

# Deployer plugin



The plugin is bundled since TeamCity 10.0. The recent version is [described](#) in the online documentation for the latest TeamCity version.

## General Info

Vendor	Nikita Skvortsov
License	<a href="#">Apache 2.0</a>
Type	open-source
Status	beta

## Download

Last stand-alone version: [deploy-runner.zip](#)

## Description

Enables TeamCity to upload artifacts to external locations in a number of ways.

## Installation

Follow the TeamCity documentation on [installing additional plugins](#).

## Usage

After successful installation, new Build Runners will be available in the Build Step selector of a build configuration. Most of deployers use TeamCity artifact paths to define which files will be deployed. Artifacts paths have the following syntax:

```
file_name|directory_name|wildcard [ => target_directory|target_archive ]
```



Please note:

`target_directory` and `target_archive` are always treated as relative to a remote location specified in the Target parameter of the runner.

Refer to the [TeamCity documentation](#) for details.

### SMB Deployer

Allows uploading files to Windows shares via the SMB protocol. Implementation relies on the JCIFS library. To configure the JCIFS client using system properties, update [the TeamCity Agent startup properties Configuring Build Agent Startup Properties](#).

Target URL should point to a host + share at least. Subdirectories are allowed here and will be created if missing. Valid examples:

```
\\host\share_name  
\\host\share_name\some\path  
\\host\c$\Temp
```

Name resolution (advanced option) - allows to switch JCIFS to "DNS-only" mode. May fix performance or out of memory exceptions (see [this bitbucket issue](#) for details). Is equivalent to following JCIFS settings:

```
-Djcifs.resolveOrder=DNS
-Djcifs.smb.client.dfs.disabled=true
```

## FTP Deployer

Allows uploading files to an FTP server.

Target host should point to an FTP server (by hostname or IP address) and a remote directory (relative to the FTP user's home). To use absolute \*nix path, use %2F as the forward slash. For example:

```
ftp://hostname.com/
hostname.com:34445/subdir
127.0.0.1/%2Fetc/
```

Secure connection allows choosing between an insecure (FTP) or secure connection (FTPS, SFTP).



### Limited security note

The current secure connection implementation accepts any certificate provided by a remote host. No trust checks are performed!

Authentication method is either Anonymous (will submit username "anonymous" and single space as password) or username/password (for custom credentials)

Transfer Mode allows you to force the ASCII or Binary FTP transfer modes (if the automatically detected mode leads to broken files transfer)

## SSH Deployer

Allows uploading files via SSH (using SCP or SFTP protocols)

Target should point to an SSH server location. The syntax is similar to the one used by the \*nix scp command:

```
{hostname|IP_address}[:target_dir[/sub_path]]
```

where target\_dir can be absolute or relative; sub\_path can have any depth.

Transport allows choosing a protocol to transfer data over SSH. Options are: SCP and SFTP

Authentication method allows choosing an SSH authentication method.

- "Default private key" will try to perform private key authentication using the `~/.ssh/config` settings. If no settings file exists, will try to use the `~/.ssh/rsa_pub` public key file. No passphrases should be set.
- "Custom private key" will try to perform private key authentication using the given public key file with given passphrase
- "Password" - simple password authentication.
- "SSH-Agent" - use ssh-agent for authentication. To be used in combination with TeamCity [SSH-Agent build feature](#). Available since 9.1



### Limited security note

Current secure connection implementation accepts any certificate provided by remote host. No trust checks are performed!

## SSH Exec

Allows executing arbitrary remote commands using SSH

Target should point to an SSH server.

Authentication method allows choosing an SSH authentication method.

- "Default private key" - will try to perform private key authentication using the `~/.ssh/config` settings. If no settings

- file exists, will try to use the `~/.ssh/rsa_pub` public key file. No passphrases should be set.
- "Custom private key" will try to perform private key authentication using a given public key file with a given passphrase
- "Password" - simple password authentication.
- "SSH-Agent" - use ssh-agent for authentication. To be used in combination with TeamCity [SSH-Agent build feature](#). Available since 9.1



#### Limited security note

Current secure connection implementation accepts any certificate provided by remote host. No trust checks are performed!

Commands is a new-line delimited set of commands that will be executed in the remote shell. The remote shell will be started in the home directory of an authenticated user. The shell output will be available in the TeamCity build log.

## Container Deployer

Allows deploying WAR application archives to a remote Tomcat instance  
Supports the following Tomcat versions: 5.x, 6.x, 7.x and 8.x

Requires Manager webapp installed in the target Tomcat instance and a user with the "manager-script" role.

## Compatibility

TeamCity 9.1+  
Oracle JDK 1.6 or OpenJDK 1.7.0\_07 and later.

## Sources

<https://github.com/JetBrains/deployer>

## Issue tracker

<https://github.com/JetBrains/deployer/issues>

## Continuous Integration

The current build status is  
[Detailed CI status page](#)

## Third party libraries

Name	Site	License
Cargo	<a href="#">Cargo library</a>	Apache 2.0
Commons-Net	<a href="#">Commons-Net</a>	Apache 2.0
JSch	<a href="#">JSch</a>	BSD-style
JCifs	<a href="#">JCifs</a>	LGPL 2.1