed by typing `python3 --version` in a terminal window.
 - If you do not have Python 3.x installed, install the default Python v3.x for your Linux installation using one of these commands as root:

Ubuntu / Debian / Mint: `sudo apt-get install python3`

CentOS / RHEL (Red Hat): `sudo yum install python`

- If the installation cannot complete, you may need to prepare your machine for the Python installation by updating the installer packages. For example, `sudo yum -y update` or `sudo apt-get update`.
- If a Python3 version other than 3.7 is installed, you can install 3.7 manually by following these instructions:
 - In a terminal window, download the software:
 - `sudo cd /usr/src`
 - `sudo wget https://www.python.org/ftp/python/3.7.9/Python-3.7.9.tgz`
 - Then, extract the downloaded package:
 - `sudo tar xzf Python-3.7.9.tgz`
 - Finally, compile the Python source code on your system using `altinstall`:
 - `cd Python-3.7.9`
 - `sudo ./configure --enable-optimizations`
 - `sudo make altinstall`
 - Once complete, verify the installation using the `python3.7 --version` command, which should now return 3.7.9. You can now install the FirstClass software.

Note: Depending on your Linux OS installation and previously installed applications, you may see errors when using Python or during its installation process.

- Here are some of the more common problems and suggested solutions:
 - On Ubuntu, if you see compiler errors, you can use:


```
sudo apt install build-essential openssl-devel bzip2-devel
libffi-devel make
```
 - For yum based systems, if it is a Python3: ImportError: No module named '_ctypes' error when using Value from module multiprocessing, use:


```
sudo yum -y install gcc gcc-c++
sudo yum -y install zlib zlib-devel
sudo yum -y install libffi-devel
```
 - If python or pip commands are not recognized, create python and pip symlinks:


```
sudo ln -s /usr/local/bin/python3.7 /usr/bin/
sudo ln -s /usr/local/bin/pip3.7 /usr/bin/pip3
```
 - If you get a shared library error, activate the shared library location:


```
ldconfig /usr/local/lib
```
 - Please refer to the documentation for your version of Linux, additional information on compiler dependencies, and correcting errors, if needed.

Installation Instructions

- Ensure the 'fcadmin' account has been created as per the instructions above, the account is a member of the 'adm' group, and that you have signed out and signed in as 'fcadmin'. This is required to ensure that the 'adm' group change is applied.
- Download the attached (fcws-16.2.2.102N-linux-py3.7.tgz) TAR file.
- Start a shell Terminal window, and 'cd' to your download location.
- Extract the files in the installer TAR file, using this command:

```
tar xzf fcws-16.2.2.102N-linux-py3.7.tgz
```
- 'cd' into the installer package directory as in the below example.

```
cd fcws-16.2.2.102N-linux-py3.7.tgz
```
- Start the installer, and answer the prompts as asked.

```
./install.sh
```

Note: If your fcadmin account does not have sudoer privileges, you will be asked to re-run the installer after first becoming 'root' by executing the 'su' command.
- Type 'yes' to accept the End User License Agreement.
- After the installation is complete, you can delete the temporary package directory and downloaded TAR file.

After running the installer script, you should have the following structure:

A 'FirstClass Web Services' folder is created at '/opt/FirstClass Web Services'.

Contents:

- opt/FirstClass Web Services/fcws-16.2.2.102N-linux/ package directory.
- 'startfcws.sh' is a preconfigured startup script for starting the FirstClass Web Services. This script can be customized and does not get overwritten by upgrades. This script is also called by the 'fcwsctl' control script.
- 'PkgArchives' is used to archive the previously installed package so that any customized config does not get lost. The following user configuration is copied from the archived package during the install; ◦ fcws.cfg ◦ any user file in the root fcws directory ◦ startup command or batch file ◦ processes.txt ◦ css/custom.css ◦ css/ -- any user added css files ◦ css/themes/ - - any user added themes ◦ images/action_more_ios48_48*.png ◦ images/checkbox18_sprite*.png ◦ images/sprite_menu*.png ◦ images/action_ios*.png ◦ images/toolbar_icons32_sprite.png ◦ images/brand.png ◦ images/icons/ -- any user added icons ◦ images/ --- any user added images ◦ localized/<lang cde>/user/html/ -- any user added HTML files.

Note: If it is a new installation, configure the following items:

1. The 'fcws.cfg' file located in the '/opt/FirstClass Web Services/fcws-16.2.2.102N-linux/fcws' folder.

This file is used to configure the FirstClass servers your instance of Web Services can connect to and on which FCP port.

- **FCSPort** - The FCP port on which FCWS will connect to your FirstClass Server.
- **ValidFCServers** - A ';' delimited list of <Server Label>,<Server Address> pairs. The label will be displayed to the user at the sign in screen in a drop-down list only if more than one server option exists.
- Example: testServer,192.168.1.54;FirstClass,fc.mydomain.com
Note: If there is only one entry in the list, the drop-down list on the sign-up screen will be suppressed.

Note: If administrators want to make their system content available through the new API, one of the entries in the list must be **default**.

Example:

ValidFCServers=default,fc.mydomain.com;testServer,192.168.1.54 or simply
ValidFCServers=default,fc.mydomain.com if single server.

- **WebSocketEnabled** - This option should currently be set to 1 if running the CherryPy web server, and 0 if running an Apache web server.
- **LoggingLevel** - Current values are: DEBUG, INFO, WARN, ERROR, CRITICAL, FATAL
fcws.cfg sample:

```
FCSPort=510
ValidFCServers=default,fc.mydomain.com

WebSocketEnabled=1
```

2. The 'startfcws.sh' file located in the '/opt/FirstClass Web Services' folder.

This file is currently in place to start the installed web server with the proper configuration.

- With the default installed file, it should start a web server on port 80. If you add SSL certificates as mentioned below, it will start a HTTPS server on port 443.
- By default, it tries to start a HTTPS server on port 443 unless no SSL certificate files are found. In such case, it falls back to starting a HTTP server on port 80.
- You can specify as it starts a HTTP server in addition to the HTTPS server by adding the '-e' as an arg.
- In the current release, it is configured for starting the CherryPy server through passing parameters to the cherry.pyc file in the FCSS folder.

The startup parameters currently supported are as follows:

Usage: cherry.pyc [options]

Options:

-h, --help

show this help message and
exit

-H HOST, --host=HOST

Specify the host name for the CherryPy Server. 0.0.0.0 will serve on all
interfaces. (default: 0.0.0.0)

-f FCS_PORT, --fcserverport=FCS_PORT

Specify the FCP port number used to connect to the FirstClass Server.
(default: 510)

-s SSL_PORT, --sslport=SSL_PORT

Specify the SSL / HTTPS port number. (default: 443)

-k SSL_PRIVATE_KEY, --key=SSL_PRIVATE_KEY

Specify the SSL key file name or path. (default: key.pem)

-c SSL_CERTIFICATE, --cert=SSL_CERTIFICATE

Specify the SSL cert file name or path. (default: cert.pem)

-i SSL_CERTIFICATE_CHAIN, --intermediate=SSL_CERTIFICATE_CHAIN

Specify the SSL intermediate certificate file name or path. (default: none)

-t SSL_THREAD_POOL, --sslthreads=SSL_THREAD_POOL

Specify the number of threads for the SSL CherryPy Server. (default: 500)

-e, --enablehttp

Enable the HTTP server. (default: False)

-p PORT, --httpport=PORT

Specify the HTTP port number. (default: 80)

--httpthreads=HTTP_THREAD_POOL

Specify the number of threads for the HTTP CherryPy Server. (default: 100)

(ie.) Modifying the startup line in the 'startfcws.sh' file to either of the following would cause the web server to start up both a HTTPS server on port 8443 and a HTTP server on port 8000.

```
/usr/bin/python3.7 "/opt/FirstClass Web Services/fcws-16.2.2.102Nlinux/fcws/cherry.pyc" -H 192.168.1.111 -s 8443 -ep 8000
```

```
/usr/bin/python3.7 "/opt/FirstClass Web Services/fcws-16.2.2.102Nlinux/fcws/cherry.pyc" --host=192.168.1.111 --sslport=8443 --enablehttp -httpport=8000
```

Starting and connecting to OpenText™ FirstClass® Web Services:

- For starting and stopping FirstClass Web Services, you will find a 'fcwsctl' control script under /usr/sbin.
- For example, you can start the FirstClass Web Services with '/usr/sbin/fcwsctl start' and stop the FirstClass Web Services with '/usr/sbin/fcwsctl stop'. Check status with '/usr/sbin/fcwsctl status'.
- Connecting to your FCWS server.
 - Open a web browser and type in the URL, which points to the computer on which you installed FirstClass Web Services.
 - Example: <http://www.mydomain.com> or if you configured it to start on an alternate port; <http://www.mydomain.com:8080>
 - If all goes well, a sign in screen will appear.

6.2 Upgrade notes

Follow the instruction, before upgrading:

- Ensure that you have a properly installed and configured FirstClass Server 16.0 Build 2540 or later before proceeding.

Upgrade Installation Overview

Follow the [Installation Notes](#) to upgrade your existing installation.

7 Patches

A *patch* is a piece of software that is designed to fix or improve a computer program or its supporting data. These may include repairs to security vulnerabilities or resolution of bugs and may also improve usability or performance. On OpenText My Support you will find two general types of patches.

Hotfixes are also known as quick-fixes or bug fixes. *Updates* are also known as service packs or service releases.

OpenText recommends administrators to check the 'FC Update Service' container on their FirstClass server or the OpenText Knowledge Center (<https://knowledge.opentext.com/>) for any patches or documentation updates that may have been posted after this release.

Please follow the [Installation Notes](#) above to install any patches, including both hotfixes and updates.

7.1 Hotfixes

Hotfixes are small patches that address software issues. Typically, there is no new functionality in a hotfix.

All hotfixes to OpenText™ FirstClass® Web Services will be cumulative and will be delivered via the FirstClass Update server.

7.1.1 Available hotfixes

The following table lists and describes the hotfixes available for this release.

Hotfix name	Additional information	Downloads
	No Hotfix is available for this version	All hotfixes are delivered through the FirstClass Updates server.

7.2 Updates

Updates consist of a number of fixes combined into a single patch. An update may also include new features proactively introduced into the product. For OpenText™ FirstClass® Web Services, updates are cumulative.

7.2.1 Available updates

The following table lists and describes the hotfixes available for this release.

Update name	Additional information	Downloads
FirstClass Web Services Build 16.2.3	Available for Web Services 16.2 SP3	All updates are delivered through the FirstClass Updates server.

8 Fixed issues

This section provides information about past issues that have been fixed in this release.

Issue number	Issue description
FC-4055	Updated CKEditor to 4.27.0
FC- 4041	Further XSS security fixes
FC-3782	Menus in the right pane, open on top of another window.
FC-4033	CKEditor WYSIWYG editor - CVE-2023-4771 (+1)

9 Known issues

The following known issues exist in this release.

Issue number	Issue description
FC-2045	Filter/Search - Issues in FCGO
FC-2344	Blank messages on open
FC-2498	Conference is showing no content until the window is resized
FC-3615	FCWS 16.1.0.019 - "My People" blank
FC-3672	New communities always called "New Conference"
FC-3673	Community invite for external user opens a blank page
FC-3743	New Communities in FCWS not sending email notifications when changes happen
FC-3775	Message list graphical glitch when clicking the left and right docking arrows
FC-3781	Pulse comments get a 400 error and cannot be posted
FC-3795	User Preference restrictions not applied through FCWS

FC-3839	Some conferences with folders and messages display as icons in FCWS
FC-3860	Error 1053 when starting FCWS Windows Service
FC-3864	Cannot leave comments in Task Manager container if it is not listed in the Directory

10 Contact information

OpenText Corporation
275 Frank Tompa Drive
Waterloo, Ontario
Canada, N2L 0A1

For more information, visit the [OpenText](#) or [My Support](#) websites.

© 2024 Open Text

Patents may cover this product, see <https://www.opentext.com/patents>.

Disclaimer

No Warranties and Limitation of Liability

Every effort has been made to ensure the accuracy of the features and techniques presented in this publication. However, Open Text Corporation and its affiliates accept no responsibility and offer no warranty whether expressed or implied, for the accuracy of this publication.