Moving and Upgrading:

If you are also going to move the install at the same time that you upgrade see the article "Moving to Another Computer" in the 3.11 Robo-FTP Help.

Upgrading From Version 3.11.x:

If you are upgrading to the current release of 3.11 from a previous version of 3.11 then the new version will simply replace the old version and automatically inherit all of your configuration settings. You do not need to reactivate your license or modify any of your command scripts, scheduled tasks or services. Minor revisions do not support the option of side-by-side installation.

1. Log in as Administrator on the computer where the previous version of Robo-FTP 3.10 is installed.
2. Download the current release of the Robo-FTP 3.11 installer.
3. Close any running instances of Robo-FTP 3.10, including any Robo-FTP services you may have installed. Also close any related programs such as the Script Editor, the Configurator, or the Robo-FTP Help File. Note: The Robo-FTP Scheduler Service does not need to be stopped as the installer will stop it and restart it after the upgrade completes.
4. Double-click to launch the Robo-FTP 3.11 installer.
5. Wait for the install to complete.

Upgrading From Version 3.10.x or Earlier:

If your existing environment is particularly old or complex we suggest care be taken in the upgrade process to carefully test each piece in the new version, allowing any problems that might arise to be detected and fixed by Support. We offer Professional Services at a reasonable hourly rate if you want help with this process.

We strongly recommend that you install Robo-FTP 3.11 in a new folder rather than overwriting your previous installation so that you can do the side by side testing. Overwriting your previous installation is risky because you can't roll back if you experience problems in the new version. Please consult the help file article Moving to Another Computer for the procedure to move Robo-FTP 3.11 to a new computer.

Procedure for Side-by-Side Installation:

To install Robo-FTP 3.11 alongside your previous version: (This is the supported installation method.)

1. Log in as Administrator on the computer where Robo-FTP is installed.
2. Download the Robo-FTP 3.11 Full Installer exe package.
3. Double-click to launch Robo-FTP 3.11 Full Installer.
4. Check the Accept License box then click the install button.
5. Launch the program named "*Import Settings*" from the Start Menu
6. select the row for the latest version of Robo-FTP that you actually used

7. click the "Import" button.
8. Click the "Close" button after the settings are imported.
9. Run the Configurator to verify that your settings were imported successfully.

Procedure for Overwriting a Previous Installation:
There are almost no circumstances in which you should install Robo-FTP 3.11 into a folder where Robo-FTP 3.10 or earlier is installed. To install Robo-FTP 3.11 in the same folder as version 3.10 or earlier:
1. Perform a full system back-up so that you can format all of your hard drives and restore from scratch.
2. Log in as Administrator on the computer where Robo-FTP is installed.
3. Download the Robo-FTP 3.11 Full Installer exe package.
6. Close Robo-FTP and Remove any Robo-FTP services you may have installed.
7. Close any related programs such as the Script Editor, the Configurator, or the Robo-FTP Help File.
8. Open the Windows Control Panel and uninstall your current version of Robo-FTP.
9. Double-click to launch Robo-FTP 3.11 Full Installer.
10. Check the Accept license checkbox
11. Click the Options button to change the installation folder
12. Then Click the install button and wait for the installer to complete.
13. Run the configurator from the start menu. If it asks Current User or System, select System.
14. The first time it is run the settings importer will come up.
15. Select the row for the latest version of Robo-FTP that you actually used and click the import button.
16. Once the configurator finishes importing and starts select the license tab on the right
17. Click Activate License. It should find the old version's active license and offer to move the license to this new version.

Completing the Upgrade

User Configuration Settings:
   If a non-administrative user account (or service account) had separate configuration settings in your previous version of Robo-FTP and that version is 3.9 or lower, you must import those settings by logging in to Windows with that account and running the *Import Settings* program from the Robo-FTP 3.11 folder in the Start Menu.

Migrating Framework Jobs:
   Framework jobs can be migrated from a previous version by running the Settings Importer. The jobs will be imported with their schedules disabled. To finalize the migration, each job's schedule will need to be disabled in the old version and then enabled in the new version. This can be done using the enable/disable button on the schedule page for a given job in the Framework web UI.
   By default, the Settings Importer imports jobs into the default SQL Server Compact Edition

database. This is OK for testing purposes but for production use we recommend using an Oracle or SQL Server database. To that end we recommend the following steps.

1. Create a new database for the new version to use as a backend.
2. Configure the new version to use the new database from the Scheduler Settings page in the Configurator.
3. Import the settings from the old version again.
4. Proceed with enabling/disabling jobs as described above.

Starting Robo-FTP:

← If you are running Robo-FTP as a Windows service created using the *Service Installer*, please remove and recreate the service with the new version of the *Service Installer* that is included with version 3.11.

← If you are running Robo-FTP as a Windows service created using the *Enterprise Dashboard* please restart your service(s) from the 3.11 *Enterprise Dashboard*.

← The supported method of running Robo-FTP from the command line (batch file, ASP script, scheduler, etc.) is to add the installation directory to the PATH environment variable rather than hard-coding the path for Robo-FTP.exe. When you install a newer version and wish to start using it system-wide, you simply update the PATH environment variable. See the "Adding Robo-FTP to the Path" topic in the Help file for details.

Licensing:

Version 3.11 is installed with a 30 day evaluation period. You can transfer the license from your old version of Robo-FTP by opening the configurator and going to the license tab and clicking activate. If there is an already activated older version installed it will offer to move the license to the new version.

The old version of Robo-FTP may be uninstalled whenever you are ready but we recommend keeping it installed until you are sure that you do not need to roll back.

COM Automation:

An exception to the side-by-side installation applies if you have developed an external program that runs Robo-FTP via its OLE/COM automation interface. To change which version of Robo-FTP is created via the COM interface, start a Command Prompt as administrator and run REGSVR32 on the version of RoboFTPAutomation.dll that you wish to use. For example:

```
Regsvr32 "c:\Program Files\Robo-FTP 3.11\Serengeti.Robo-
             FTP.SxS\RoboFTPAutomation.dll"
```

Compatibility with Previous versions

Backward compatibility with previous versions of Robo-FTP is preserved whenever possible. However, there are cases where it is necessary to break with this compatibility in order to provide enhanced performance and new features. This section describes the changes you may need to take to make your scripts work with this version.

- ← Robo-FTP 3.11 stores trusted ssh site keys using sha256 instead of sha1 fingerprints. This means the settings importer will not be able to import the ssh site keys from earlier versions. This means that when running the settings importer you will need to click ok to save each of the remote sites ssh fingerprints during the post import test for each of your ssh sites.

- ← Robo-FTP includes 2 new reserved variable names %counter and %random. If you are using either of those variable names in your scripts you will need to rename them.

- ← The Enterprise Framework concat step was always inserting a carriage return line feed between each file it concatenated together. This has been fixed. If you need this behavior to continue you will need to put %crlf in the separator field for the concat step.

- ← The Enterprise Framework mount step no longer uses a drive letter. Just specify the UNC path, username, and password. Then in the later step where you need to access that UNC path specify the UNC path itself rather than the previously mounted drive letter.

Additional notes for users upgrading from version 3.9.x and earlier:

- ← Robo-FTP 3.10 has an entirely new SSL/SFTP/FTP/HTTP protocol stack. It has been tested extensively with a wide variety of servers and clients but there are many more than we could possibly find or test out in the wild. If you encounter compatibility issues with a client or server that worked in the previous version please contact Support and we will address the issue promptly.

- ← SENDFILE /resume now reports $ERROR_CMD_NOT_SUPPORTED when used with a server that does not support resuming.

- ← %db_COLNAME and %db[COLNAME] variables are now unset as soon as dbgetresults returns an error of any kind.

- ← HTTP no longer supports some apache listing formats. Instead it uses the DAV verb PROPFIND if available or else uses GET. If you need timestamps and sizes without DAV

enabled use the FTPLOGON /precisetime option.

← [FTP]DIFF and [FTP]SNAPSHOT now only reset differences for existing results matching their path argument.

← [FTP]GETFILE no longer allow the /newest or /oldest option. These options did not work correctly before. To use them use GETNEXTFILE or GETSITEFILE.

← GETFILE with no wildcard and without /subdirs always returns success if the file exists since without a wildcard you are clearly not trying to iterate over a collection of files.

Additional notes for users upgrading from version 3.8.x and earlier:

← In FTP and FTPS modes with passive mode operation Robo-FTP will attempt to open a data channel to the same IP address used for the control channel. This is very unlikely to affect any currently working connections. You can override this behavior by adding the **/restrictipaddr=false** option to the FTPLOGON command line.

← The FTPLOGON and FTPLIST commands now return specific numeric error codes so scripts that expect specific codes from these commands may need to be modified accordingly.

← The PGPENCRYPT and PGPDECRYPT commands don't allow the target file to be the same as the source.

← When connecting to an HTTP/HTTPS site Robo-FTP uses any Internet Explorer proxy settings by default.

← COM license activation and removal functions raise exceptions instead of returning an error code.

← SNAPSHOT and FTPSNAPSHOT no longer reset differences generated with the DIFF or FTPDIFF commands

← Continue reading the next section if you are upgrading from version 3.7.x or earlier.

Additional notes for users upgrading from version 3.7.x and earlier:

These are the changes you need to be aware of if you are upgrading from any version of 3.7 **or earlier**.

← On a fresh installation, the top-level Robo-FTP installation folder is no longer the default local working folder. This could cause problems if the logic in your command scripts assumes otherwise. You can change the default local working folder in the Configurator under the *Files* sub-menu or you could add the following command to the start of your scripts: `WORKINGDIR %installdir`

← Variables initially declared as input arguments of user-defined functions now go out of scope when the function returns.

← Many types of external files used by Robo-FTP which were previously written to the installation folder by default are now written to a new group of subfolders which are automatically included when you export a settings file. These new subfolders for shared files are created in the Robo-FTP installation folder under a new folder named **ProgramData** and user-specific files are created in an additional new folder **UserData**. This change will make future upgrades easier but you may need to modify existing scripts to specify the complete path of some files. For example, if an existing script expects to find a DIFF database file in the current script folder instead of its new default folder then you will need to either move the existing file to the new default location or change a command line like this:

```
DIFF "c:\inbound" "snapshot.sql"
```

To specify the full path of the database file like this:

```
SET dbPath = %currentscriptdir + "\snapshot.sql"
DIFF "c:\inbound" dbPath
```

The new subfolders are named according to their contents as follows:

• Cron – The schedule files used by the CRON command are created in this folder

• Debug - A new type of log file is written here

• Known SSH Sites – Contains copies of SSH keys from previously trusted SFTP servers

- Logs – This is the default location for the Script Log, Trace Log and Session Log files

- PGP Keys – Keyrings containing PGP keys created by or imported into the Configurator

- Scripts – This is the new default location for your Robo-FTP command scripts and the new default working directory on a fresh installation

- Snapshots – SQLite databases used with DIFF, GETDIFF, FTPDIFF and FTPGETDIFF commands

- SQLite – SQLite databases used with the DBUSE command

- SSH Keys – Client Keys for SFTP authentication are stored in this folder

- SSL Certificates – Client Certificates for FTPS authentication are stored in this folder

- Trusted Certificates – Contains copies of SSL Certificates from previously trusted FTPS servers

New search logic applies when the following commands are called without a fully qualified file name: CALL, CHAIN, CRON, DBUSE, DIFF, GETDIFF, FTPDIFF, FTPGETDIFF, LOG, SESSIONLOG and TRACELOG. In previous versions, these commands searched the current working folder and/or the Robo-FTP installation folder. Please see the Help file topic entitled "**Alternate Default File Paths for Select Commands**" for specifics.

1. ←   When commands that either use an existing file or create a new file are called without a fully qualified file name and there is no existing file, the new file is created in the appropriate **ProgramData** or **UserData** subfolder depending on the file permissions of the current user. This applies to the following commands: CRON, DBUSE, DIFF, FTPDIFF, LOG, SESSIONLOG, and TRACELOG.

2. ←   The /skip option of the SYNC command now affects any file that is not the same on both the local and remote system. In previous versions of Robo-FTP this command

only affected orphan files.

3. ←   Newer versions of Windows no longer support allowing a service to interact with the desktop so this option was removed from Dashboard and the Service Installer.

4. ←   The READFILE command now considers the end of file marker as a valid record terminator. This makes it easier to process data files when the last record lacks a normal CR/LF terminator sequence. Existing scripts that rely on an error condition when a custom terminating sequence is not found before the end of a file must be modified.

5. ←   The PGPENCRYPT command is now able to use multiple public keys when encrypt a single file. In previous versions of Robo-FTP it might have been necessary to use a combination of the /user, /comment and /email options to uniquely identify a single key if there were other similar keys in the key ring file. If executed in the new version, scripts using this syntax may encrypt a file with multiple keys. To avoid this, modify the script to pass the entire key identification string to the /user option. This string is shown in the **Key Name** column on the *Manage Keys* form which can be opened from the **Manage PGP** menu in the Configurator. The string format is: `user name (comment) <email address>`

6. ←   The APPEND command no longer changes line endings from LF to CR+LF.

7. ←   A variable assignment using a command line argument that was not passed now results in a $ERROR_VAR_NOT_FOUND response code, for example: `SET var = %1`

8. ←   The default transfer method was switched from active to passive. To continue using active mode transfers add /pasv=false to your FTPLOGON command line.

9. ←   Continue reading the next section if you are upgrading from version 3.6.x or earlier.

Additional notes for users upgrading from version 3.6.x and earlier:

These are the changes you need to be aware of if you are upgrading from any version of 3.6 **or earlier**.

←   FTP transfers now default to Passive Mode. If necessary, you can disable Passive Mode in your script by adding the */pasv=false* option to your FTPLOGON command. You may also change the default behavior for a specific managed site by

editing that site's properties in the Configurator.

←   The output format for the FTPLIST command has been changed. The new format provides more detail from some servers than the previous format.

←   The SSL certificate trust store in the new version of Robo-FTP is stricter for improved security. This change will require you to re-trust server SSL certificates using the *trust=all* option on the FTPLOGON command. This change applies to FTPS and HTTPS connections.

←   ALTFTP (Alternate Standard FTP) has been promoted to being the main FTP implementation. This is an automatic change requiring no action on the part of the user.

←   FTPS connections now default to encrypting both the data and control channel. This was changed to prevent people unintentionally sending data across the internet in plain text. If the remote site you are using doesn't support an encrypted data channel you will need to modify your script so that the FTOPLOGON line specifies the */servertype=FTPSCONTROL* option.

←   DISCONNECT now only hangs up a dial-Up connection previously established using the DIALNET command. The DISCONNECT script command no longer terminates an active FTP session. The FTPLOGOFF script command should be used to terminate FTP sessions.

←   The script commands IFSTRCMP and IFSTRCMPI now function as described in the Help file. The behavior of these two commands was mistakenly swapped in previous versions of Robo-FTP. This change only impacts users who intentionally modified their scripts to account for the mistake in previous versions.

←   The Windows 2000 operating system is no longer supported by Robo-FTP.

←   Version 3.6 and earlier did not support importing multiple SSH keys or SSL certificates for client authentication. The workaround for this limitation was to export multiple settings files and then launch Robo-FTP using the -g command line switch to import the settings file required for each site. This limitation was removed in Robo-FTP 3.7. If you previously used this work-around you must import your client keys and certificates into the Configurator and then associate each of them with the Managed Site record for each server.

←   If you want the public key for an SSH key created in Robo-FTP, right click on the key in the configurator and choose export public key. This will export the key in the format that 3.6 output in the .openssh file. The other formats have become very rare and are no longer supported.

←   Continue reading the next section if you are upgrading from version 3.5.x or earlier.

Additional notes for PGP users upgrading from Version 3.5.x and earlier:

In version 3.6 Robo-FTP began using a new PGP implementation based on OpenPGP that features standards-compliant encryption and decryption. This allows multiple private keys and their passphrases to be stored without the need for multiple key ring files. The new implementation has been made backward compatible as much as possible, so most scripts should "just work." However, there are a few important differences for users upgrading from Robo-FTP 3.5 or older:

Specifying key ring file location with PGPENCRYPT and PGPDECRYPT script command
- This has been deprecated and removed. If the location you are specifying is the default location (usually the installation directory) then you can simply remove this argument from the command line. If your script specifies a different location, you should import that key ring in the Robo-FTP Configurator and then remove this argument from the command line.

Signature verification
- Signature verification options are now managed using the /signature option with the **PGPDECRYPT** command. Change references like this: */gpgopt="—skip-verify"* to this */signature=ignore*

Using /gpgopt or /gpglog options
- These options are no longer used and will be ignored by Robo-FTP. Contact Technical Support if you are using these options and have questions about how this change will affect your script.

Compatibility with old versions of PGP Desktop
- Recipients using very old versions of PGP Desktop may not be able to decrypt files encrypted with Robo-FTP. If you encounter this problem use the */compat* option with **PGPENCRYPT** to create PGP Desktop 2.x compatible files (2 GB file size limit).

Signing files
- If you are using more than one secret key for singing purposes by using multiple key ring files you should import those keyrings into the Robo-FTP Configurator and use specify which key to use with the */signwith option* (see Help File for details).