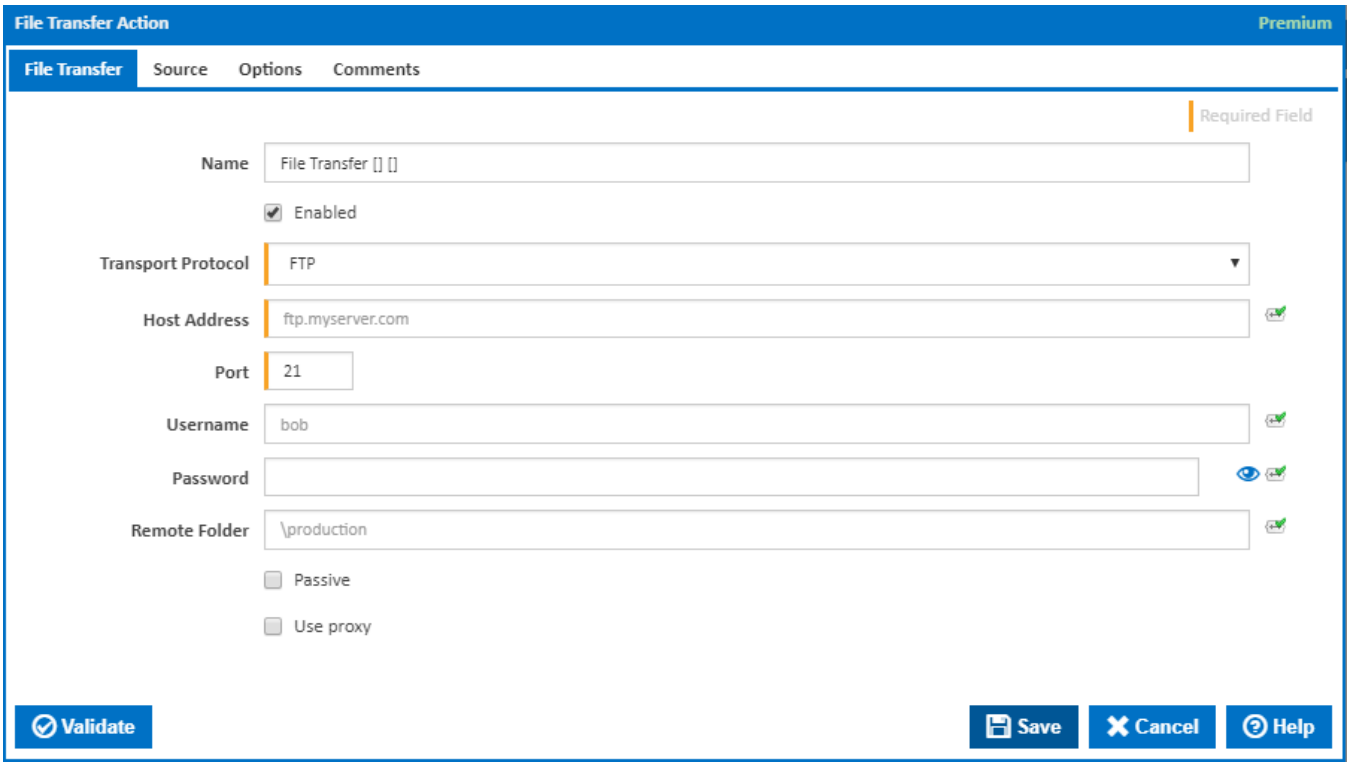
File Transfer Action

The File Transfer action is a premium action that can be used to transfer packages or application files to a remote server. The action allows transfer via four protocols: FTP, FTPS implicit, FTPS explicit and SFTP.

File Transfer



Name

A friendly name for this action (will be displayed in the actions workflow area).

Enabled

Determines if this action will be run within the relevant stage.

Transport Protocol

Selected the transfer protocol to use:

- FTP: This allows files to be transferred using a connection without encryption. Authentication is provided with a clear-text username and password, or anonymous login (without providing a password) if the server is configured to allow it.
- FTPS Implicit - This allows files to be transferred using a TLS/SSL secured connection to the host. TLS/SSL encryption is switched on **implicitly** a s soon as the channel is established.
- FTPS Explicit - This allows files to be transferred using a TLS/SSL secured connection to the host. The client **explicitly** requests TLS/SSL encryption to be switched on.
- SFTP: This files to be transferred to a SSH host.

Host Address

Enter the address of the host to transfer the files to. This can either be a DNS name, URL or IP4 address.

Port

The port that the host is listening on for the selected protocol. Standard ports for each protocol are:

- FTP: 21
- FTPS Explicit: 21
- FTPS Implicit: 990
- SFTP: 22

Some servers do use non-standard ports for a variety of reasons, please check with your provider.

Username

The name of the user for logging into the host machine. This can be blank as some file transfer services do not require a username or password to be provided. If a username is required ,you will receive an error from the action stating that a login is required by the host.

Password

The matching password for the supplied username. This can be blank as some file transfer services do not require a username or password to be provided. If a password is required, you will receive an error from the action stating that a login is required by the host.

Note that if the password contains a $ or % they will need to be escaped, e.g. $$ and %%, as these are identifiers for objects and variables in Continua CI.

Remote Folder

The folder on the remote machine to upload the files to. Note that this is relative to the base directory for the user on the host. Note that the source folder structure will be preserved when transferring files to the host.

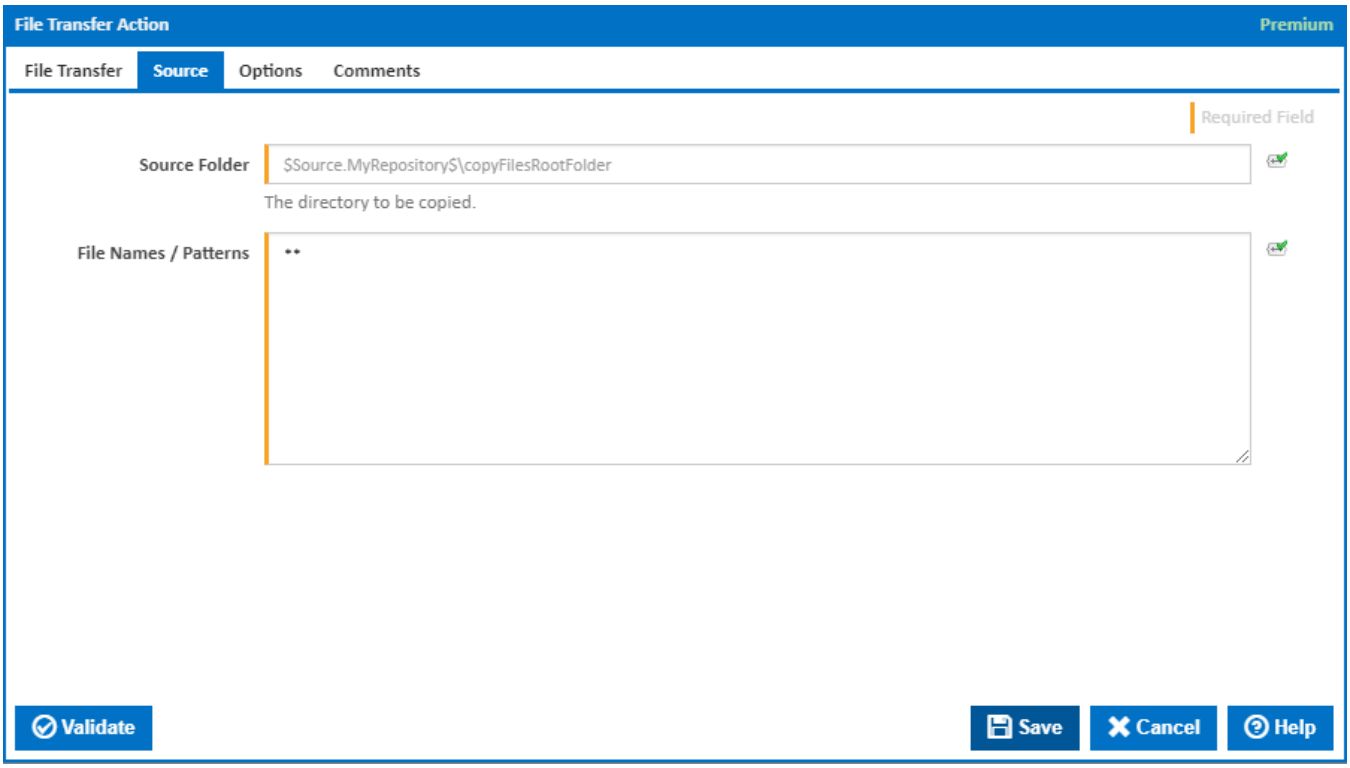
Passive

Tick to enable the passive file transfer protocol.

Use Proxy

Tick to make the 'Proxy' tab visible.

Source



Source Folder

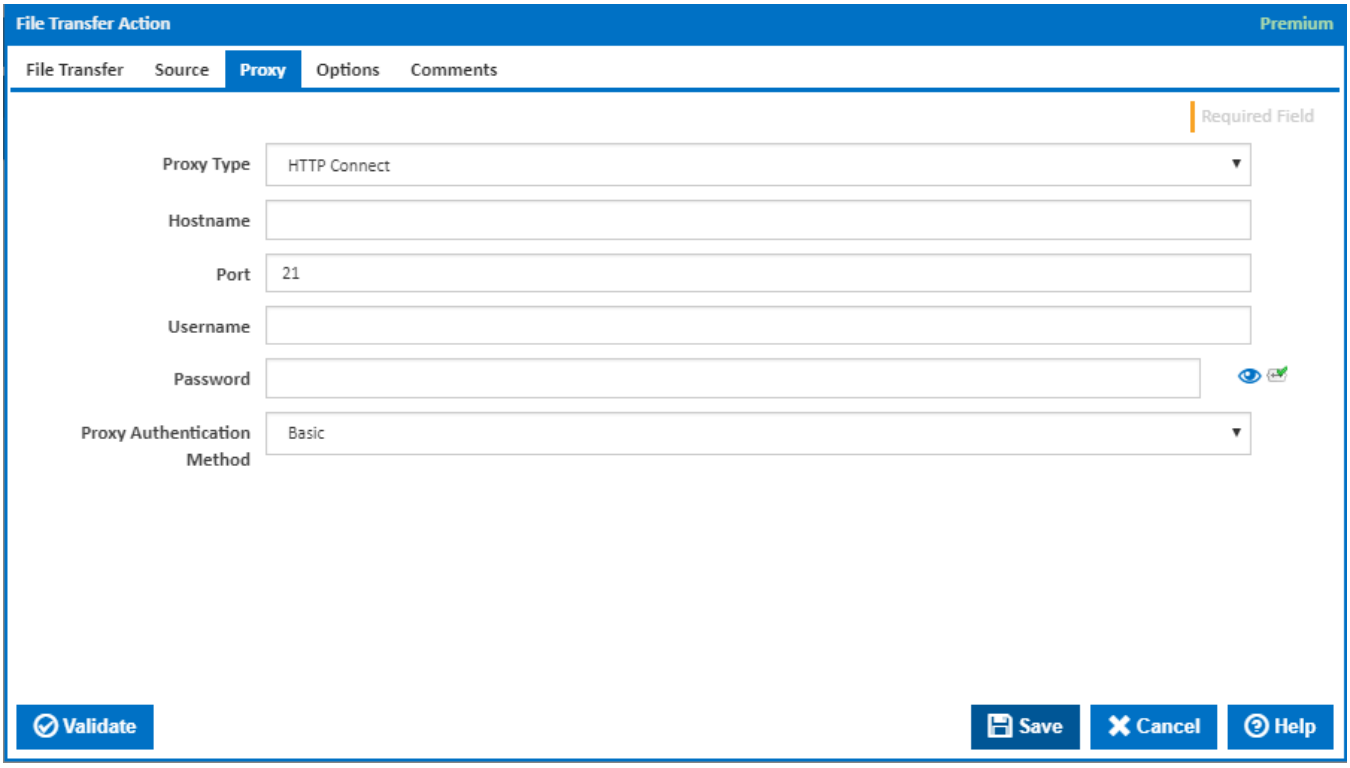
Enter the folder containing all the files to be uploaded to the host folder. This is typically a workspace location such as "$Workspace$\Output\".

File Names / Patterns

Enter a list of files or file patterns to match the file to be uploaded. Each file/pattern must be entered on a new line. Patterns can contain wildcards as described on the Ant Pattern Usage page.

You can exclude files by prefixing the file name or pattern with a dash. e.g -*.ignore. Exclude patterns always take precedence over include patterns.

Proxy



Proxy Type

The proxy types handled by the File Transfer action are:

- SOCKS4
- SOCKS4a
- SOCKS5
- HTTPConnect

The type of proxy connection will be determined by the proxy server in use. Please talk to your network administrator as to which proxy server your company uses.

Host Name

Enter the name or address of the host to proxy the file transfer connection through. This can either be a DNS name, URL or IP4 address.

Port

Enter the port to use to connect to the proxy. This will depend on the configuration of the proxy server in question. Talk to your network administrator or proxy provider as to what settings are required for your particular file transfer.

Username

Enter the username required to access the proxy. Some proxies require authentication before you can tunnel your connection through them. If the username is left blank, then no username or password is used for connecting to the proxy.

Password

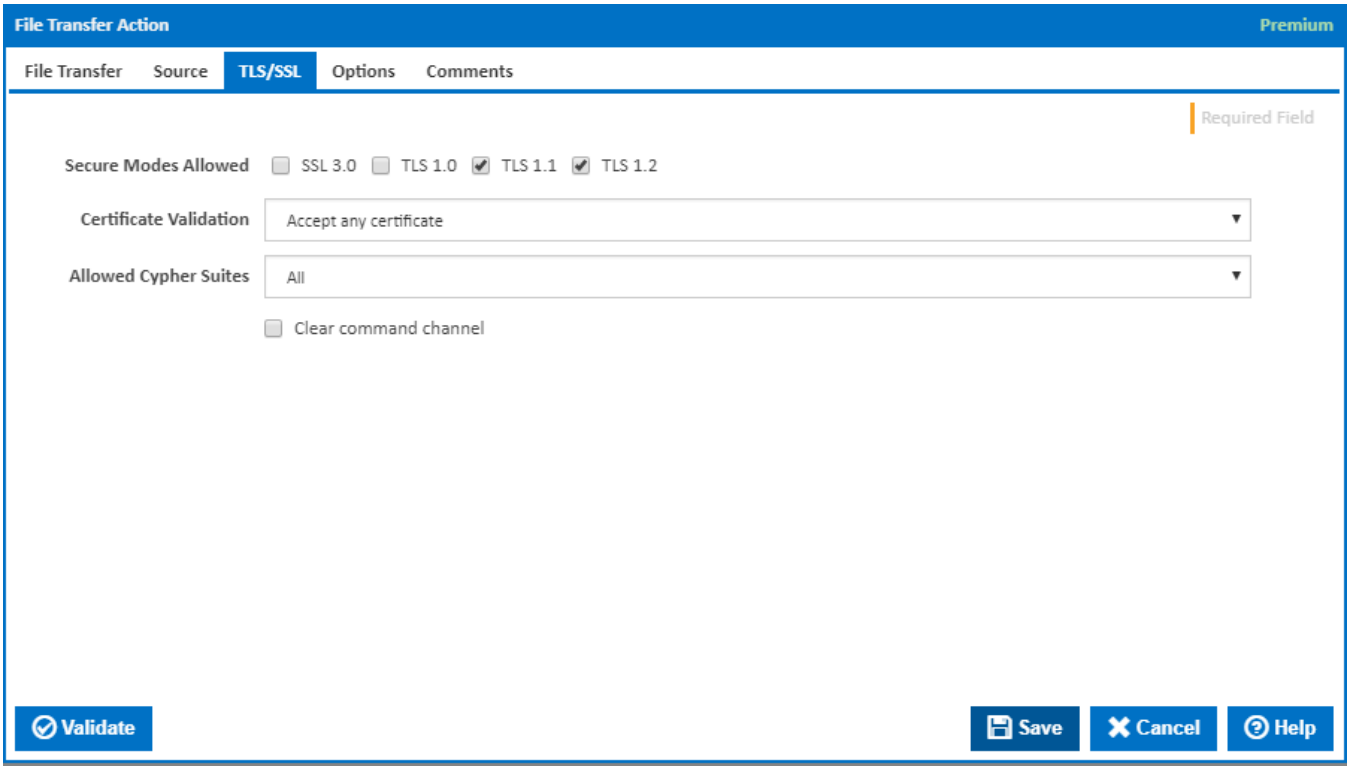
Enter the password required to access the proxy. This password should match the username provided for the proxy. The password is encrypted on the system, and once entered will not be readable when editing the action.

Proxy Authentication Method

The method of authentication required by the proxy server. The following proxy authentication settings are available:

- Basic
- NTLM
- Digest

TLS/SSL



Secure Modes Allowed

Choose one or more of the following TLS protocols to be accepted when connecting to the host:

- SSL 3.0
- TLS 1.0
- TLS 1.1
- TLS 1.2

Note that these options are only available when a TLS based file transfer protocol is selected, namely the FTPS protocols. Please check with your host FTPS provider as to which TLS protocols to use. Depending on the configuration of the host, certain protocols might not be allowed, or would provide better security for the connection. The action will attempt to select the highest security protocol as a default and log which one was used.

Certificate Validation

The file transfer action allows the following certificate verification methods.

- Accept any certificate: This option will accept any certificate the server presents to the connection. This might be used if the certificate is unknown, self-certified or can not be validated on the agent machine.
- Use windows infrastructure: This option uses the windows certificate store to validate the certificate presented by the host. If the host certificate chain is not trusted by the windows key store on the agent machine then the connection will be rejected.
- Locally stored thumbprint: This is typically the simplest to setup. A thumbprint is entered and used to validate the hosts certificate. To obtain a thumbprint, ask your host provider to generate a thumbprint of their certificate, or use an ftp client which presents it to you on connection.
- Reject certificates: This is typically not used, but included to allow for user testing of rejecting certificates. This option will simply reject any certificate supplied and terminate the connection. This will, in turn, fail the action.

Thumbprint (SHA1 hash)

This box is shown when "Locally stored thumbprint" is selected for the Certification Validation. Enter the thumbprint of the server certificate to accept.

Allowed Cypher Suites

Select the type of cipher suites allow when connecting. The options available are:

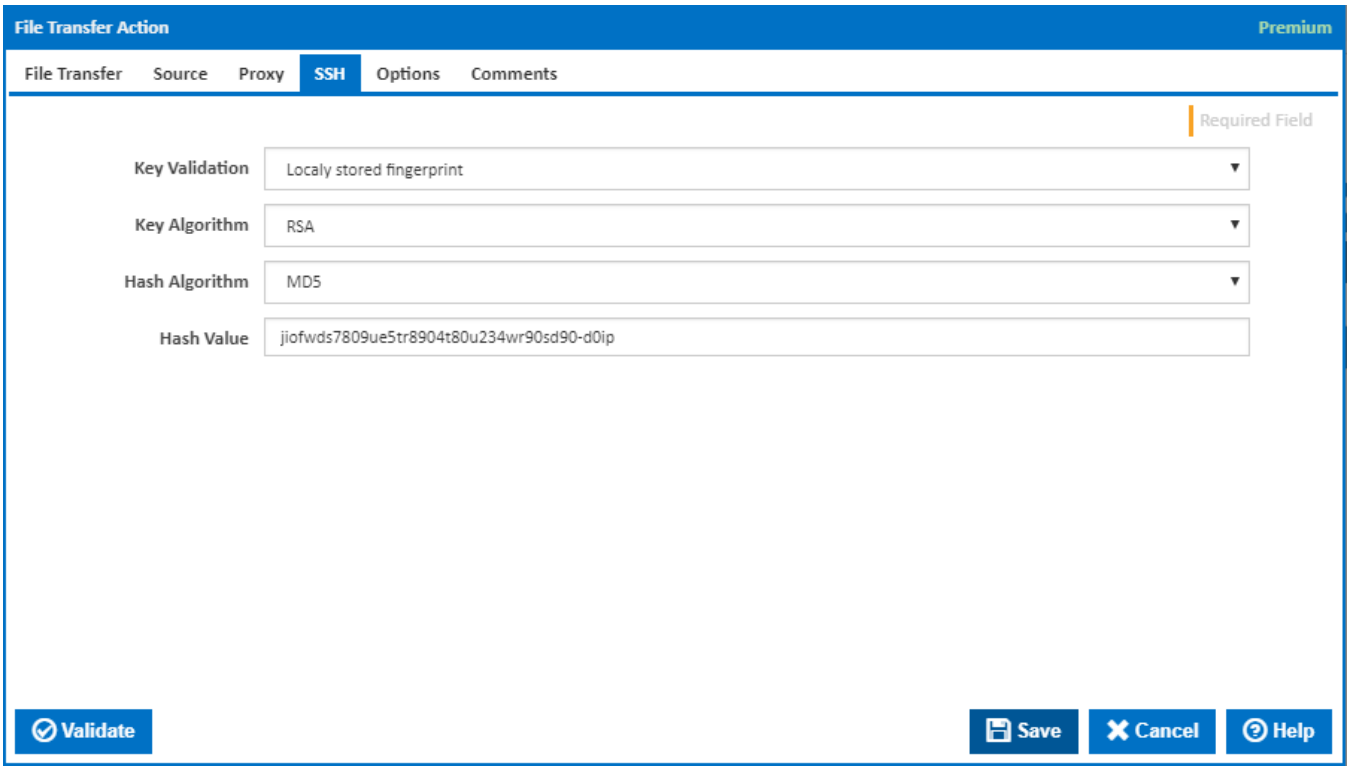
- None - Do not allow any cipher suites to be used in securing the connection. Typically a server requires that a cipher suite is chosen (as the connection would not be secured) and will drop the connection.
- Anonymous - This allows any of the anonymous ciphers to be used to secure the connection. For example, "Anonymous DES in CBC mode with SHA-1 hash" is one cipher which fits into this category.
- Secure - This allows any of the cipher suites that are considered secure to be used. This is the preferred option for securing the connection.

- All - This option allows for all cipher suites known to the file transfer action to be used. This option can be chosen if the cipher suite to use is unknown, or a broader set of suites is required that isn't covered by one of the above.

Clear Command Channel

Tick this to clear the command channel on the secure connection after login. Some servers require this to function correctly. If your experiencing any login errors where the server is reporting unexpected commands, please turn this option on.

SSH



Key Validation

The file transfer action provides the following ways to validate the SSH key of the host:

- Accept any key: This option will accept any key supplied by the server. This might be used if the key is unknown and storing it in the action is not possible.
- Locally stored fingerprint: This option uses a locally stored fingerprint to identify the key. This method requires access to the hosts SSH key algorithm, hash and algorithm used to create the hash.
- Reject key: This option will reject any key supplied by the server. Typically this is used in testing whether servers keys can be rejected and is not used in production systems.

Key Algorithm

This option specifies the algorithm that should be used to read the key. This will be known by the administrator of the host.

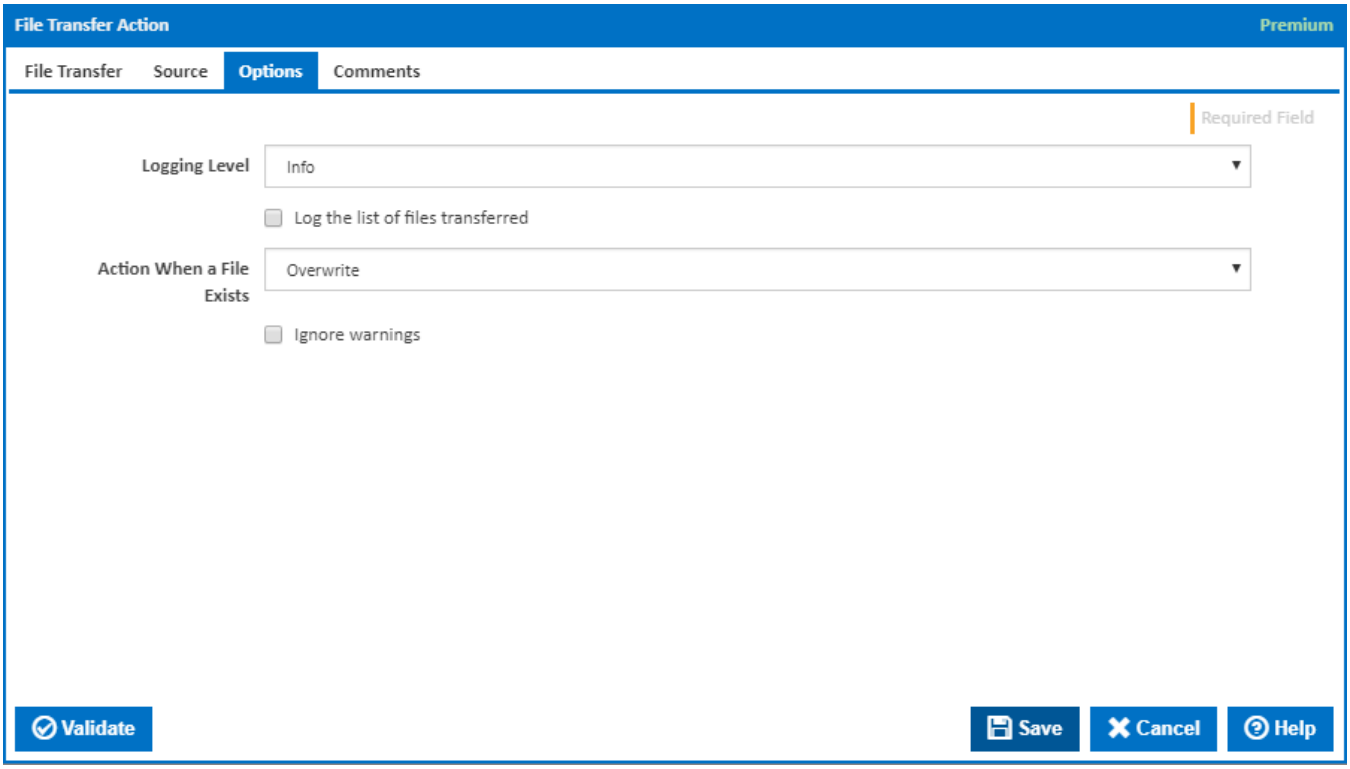
Hash Algorithm

This option specifies the hash algorithm used to generated a hash of the SSH key. If the algorithm used does not generate the hash listed below, the key will be rejected.

Hash Value

The hash of the SSH hosts key. This should be generated using the hash algorithm listed above.

Options



Logging Level

The amount of information detail to display in the build log.

Log the list of files transferred

Write the file name of each file transferred to the Build Log.

Action When a File Exists

Select the action to perform when a file already exists:

- Return as failure: An existing file causes the whole multi-file transfer to be cancelled and the action to fail.
- Skip: All existing files are skipped.
- Overwrite: Existing files are always overwritten.
- Overwrite if file is reported as older: Existing files are overwritten if they are older than source files. Otherwise they are skipped. This is strongly discouraged because modification dates are often misreported by FTP and SFTP servers, making this mode highly unreliable. We strongly recommend that you use a different mode.
- Overwrite if file has different size: Existing files with different sizes are overwritten. Otherwise they are skipped.
- Attempt to resume if file is smaller: Existing files are resumed if they are smaller than source files. Otherwise they are skipped.
- Rename with number: Existing files are renamed according the pattern "filename[number].extension".
- Overwrite if file has different checksum: Existing files with different checksum are overwritten. FTP uses checksums provided by the server (some FTP servers support no checksums).

Ignore warnings

If this is ticked, any warnings logged will not mark the action with a warning status.