

Known Issues

This page contains a list of workarounds for known issues in TeamCity.

- Agent running as Windows Service Limitations
 - Security-related issues
 - Windows service limitations
 - Issues with automated GUI and browser testing
 - Running automated GUI tests and using RDP
 - Early start of the service before other resources are initialized
- 'Channel is not opened' error in git repository on bitbucket.org
- Clearing Browser Caches
- Logging with Log4J in Your Tests
- Agent Service Can Exit on User Logout under Windows x64
- Failed Build Can be Reported as a Successful One With Maven 2.0.7
- Conflicting Software
- Subversion issues
 - svn: E175002: Received fatal alert: bad_record_mac
 - Subversion-related JVM Crashes
- NUnit 2.4.6 Performance
- StarTeam Performance
- Perforce 2009.2 Performance on Windows
- Wrong times for build scheduled triggering (Timezone issues)
- Upgrading IntelliJ IDEA May Affect Active Pre-Tested Commits
- Other Java Applications Running on the Same Server
- The Server Does Not Start Claiming the Database is in Use
- Slow download from TeamCity server
- Failure to publish artifacts to server behind IIS reverse proxy
- SSL problems when connecting to HTTPS from TeamCity (handshake alert: unrecognized_name)

Agent running as Windows Service Limitations

When a TeamCity build agent is installed as a Windows service, there may appear various "Permission denied" or "Access denied" errors during the build process, see details below.

Security-related issues

The user account used by the service is required to have sufficient permissions to perform the build and [manage the service](#). If you run the TeamCity agent service under the SYSTEM account, do the following:

1. [Change SYSTEM](#) for a usual user account with necessary permissions granted.
2. Restart the service.

Windows service limitations

As a Windows service, the TeamCity agent and the build processes are not able to access network shares and mapped drives.

To overcome these restrictions, run TeamCity agent [via console](#).

Issues with automated GUI and browser testing

These problems include errors running tests headless, issues with the interaction of the TeamCity agent with the Windows desktop, etc.

To resolve/ avoid these:

1. Run TeamCity agent [via console](#).
2. Configure the build agent machine not to launch a screensaver locking the desktop.



Note that there is a Windows limitation to accessing a remote computer via mstsc: the desktop of the remote machine will be locked on RDP disconnect, which will cause issues running tests. The VNC protocol allows you to remote control another machine without locking it.

To run GUI tests and be able to use RDP, see the [workaround](#) below.

3. Configure the TeamCity agent to start [automatically](#) (e.g. configure an automatic user logon on Windows start and then configure the TeamCity agent start (via agent.bat start) on the user logon).
For graphical tests the build agent cannot be started as a service and it is recommended to configure the build agent launch with a 1 minute delay after the user auto-logon, e.g. using the "bin\agent.bat start" command in the task scheduler and configuring the delay there.

Running automated GUI tests and using RDP

RDP uses its own video driver overriding the one from the machine's video card for the session. Redirecting the session to console will unload the Windows graphical drivers. This can be done by adding the following step to your build configuration prior to your tests (the example below is for PowerShell, but other languages (DOS, Python) can be used too):

```
$sessionInfo=((quser $env:USERNAME | select -Skip 1) -split '\s+')  
if ($sessionInfo[1] -like "rdp-tcp*") { tscon $sessionInfo[2] /dest:console }
```

where "quser [current username]" lists all the connections to that machine for the user, either console or graphical. The one listed as rdp-tcp##* is the remote desktop connection which can be redirected to the console using "tscon [connection id] /dest:console".



An unsupervised computer with a running desktop permanently logged into a user session might be considered a network security threat, as access to it can be difficult to trace. Therefore, it is recommended to run automated GUI and browser tests on a virtual machine isolated from sensitive corporate network resources, e.g. on a machine not included in a Windows domain.

Early start of the service before other resources are initialized

To handle this, consider using the Automatic (Delayed Start) option of the service settings or configure [service dependencies](#).

For more investigation steps, see the [Common Problems](#) page.

[Back to top](#)

'Channel is not opened' error in git repository on bitbucket.org

In July 2016, Bitbucket Cloud has made changes to ssh protocol handling disabling multiplexing for ssh connections. As a result, Git checkout for VCS roots configured with ssh:// protocol in TeamCity fails with "channel is not opened." error (related TeamCity [issue](#)).

As a workaround, install an updated version of git-plugin corresponding to your TeamCity version: [9.1.x](#), [9.0.x](#) or [8.1.x](#) (or [10 EAP](#)) - use artifacts from the latest successful build. To install the plugin, put downloaded jetbrains.git.zip into <TeamCity Data Directory>/plugins and restart the TeamCity server.

Clearing Browser Caches

There is a web UI-related issue which some of our users have encountered (and it cannot be reproduced on other computers) which is tied with the cached versions of content. If you have come across such problem, make sure your browser does not use cached versions of content by [clearing browser caches](#).

[Back to top](#)

Logging with Log4J in Your Tests

If you use Log4J logging in your tests, in some cases you may miss Log4J output from your logs. In such cases please do the following:

- Use Log4J 1.2.12
- For Log4J 1.2.13+, add the "Follow=true" parameter for console appender, used in Log4J configuration:

```
<appender name="CONSOLE" class="org.apache.log4j.ConsoleAppender">
  <param name="Follow" value="true"/>
</appender>
```

[Back to top](#)

Agent Service Can Exit on User Logout under Windows x64

The used version of [Java Service Wrapper](#) does not fully support Windows 64 and this causes agent launcher process to be killed on user logout. The agent itself will be function until the next restart (server upgrade or agent properties change).

[Back to top](#)

Failed Build Can be Reported as a Successful One With Maven 2.0.7

This is a known bug in this version of Maven. Consider using any later version.
In case it's not possible you can patch mvn.bat yourself by replacing the fragment at line 148 of mvn.bat:

```
:error
set ERROR_CODE=1
```

with the following one:

```
:error
if "%OS%"=="Windows_NT" @endlocal
set ERROR_CODE=1
```

[Back to top](#)

Conflicting Software

Most common indicators of conflicting software are errors like "Access is denied", "Permission denied" or java.io.FileNotFoundException mentioning the file that is present and is writable by the user the agent/build runs under. Also, certain software running in background (like antiviruses) can significantly slow down build agent operations like sources checkout, artifact publishing or even build running.

Certain antivirus software like Kaspersky Internet Security can result in Java process crashes or other misbehavior like inability to access files. e.g. see [the issue](#).

ESET antivirus can also slow down Ant/IntelliJ IDEA project builds a great deal (slowing down TCP connections to localhost on agent).

If you run antivirus on the TeamCity server or agent machines and get disk access errors or experience degraded performance, please disable or better completely uninstall the antivirus software before investigating the issue and reporting the issue to JetBrains.

It is recommended to exclude entire TeamCity server home and [TeamCity Data Directory](#) from the background checks and perform periodical checks there in the well-known maintenance window so that those do not affect server performance much. On TeamCity agent, it is recommended to exclude TeamCity agent home from the background checks.

Please disable various indexing services. e.g. there might be problems with Windows Indexing Service. See [issue](#) for more details. Windows System Restore Feature might also need disabling.

Please also do not install software with background indexing like WinCVS, TortoiseCVS, TortoiseSVN and other Tortoise* products. This applies to server and also to agents if you use agent-side checkout.

Skype software is known to:

- use port 80 on the system so you might not be able to use TeamCity server using default 80 port.

- corrupt layout of pages displayed in Internet Explorer. Internet Explorer Skype plugin is to blame. ([TW-13052](#)).

[Back to top](#)

Subversion issues

svn: E175002: Received fatal alert: bad_record_mac

Please add system property `-Dsvnkit.http.sslProtocols=SSLv3,TLS` on the build server (see [Configuring TeamCity Server Startup Properties](#)).

If you use checkout on agent, add this property [on build agent](#) as well.

Subversion-related JVM Crashes

If JVM crashes while executing SVN-related code (e.g. under `org.tmatesoft.svn` package), you can try to disable it by either:

- Passing `-Dsvnkit.useJNA=false` JVM option to the crashing process (server or agent), or
- Making NTLM support less prioritative by passing `-Dsvnkit.http.methods=Basic,Digest,NTLM` JVM option.

Anyway, upgrading the JVM used to the [latest available version](#) is recommended.

[Back to top](#)

NUnit 2.4.6 Performance

Due to an issue in NUnit 2.4.6, its performance may be slower than NUnit 2.4.1. For additional information, please refer to the corresponding issue in our issue tracker: [TW-4709](#)

[Back to top](#)

StarTeam Performance

Using StarTeam SDK 9.0 instead of StarTeam SDK 9.3 on the TeamCity server can significantly improve VCS performance when there is a slow connection between TeamCity and StarTeam servers.

[Back to top](#)

Perforce 2009.2 Performance on Windows

If you run Perforce 2009.2 on Windows you may experience significant slow down. This is an issue with P4 server running on Windows. Please refer to corresponding [section](#) in Perforce documentation.

Wrong times for build scheduled triggering (Timezone issues)

Please make sure you use the latest JDK available for your platform (e.g. Oracle JDK [download](#)).

There were fixes in JDK 1.5 and 1.6 to address various wrong timezone reporting issues.

[Back to top](#)

Upgrading IntelliJ IDEA May Affect Active Pre-Tested Commits

Before you upgrade to IntelliJ IDEA X (or other IntelliJ X platform products) please make sure you do not have active pre-tested commits, otherwise they will not be able to be committed after upgrade.

This is only relevant if you use directory-based IDEA project (project files are stored under `.idea` directory).

Other Java Applications Running on the Same Server

If other web applications are available via the same hostname, a session cookie conflict can occur. This usually is visible via

random user logouts or losing session-level data. (e.g. [TW-12654](#)). To resolve this, you can use different host names when accessing the applications.

[Back to top](#)

The Server Does Not Start Claiming the Database is in Use

Only a single TeamCity server can work with one database, which is checked on the TeamCity server start. "The Database is in Use" error on the start-up is reported in either of the following cases:

- An attempt to start more than one TeamCity server connected to the same database
- A second TeamCity instance detected
- The internal HSQL database is being used by another application

The error is most probably caused by the fact that there is another running TeamCity installation which is connected to the same database. Check that the [database properties](#) are correct and there is no other TeamCity server using the same database.

In TeamCity 8.0 and earlier, if all the settings are correct, the error can occur when the TeamCity server or the database server has been shut down incorrectly.

The resolution depends on the database type:

- MySQL: restart the MySQL server and then start TeamCity again.
- PostgreSQL, Oracle, MS SQL: kill the connections from the incorrectly shut down TeamCity, and then start TeamCity again.
- Internal database (HSQL): remove the `buildserver.lock` file from the `TeamCity Data Directory\system` directory, and then start TeamCity again.

[Back to top](#)

Slow download from TeamCity server

If you experience slow speed when downloading artifacts from TeamCity, try checking the speed on the server machine, downloading from localhost.

If the speed is OK for the localhost, the issue can be in the network configuration or OS/hardware settings when combined with TeamCity(Tomcat) settings.

Please also make sure `<TeamCity Home>\conf\server.xml` file corresponds to the file included in TeamCity distribution (can be checked in `.tar.gz` distribution).

If you have the following "Connector" node (ports numbers can be different):

```
<Connector port="8111" protocol="HTTP/1.1"
  connectionTimeout="60000"
  redirectPort="8543"
  useBodyEncodingForURI="true"
/>
```

change it to:

```
<Connector port="8111" protocol="org.apache.coyote.http11.Http11NioProtocol"
  connectionTimeout="60000"
  redirectPort="8543"
  useBodyEncodingForURI="true"
  socket.txBufSize="64000"
  socket.rxBufSize="64000"
  tcpNoDelay="1"
/>
```

and restart TeamCity server.

If this does not help with the download speed, to investigate the case you might need to find an administrator with appropriate network-related issues investigation skills to look into the case.

Failure to publish artifacts to server behind IIS reverse proxy

This problem is only relevant to configurations that involve IIS reverse proxy between build server and agents. Sometimes a build agent can be found in infinite loop trying to publish build artifacts to server. Build log looks like this:

```
[11:15:05]Publishing artifacts
[11:15:05][Publishing artifacts] Collecting files to publish: [toZip/** => artifact.zip]
[11:15:06][Publishing artifacts] Creating archive artifact.zip (9s)
[11:15:06][Creating archive artifact.zip] Creating
C:\BuildAgent\temp\buildTmp\ZipPreprocessor2847146024236637664\artifact.zip
[11:15:15][Creating archive artifact.zip] Archive was created, file size 32142324 bytes
[11:15:15][Publishing artifacts] Sending toZip/**
[11:15:25][Publishing artifacts] Sending toZip/**
[11:15:39][Publishing artifacts] Sending toZip/**
[11:15:48][Publishing artifacts] Sending toZip/**
[11:16:01][Publishing artifacts] Sending toZip/**
[11:16:16][Publishing artifacts] Sending toZip/**
```

meanwhile teamcity-agent.log is filled with 404 responses from IIS:

```
[2012-08-01 12:04:55,514] WARN - jetbrains.buildServer.AGENT - <!DOCTYPE html PUBLIC
"-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
<title>404 - File or directory not found.</title>
<style type="text/css">
<!--
body{margin:0;font-size:.7em;font-family:Verdana, Arial, Helvetica,
sans-serif;background:#EEEEEE;}
fieldset{padding:0 15px 10px 15px;}
```

The most common cause for this is `maxAllowedContentLength` setting (in IIS) either

- is set to too small value
- is left unconfigured and so defaults to 30000000 bytes (<30 Mb)

So any artifact larger than `maxAllowedContentLength` is discarded by IIS
Check the settings value and try to rerun your build

SSL problems when connecting to HTTPS from TeamCity (handshake alert: unrecognized_name)

This problem may happen when changing JVM from 1.6 to 1.7 and connecting some incorrectly configured HTTPS servers. The problem and workaround for it are described in this issue: <http://youtrack.jetbrains.com/issue/TW-30210>

Please try running with antivirus software uninstalled before reporting the issue to JetBrains. e.g. see [the issue](#).