

# Deployit Command Plugin Manual

Version 3.8.1

## Table of Contents

Table of Contents	2
Preface	3
Overview	3
Features	3
Plugin Concepts	3
Command	3
Undo Command	3
Command Order	3
Requirements	3
Usage in Deployment Packages	3
Using the deployables and deployed	4
Deployable vs. Container Table	4
Deployed Actions Table	4
Sample Usage Senario - Provision a Tomcat server	4
CI Reference	5
Configuration Item Overview	5
Deployables	5
Deployeds	5
Containers	5
Other Configuration Items	5
Configuration Item Details	6
cmd.Command	6
cmd.DeployedCommand	6
overthere.CifsHost	6
overthere.Host	8
overthere.Jumpstation	9
overthere.LocalHost	9
overthere.RemoteHost	9
overthere.SshHost	10
overthere.SshJumpstation	12

## Preface

This document describes the functionality provided by the Command Plugin.

Refer to the *Deployit Reference Manual* for background information on Deployit and deployment concepts.

## Overview

As a system administrator, the need occasionally arises to execute adhoc scripts or OS commands on remote systems. The process usually entails having to manually login to each system, copy any required resources to said system and finally executing scripts/commands to process the resources or configure the remote system. The process is acceptable for a single system, but tends not to scale when performing the tasks on entire server farms. The manual intensive process becomes tedious and error prone. The Command Plugin helps with these tedious processes and significantly reduces the chances of errors.

A system administrator could also use the Command Plugin to reuse existing deployment scripts with Deployit, before choosing to move the deployment logic to a more reusable, easily maintainable plugin form.

## Features

- Execute an OS (Unix, Windows) command on a host
- Execute a script on a host
- Associate undo commands
- Copy associated command resources to a host

## Plugin Concepts

### Command

A Command encapsulates an OS specific command, as one would enter at the command prompt of a native OS command shell. The OS command is captured in the Command's `commandLine` property; e.g. 'echo hello >> /tmp/hello.txt'. The Command also has the capability of uploading any dependent files to the target system and make those available to the `commandLine` with the use of a placeholder; e.g. 'cat \${uploadedHello.txt} >> /tmp/hello.txt'.

### Undo Command

An undo Command has the same characteristics as a Command, except that it reverses the effect of the original Command it is associated with. An undo Command usually runs when the associated Command is undeployed or upgraded.

### Command Order

The order in which the Command is run in relation to other commands. The order allows for the chaining of commands to create a logical sequence of events. For example, an install tomcat command would execute before an install web application command, while a start tomcat command would be the last in the sequence.

## Requirements

This plugin requires:

- **Deployit:** version 3.5+

## Usage in Deployment Packages

Please refer to *Packaging Manual* for more details about the DAR packaging format.

Sample DAR MANIFEST.MF entries defining a package that can (un)provision a tomcat server using an install and uninstall script

```

Manifest-Version: 1.0
Deployit-Package-Format-Version: 1.3
CI-Application: CommandPluginSample
CI-Version: 1.0

Name: install-tc-command
CI-type: cmd.Command
CI-order: 50
CI-commandLine: /bin/sh ${install-tc.sh} ${tomcat.zip}
CI-undoCommand: uninstall-tc-command
CI-dependencies-EntryValue-1: install-tc.sh
CI-dependencies-EntryValue-2: tomcat.zip
CI-name: install-tc-command

Name: uninstall-tc-command
CI-type: cmd.Command
CI-order: 45
CI-commandLine: /bin/sh ${uninstall-tc.sh}
CI-dependencies-EntryValue-1: uninstall-tc.sh
CI-name: uninstall-tc-command

Name: tomcat-6.0.32.zip
CI-name: tomcat.zip
CI-type: file.File

Name: install-tc.sh
CI-type: file.File
CI-name: install-tc.sh

Name: uninstall-tc.sh
CI-type: file.File
CI-name: uninstall-tc.sh

```

## Using the deployables and deployed

### Deployable vs. Container Table

The following table describes which deployable / container combinations are possible. Note that the CIs can only be targeted to containers derived from [Host](#).

Deployables	Containers	Generated Deployed
cmd.Command	overthere.Host	cmd.DeployedCommand

### Deployed Actions Table

The following table describes the effect a deployed has on its container.

Deployed	Create	Destroy	Modify
cmd.DeployedCommand	<ul style="list-style-type: none"> <li>Upload command resources to host</li> <li>Resolve command line placeholder references with absolute paths to the uploaded resource files on host</li> <li>Execute command line on host</li> </ul>	<ul style="list-style-type: none"> <li>Run the undo command associated with the deployed command, if exists. Actions are same as described for <i>Create</i></li> </ul>	<ul style="list-style-type: none"> <li>Run the undo command associated with the deployed command, if exists. Actions are same as described for <i>Create</i></li> <li>Run the modified command. Actions are same as described for <i>Create</i></li> </ul>

### Sample Usage Senario - Provision a Tomcat server

For illustration purposes, we take a simplistic view of installing Tomcat. In reality however, your installation of Tomcat would take on a far more comprehensive form.

Tomcat is distributed as a zip. For this example, we create an installation script to unzip the distribution on the host. The uninstall script simply shuts down a running Tomcat and deletes the installation directory.

#### Create the installation script (install-tc.sh)

```

#!/bin/sh
set -e
if [ -e "/apache-tomcat-6.0.32" ]
then
    echo "/apache-tomcat-6.0.32 already exists. remove to continue."
    exit 1
fi
unzip $1 -d /
chmod +x /apache-tomcat-6.0.32/bin/*.sh

```

#### Create the uninstall script (uninstall-tc.sh)

```
#!/bin/sh
set -e
/apache-tomcat-6.0.32/bin/shutdown.sh
rm -rf /apache-tomcat-6.0.32
```

#### MANIFEST snippet defining the command to trigger the execution of the install script for the initial deployment

The following command will be executed at order 50 in the generated step list. '/bin/sh' is used on the host to execute the install script which takes a single parameter, the absolute path to the tomcat.zip on the host. When the command is undeployed, `uninstall-tc-command` will be executed.

```
Name: install-tc-command
CI-type: cmd.Command
CI-order: 50
CI-commandLine: /bin/sh ${install-tc.sh} ${tomcat.zip}
CI-undoCommand: uninstall-tc-command
CI-dependencies-EntryValue-1: install-tc.sh
CI-dependencies-EntryValue-2: tomcat.zip
CI-name: install-tc-command
```

#### MANIFEST snippet defining the undo command to trigger the execution of the uninstall script for the undeploy

The undo command will be executed at order 45 in the generated step list. Note that it has a lower order than the `install-tc-command`. This ensures that the undo command will always run before the `install-tc-command` during an upgrade.

```
Name: uninstall-tc-command
CI-type: cmd.Command
CI-order: 45
CI-commandLine: /bin/sh ${uninstall-tc.sh}
CI-dependencies-EntryValue-1: uninstall-tc.sh
CI-name: uninstall-tc-command
```

See the Usage in Deployment Packages section for the complete MANIFEST.MF

## CI Reference

### Configuration Item Overview

#### Deployables

CI	Description
<a href="#">cmd.Command</a>	Command specification that is executed on a host

#### Deployeds

CI	Description
<a href="#">cmd.DeployedCommand</a>	Command deployed to a Host

#### Containers

CI	Description
<a href="#">overthere.CifsHost</a>	Machine that can be connected to using either WinRM or Telnet and can perform file manipulation via the CIFS protocol
<a href="#">overthere.Host</a>	Machine that runs middleware, on which scripts can be executed, etc
<a href="#">overthere.Jumpstation</a>	Base class for jumpstations
<a href="#">overthere.LocalHost</a>	Machine on which the Deployit Server is running
<a href="#">overthere.RemoteHost</a>	Description unavailable
<a href="#">overthere.SshHost</a>	Machine that can be connected to using SSH
<a href="#">overthere.SshJumpstation</a>	Machine that can be used to create a tunneled connection to a destination host

#### Other Configuration Items

CI	Description
<a href="#">cmd.Command</a>	Command specification that is executed on a host
<a href="#">cmd.DeployedCommand</a>	Command deployed to a Host
<a href="#">overthere.CifsHost</a>	Machine that can be connected to using either WinRM or Telnet and can perform file manipulation via the CIFS protocol
<a href="#">overthere.Host</a>	Machine that runs middleware, on which scripts can be executed, etc
<a href="#">overthere.Jumpstation</a>	Base class for jumpstations
<a href="#">overthere.LocalHost</a>	Machine on which the Deployit Server is running
<a href="#">overthere.RemoteHost</a>	Description unavailable
<a href="#">overthere.SshHost</a>	Machine that can be connected to using SSH
<a href="#">overthere.SshJumpstation</a>	Machine that can be used to create a tunneled connection to a destination host

## Configuration Item Details

### cmd.Command

**Type Hierarchy** udm.BaseDeployable >> udm.BaseConfigurationItem

**Interfaces** udm.Tagable, udm.Deployable, udm.ConfigurationItem

Command specification that is executed on a host

Public Properties	
<b>* order</b> : <b>INTEGER</b> = 50	Order of the command
<b>commandLine</b> : <b>STRING</b>	Command line to execute on host. Dependent artifacts can be referred to using \${artifact name}.
<b>dependencies</b> : <b>SET_OF_CI</b> <file.File>	Artifacts that the command depends on
<b>runUndoCommandOnUpgrade</b> : <b>BOOLEAN</b> = true	Indicates whether the undoCommand should be run on an upgrade
<b>tags</b> : <b>SET_OF_STRING</b>	If set, this deployable will only be mapped automatically to containers with the same tag.
<b>undoCommand</b> : <b>CI</b> <cmd.Command >	Command to execute when undeploying command

### cmd.DeployedCommand

**Type Hierarchy** udm.BaseDeployed >> udm.BaseConfigurationItem

**Interfaces** udm.Deployed, udm.ConfigurationItem

Command deployed to a Host

Parent	
<b>* container</b> : <b>CI</b> <udm.Container>	The container on which this deployed runs.
Public Properties	
<b>* order</b> : <b>INTEGER</b> = 50	Order of the command
<b>commandLine</b> : <b>STRING</b>	Command line to execute on host. Dependent artifacts can be referred to using \${artifact name}.
<b>dependencies</b> : <b>SET_OF_CI</b> <file.File>	Artifacts that the command depends on
<b>deployable</b> : <b>CI</b> <udm.Deployable>	The deployable that this deployed is derived from.
<b>rerunCommand</b> : <b>BOOLEAN</b>	Forces the command to be rerun.
<b>runUndoCommandOnUpgrade</b> : <b>BOOLEAN</b>	Indicates whether the undoCommand should be run on an upgrade
<b>undoCommand</b> : <b>CI</b> <cmd.Command >	Command to execute when undeploying command

### overthere.CifsHost

**Type Hierarchy** [overthere.RemoteHost](#) >> [overthere.Host](#) >> udm.BaseContainer >>  
udm.BaseConfigurationItem

**Interfaces** udm.Taggable, udm.ConfigurationItem, udm.Container,  
overthere.HostContainer

Machine that can be connected to using either WinRM or Telnet and can perform file manipulation via the CIFS protocol

Hidden Properties		
* <b>connectionTimeoutMillis</b> : <b>INTEGER</b> = 1200000	Number of milliseconds Overthere waits for a connection to a remote host to be established	
* <b>protocol</b> : <b>STRING</b> = cifs	Protocol	
* <b>tmpFileCreationRetries</b> : <b>INTEGER</b> = 1000	Number of times Overthere attempts to create a temporary file with a unique name	
* <b>winrmContext</b> : <b>STRING</b> = /wsman	Context used by the WinRM server	
* <b>winrmEnvelopSize</b> : <b>INTEGER</b> = 153600	Envelop size for WinRM messages	
* <b>winrmHttpsCertificateTrustStrategy</b> : <b>ENUM</b> [STRICT, SELF_SIGNED, ALLOW_ALL] = STRICT	HTTPS certifiacte trust strategy for WinRM over HTTPS	
* <b>winrmHttpsHostnameVerificationStrategy</b> : <b>ENUM</b> [STRICT, BROWSER_COMPATIBLE, ALLOW_ALL] = STRICT	HTTPS host name verification strategy for WinRM over HTTPS	
* <b>winrmLocale</b> : <b>STRING</b> = en-US	Locale to use for WinRM messages	
* <b>winrmTimeout</b> : <b>STRING</b> = PT60.000S	Timeout to use for WinRM messages in XML schema duration format	
<b>tmpDeleteOnDisconnect</b> : <b>BOOLEAN</b> = true	Whether to delete the temporary connection directory when the connection is closed	
<b>winrmKerberosAddPortToSpn</b> : <b>BOOLEAN</b> = false	Add the port number (e.g. 5985) to the service principal name (SPN) for which a Kerberos ticket is requested	
<b>winrmKerberosDebug</b> : <b>BOOLEAN</b> = false	Enable Kerberos debug messages	
<b>winrmKerberosUseHttpSpn</b> : <b>BOOLEAN</b> = false	Use the HTTP protocol in the service principal name (SPN) for which a Kerberos ticket is requested, instead of the default WSMAN protocol	
Control Tasks		
<b>checkConnection</b>	Check connection	

## overthere.Host

### Virtual Type

**Type Hierarchy** udm.BaseContainer >> udm.BaseConfigurationItem

**Interfaces** udm.Taggable, udm.ConfigurationItem, udm.Container, overthere.HostContainer

Machine that runs middleware, on which scripts can be executed, etc.

Public Properties	
* <b>os</b> : <b>ENUM</b> [WINDOWS, UNIX]	Operating system
<b>tags</b> : <b>SET_OF_STRING</b>	If set, only deployables with the same tag will be automatically mapped to this container.
<b>temporaryDirectoryPath</b> : <b>STRING</b>	Directory into which temporary files are stored. Will be cleaned up when the connection is closed.
Hidden Properties	
* <b>protocol</b> : <b>STRING</b>	Protocol to use when connecting to this host
* <b>tmpFileCreationRetries</b> : <b>INTEGER</b> = 1000	Number of times Overthere attempts to create a temporary file with a unique name
<b>tmpDeleteOnDisconnect</b> : <b>BOOLEAN</b> = true	Whether to delete the temporary connection directory when the connection is closed
Control Tasks	
<b>checkConnection</b>	Check connection



**overthere.Jumpstation****Virtual Type**

**Type Hierarchy** [overthere.RemoteHost](#) >> [overthere.Host](#) >> [udm.BaseContainer](#) >> [udm.BaseConfigurationItem](#)

**Interfaces** [udm.Taggable](#), [udm.ConfigurationItem](#), [udm.Container](#), [overthere.HostContainer](#)

Base class for jumpstations

Public Properties	
<b>jumpstation</b> : <a href="#">CI</a> < <a href="#">overthere.Jumpstation</a> >	Jumpstation that should be used to reach this host
<b>tags</b> : <a href="#">SET_OF_STRING</a>	If set, only deployables with the same tag will be automatically mapped to this container.
Hidden Properties	
* <b>connectionTimeoutMillis</b> : <a href="#">INTEGER</a> = 1200000	Number of milliseconds Overthere waits for a connection to a remote host to be established
* <b>os</b> : <a href="#">ENUM</a> [ <a href="#">WINDOWS</a> , <a href="#">UNIX</a> ] = <a href="#">UNIX</a>	Os
* <b>protocol</b> : <a href="#">STRING</a>	Protocol to use when connecting to this host
* <b>tmpFileCreationRetries</b> : <a href="#">INTEGER</a> = 1000	Number of times Overthere attempts to create a temporary file with a unique name
<b>temporaryDirectoryPath</b> : <a href="#">STRING</a>	The default platform value (/tmp) suffices as no temporary files will be placed on the jumpstation
<b>tmpDeleteOnDisconnect</b> : <a href="#">BOOLEAN</a> = <a href="#">true</a>	Whether to delete the temporary connection directory when the connection is closed
Control Tasks	
<b>checkConnection</b>	Check connection

**overthere.LocalHost**

**Type Hierarchy** [overthere.Host](#) >> [udm.BaseContainer](#) >> [udm.BaseConfigurationItem](#)

**Interfaces** [udm.Taggable](#), [udm.ConfigurationItem](#), [udm.Container](#), [overthere.HostContainer](#)

Machine on which the Deployit Server is running

Public Properties	
* <b>os</b> : <a href="#">ENUM</a> [ <a href="#">WINDOWS</a> , <a href="#">UNIX</a> ]	Operating system
<b>tags</b> : <a href="#">SET_OF_STRING</a>	If set, only deployables with the same tag will be automatically mapped to this container.
<b>temporaryDirectoryPath</b> : <a href="#">STRING</a>	Directory into which temporary files are stored. Will be cleaned up when the connection is closed.
Hidden Properties	
* <b>protocol</b> : <a href="#">STRING</a> = <a href="#">local</a>	Protocol
* <b>tmpFileCreationRetries</b> : <a href="#">INTEGER</a> = 1000	Number of times Overthere attempts to create a temporary file with a unique name
<b>tmpDeleteOnDisconnect</b> : <a href="#">BOOLEAN</a> = <a href="#">true</a>	Whether to delete the temporary connection directory when the connection is closed
Control Tasks	
<b>checkConnection</b>	Check connection

**overthere.RemoteHost**

**Virtual Type****Type Hierarchy** [overthere.Host](#) >> udm.BaseContainer >> udm.BaseConfigurationItem**Interfaces** udm.Taggable, udm.ConfigurationItem, udm.Container, overthere.HostContainer

Description unavailable

Public Properties	
<b>* os</b> : <b>ENUM</b> [WINDOWS, UNIX]	Operating system
<b>jumpstation</b> : <b>CI</b> < <a href="#">overthere.Jumpstation</a> >	Jumpstation that should be used to reach this host
<b>tags</b> : <b>SET_OF_STRING</b>	If set, only deployables with the same tag will be automatically mapped to this container.
<b>temporaryDirectoryPath</b> : <b>STRING</b>	Directory into which temporary files are stored. Will be cleaned up when the connection is closed.
Hidden Properties	
<b>* connectionTimeoutMillis</b> : <b>INTEGER</b> = 1200000	Number of milliseconds Overthere waits for a connection to a remote host to be established
<b>* protocol</b> : <b>STRING</b>	Protocol to use when connecting to this host
<b>* tmpFileCreationRetries</b> : <b>INTEGER</b> = 1000	Number of times Overthere attempts to create a temporary file with a unique name
<b>tmpDeleteOnDisconnect</b> : <b>BOOLEAN</b> = true	Whether to delete the temporary connection directory when the connection is closed
Control Tasks	
<b>checkConnection</b>	Check connection

**overthere.SshHost****Type Hierarchy** [overthere.RemoteHost](#) >> [overthere.Host](#) >> udm.BaseContainer >> udm.BaseConfigurationItem**Interfaces** udm.Taggable, udm.ConfigurationItem, udm.Container, overthere.HostContainer

Machine that can be connected to using SSH

Public Properties	
<b>* address</b> : <b>STRING</b>	Address of the host
<b>* connectionType</b> : <b>ENUM</b> [SFTP, SFTP_CYGWIN, SFTP_WINSSHD, SCP, SUDO, INTERACTIVE_SUDO, TUNNEL] = SFTP	Type of SSH connection to create
<b>* os</b> : <b>ENUM</b> [WINDOWS, UNIX]	Operating system
<b>* port</b> : <b>INTEGER</b> = 22	Port on which the SSH server runs
<b>* username</b> : <b>STRING</b>	Username to connect with
<b>jumpstation</b> : <b>CI&lt;overthere.Jumpstation &gt;</b>	Jumpstation that should be used to reach this host
<b>passphrase</b> : <b>STRING</b>	Optional passphrase for the private key in the private key file
<b>password</b> : <b>STRING</b>	Password to use for authentication
<b>privateKeyFile</b> : <b>STRING</b>	Private key file to use for authentication
<b>sudoUsername</b> : <b>STRING</b>	Username to sudo to when accessing files or executing commands
<b>tags</b> : <b>SET_OF_STRING</b>	If set, only deployables with the same tag will be automatically mapped to this container.
<b>temporaryDirectoryPath</b> : <b>STRING</b>	Directory into which temporary files are stored. Will be cleaned up when the connection is closed.
Hidden Properties	
<b>* connectionTimeoutMillis</b> : <b>INTEGER</b> = 1200000	Number of milliseconds Overthere waits for a connection to a remote host to be established
<b>* interactiveKeyboardAuthRegex</b> : <b>STRING</b> = <b>.*Password:[ ]?</b>	Regular expression to look for in keyboard-interactive authentication before sending the password
<b>* protocol</b> : <b>STRING</b> = <b>ssh</b>	Protocol
<b>* sudoCommandPrefix</b> : <b>STRING</b> = <b>sudo -u {0}</b>	Sudo command to prefix to the original command. The placeholder {0} is replaced with the sudoUsername
<b>* sudoPasswordPromptRegex</b> : <b>STRING</b> = <b>.*[Pp]assword.*:</b>	Regular expression to look for in interactive sudo before sending the password
<b>* tmpFileCreationRetries</b> : <b>INTEGER</b> = 1000	Number of times Overthere attempts to create a temporary file with a unique name
<b>allocateDefaultPty</b> : <b>BOOLEAN</b> = <b>false</b>	If true, a default PTY (dummy:80:24:0:0) is allocated when executing a command
<b>allocatePty</b> : <b>STRING</b>	Specification for the PTY to be allocated when executing a command. The format is TERM:COLS:ROWS:WIDTH:HEIGHT, e.g. xterm:80:24:0:0
<b>sudoOverrideUmask</b> : <b>BOOLEAN</b> = <b>true</b>	If true, permissions are explicitly changed with chmod -R go+rX after uploading a file or directory
<b>sudoPreserveAttributesOnCopyFromTempFile</b> : <b>BOOLEAN</b> = <b>true</b>	If true, files are copied from the connection temporary directory using the -p flag to the cp command
<b>sudoPreserveAttributesOnCopyToTempFile</b> : <b>BOOLEAN</b> = <b>true</b>	If true, files are copied to the connection temporary directory using the -p flag to the cp command
<b>sudoQuoteCommand</b> : <b>BOOLEAN</b> = <b>false</b>	If true, the original command is quoted when it is prefixed with sudoCommandPrefix
<b>tmpDeleteOnDisconnect</b> : <b>BOOLEAN</b> = <b>true</b>	Whether to delete the temporary connection directory when the connection is closed

Control Tasks	
<b>checkConnection</b>	Check connection

## overthere.SshJumpstation

**Type Hierarchy** [overthere.Jumpstation](#) >> [overthere.RemoteHost](#) >> [overthere.Host](#) >>  
 udm.BaseContainer >> udm.BaseConfigurationItem

**Interfaces** udm.Taggable, udm.ConfigurationItem, udm.Container, overthere.HostContainer

Machine that can be used to create a tunneled connection to a destination host

Public Properties
* <b>address</b> : <a href="#">STRING</a> Address of the host
* <b>port</b> : <a href="#">INTEGER</a> = 22 Port on which the SSH server runs
* <b>username</b> : <a href="#">STRING</a> Username to connect with
<b>jumpstation</b> : <a href="#">CI&lt;overthere.Jumpstation &gt;</a> Jumpstation that should be used to reach this host
<b>passphrase</b> : <a href="#">STRING</a> Optional passphrase for the private key in the private key file
<b>password</b> : <a href="#">STRING</a> Password to use for authentication
<b>privateKeyFile</b> : <a href="#">STRING</a> Private key file to use for authentication
<b>tags</b> : <a href="#">SET_OF_STRING</a> If set, only deployables with the same tag will be automatically mapped to this container.
Hidden Properties
* <b>connectionTimeoutMillis</b> : <a href="#">INTEGER</a> = 1200000 Number of milliseconds Overthere waits for a connection to a remote host to be established
* <b>connectionType</b> : <a href="#">ENUM [SFTP, SFTP_CYGWIN, SFTP_WINSSHD, SCP, SUDO, INTERACTIVE_SUDO, TUNNEL]</a> = TUNNEL Connection Type
* <b>interactiveKeyboardAuthRegex</b> : <a href="#">STRING</a> = .*Password:[ ]? Regular expression to look for in keyboard-interactive authentication before sending the password
* <b>os</b> : <a href="#">ENUM [WINDOWS, UNIX]</a> = UNIX Os
* <b>portAllocationRangeStart</b> : <a href="#">INTEGER</a> = 1025 Port from where to start looking for freely available ports to use as the local part of an SSH port forward
* <b>protocol</b> : <a href="#">STRING</a> = ssh Protocol
* <b>tmpFileCreationRetries</b> : <a href="#">INTEGER</a> = 1000 Number of times Overthere attempts to create a temporary file with a unique name
<b>temporaryDirectoryPath</b> : <a href="#">STRING</a> The default platform value (/tmp) suffices as no temporary files will be placed on the jumpstation
<b>tmpDeleteOnDisconnect</b> : <a href="#">BOOLEAN</a> = true Whether to delete the temporary connection directory when the connection is closed
Control Tasks
<b>checkConnection</b>
Check connection