Applying Patches

Microsoft Visual Source Safe Integration

To apply a patch for **`vss-native.exe`**:

1. Shut down the TeamCity server.
2. Open the `<TeamCity Home>/webapps/root/WEB-INF/plugins/vss/` or `<TeamCity Home>/webapps/root/WEB-INF/lib/` folder.
3. Back up the `vss-support.jar` file.
4. Inside the `vss-support.jar` file, replace the `/bin/vss-native.exe` with the new one.
5. Start the server.

To apply a full VSS plugin patch:

1. Shut down the TeamCity server.
2. Open `<TeamCity Home>/webapps/root/WEB-INF/plugins/vss/` or `<TeamCity Home>/webapps/root/WEB-INF/lib/.`
3. Back up `vss-support.jar`.
4. Replace `vss-support.jar` with the new one.
5. Start the server.

Capturing Logs From VSS-native

Each time TeamCity starts, it creates a new instance of the `vss-native.exe` file and places it into the `<TeamCity Home>/temp` f older. The name of the copy is generated automatically and uses the following template: `TC-VSS-NATIVE-<some digits>.exe`

To manually enable detailed logging (for debugging purposes) for VSS Native:

1. Copy the `<TeamCity Home>/temp/TC-VSS-NATIVE-<some digits>.exe` file to any folder.
2. Run the program with the `/log` switch.

To get the commandline syntax and options reference, run the program without any switch.

Microsoft Team Foundation Server Integration

To apply a patch for **`tfs-native.exe`**:

1. Shutdown the TeamCity server
2. Open `<TeamCity Server>/webapps/root/WEB-INF/plugins/tfs/` or `<TeamCity Server>/webapps/root/WEB-INF/lib/.`
3. Backup `tfs-support.jar`.
4. Inside the `tfs-support.jar` file, replace `/bin/tfs-native.exe` with a new one.
5. Start the server.

To apply a full TFS plugin patch:

1. Shutdown the TeamCity server.
2. Open `<TeamCity Home>/webapps/root/WEB-INF/plugins/tfs/` or `<TeamCity Home>/webapps/root/WEB-INF/lib/.`
3. Back up `tfs-support.jar`.
4. Replace `tfs-support.jar` with a new one.
5. Start the server.

Capturing logs from TFS-native

To enable creating logs from TFS-native:

1. Locate `tfs-native.exe` under the TeamCity `temp` folder. The file name format is `TC-TFS-NATIVE-<digits>.exe`.
2. Create a copy of the file in any other folder.
3. Run this program with the `/log` switch.

To get the command-line switches help, run the process with no parameters.
Log files will be created in the `<TeamCity agent>/temp/buildTmp/TeamCity.NET` folder. For each process a new log file will be created.

.NET runners

To patch .NET part of .NET runners:

1. Open `<TeamCity Server>/webapps/ROOT/WEB-INF/plugins/dotNetRunners/agent`.
2. Copy `dotNetPlugin.zip` to a temporary folder.
3. Back up `dotNetPlugin.zip`.
4. Extract `dotNetPlugin.zip`.
5. Replace the contents of the `/bin` folder with new files.
6. Pack the files again. Make sure there are no files in the root of the archive.
7. Create the `<TeamCity Server>/webapps/ROOT/update/plugins` directory.
8. Put `dotNetPlugin.zip` file into `<TeamCity Server>/webapps/ROOT/update/plugins`. All build agents will upgrade automatically
9. Run builds.

To enable logging from .NET runners:

1. Open `<TeamCity Server>/webapps/ROOT/WEB-INF/plugins/dotNetRunners/agent`.
2. Copy `dotNetPlugin.zip` to a temporary folder.
3. Back up `dotNetPlugin.zip`.
4. Extract `dotNetPlugin.zip`.
5. Copy `/bin/teamcity-log4net-debug.xml` to `/bin/teamcity-log4net.xml`.
6. You may patch the Log4NET config file if you need.
7. Pack the files again. Make sure there are no files in the root of the plugin archive.
8. Create the `<TeamCity Server>/webapps/ROOT/update/plugins` directory.
9. Put `dotNetPlugin.zip` file into `<TeamCity Server>/webapps/ROOT/update/plugins`.All build agents will upgrade automatically.
10. Run builds.

By default, all the log files will be stored in the `<TeamCity agent>/temp/buildTmp/TeamCity.NET` folder. Log files are created for each process separately.