

Deployit Command Plugin Manual

Version 3.7.5

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
Deployable Configuration Items	5
Deployed Configuration Items	5
Topology Configuration Items	5
Virtual Topology Configuration Items	5
Configuration Item Details	5
cmd.Command	5
cmd.DeployedCommand	6
overthere.CifsHost	6
overthere.Host	7
overthere.Jumpstation	8
overthere.LocalHost	8
overthere.SshHost	9
overthere.SshJumpstation	10

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

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

Deployable Configuration Items

CI	Description
cmd.Command	Command specification that is executed on a host

Deployed Configuration Items

CI	Description
cmd.DeployedCommand	Command deployed to a Host

Topology Configuration Items

CI	Description
overthere.CifsHost	Machine that can be connected to using either WinRM or Telnet and can perform file manipulation via the CIFS protocol
overthere.LocalHost	Machine on which the Deployit Server is running
overthere.SshHost	Machine that can be connected to using SSH
overthere.SshJumpstation	Machine that can be used to create a tunneled connection to a destination host

Virtual Topology Configuration Items

CI	Description
overthere.Host	Machine that runs middleware, on which scripts can be executed, etc
overthere.Jumpstation	Base class for jumpstations

Configuration Item Details

cmd.Command

Hierarchy udm.BaseDeployable >> udm.BaseConfigurationItem

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

Command specification that is executed on a host


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

cmd.DeployedCommand

Hierarchy udm.BaseDeployed >> udm.BaseConfigurationItem

Interfaces udm.Deployed, udm.ConfigurationItem

Command deployed to a Host

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

overthere.CifsHost

Hierarchy [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

Public Properties	
address	: STRING
Address of the host	
connectionType	: ENUM [TELNET, WINRM, WINRM_HTTP, WINRM_HTTPS] = WINRM
Connection Type	
os	: ENUM [WINDOWS, UNIX]
Operating system	
password	: STRING
Password to use for authentication	
username	: STRING

Username to connect with
cifsPort : INTEGER = 445
Port on which the CIFS server runs
deploymentGroup : INTEGER
If the group-orchestrator is enabled, all containers with the same deployment group number will be deployed to at the same time. The groups are ordered by this number.
jumpstation : CI <overthere.Jumpstation>
Jumpstation that should be used to reach this host
pathShareMappings : MAP_STRING_STRING
Mapping from Windows paths to Windows share names, e.g. C:\IBMWebSphere -> WebSphereShare
port : INTEGER
Port on which the Telnet or WinRM server runs
tags : SET_OF_STRING
If set, only deployables with the same tag will be automatically mapped to this container.
temporaryDirectoryPath : STRING
Directory into which temporary files are stored. Will be cleaned up when the connection is closed.
winrmEnableHttps : BOOLEAN = false
Enable SSL communication to the WinRM server

Hidden Properties

connectionTimeoutMillis : INTEGER = 1200000
Number of milliseconds Overthere waits for a connection to a remote host to be established
protocol : STRING = cifs
Protocol
tmpFileCreationRetries : INTEGER = 1000
Number of times Overthere attempts to create a temporary file with a unique name
winrmContext : STRING = /wsman
Context used by the WinRM server
winrmEnvelopSize : INTEGER = 153600
Envelop size for WinRM messages
winrmHttpsCertificateTrustStrategy : ENUM [STRICT, SELF_SIGNED, ALLOW_ALL] = STRICT
HTTPS certificate trust strategy for WinRM over HTTPS
winrmHttpsHostnameVerificationStrategy : ENUM [STRICT, BROWSER_COMPATIBLE, ALLOW_ALL] = STRICT
HTTPS host name verification strategy for WinRM over HTTPS
winrmLocale : STRING = en-US
Locale to use for WinRM messages
winrmTimeout : STRING = PT60.000S
Timeout to use for WinRM messages in XML schema duration format
tmpDeleteOnDisconnect : BOOLEAN = true
Whether to delete the temporary connection directory when the connection is closed
winrmDebugKerberosAuth : BOOLEAN = false
If true, enabled Kerberos debug messages

Control Tasks

checkConnection

Check connection

overthere.Host

Hierarchy udm.BaseContainer >> udm.BaseConfigurationItem

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

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

Public Properties

os : **ENUM** [WINDOWS, UNIX]

Operating system

deploymentGroup : **INTEGER**

If the group-orchestrator is enabled, all containers with the same deployment group number

will be deployed to at the same time. The groups are ordered by this number.

jumpstation : [CI<overthere.Jumpstation>](#)

Jumpstation that should be used to reach this host

tags : [SET_OF_STRING](#)

If set, only deployables with the same tag will be automatically mapped to this container.

temporaryDirectoryPath : [STRING](#)

Directory into which temporary files are stored. Will be cleaned up when the connection is closed.

Hidden Properties

connectionTimeoutMillis : [INTEGER](#) = 1200000

Number of milliseconds Overthere waits for a connection to a remote host to be established

protocol : [STRING](#)

Protocol to use when connecting to this host

tmpFileCreationRetries : [INTEGER](#) = 1000

Number of times Overthere attempts to create a temporary file with a unique name

tmpDeleteOnDisconnect : [BOOLEAN](#) = true

Whether to delete the temporary connection directory when the connection is closed

Control Tasks

checkConnection

Check connection

overthere.Jumpstation

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

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

Base class for jumpstations

Public Properties

deploymentGroup : [INTEGER](#)

If the group-orchestrator is enabled, all containers with the same deployment group number will be deployed to at the same time. The groups are ordered by this number.

jumpstation : [CI<overthere.Jumpstation>](#)

Jumpstation that should be used to reach this host

tags : [SET_OF_STRING](#)

If set, only deployables with the same tag will be automatically mapped to this container.

Hidden Properties

connectionTimeoutMillis : [INTEGER](#) = 1200000

Number of milliseconds Overthere waits for a connection to a remote host to be established

os : [ENUM \[WINDOWS, UNIX\]](#) = UNIX

Os

protocol : [STRING](#)

Protocol to use when connecting to this host

tmpFileCreationRetries : [INTEGER](#) = 1000

Number of times Overthere attempts to create a temporary file with a unique name

temporaryDirectoryPath : [STRING](#)

The default platform value (/tmp) suffices as no temporary files will be placed on the jumpstation

tmpDeleteOnDisconnect : [BOOLEAN](#) = true

Whether to delete the temporary connection directory when the connection is closed

Control Tasks

checkConnection

Check connection

overthere.LocalHost

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	
os	: ENUM [WINDOWS, UNIX]
Operating system	
deploymentGroup	: INTEGER
If the group-orchestrator is enabled, all containers with the same deployment group number will be deployed to at the same time. The groups are ordered by this number.	
jumpstation	: CI <overthere.Jumpstation>
Jumpstation that should be used to reach this host	
tags	: SET_OF_STRING
If set, only deployables with the same tag will be automatically mapped to this container.	
temporaryDirectoryPath	: STRING
Directory into which temporary files are stored. Will be cleaned up when the connection is closed.	

Hidden Properties	
connectionTimeoutMillis	: INTEGER = 1200000
Number of milliseconds Overthere waits for a connection to a remote host to be established	
protocol	: STRING = local
Protocol	
tmpFileCreationRetries	: INTEGER = 1000
Number of times Overthere attempts to create a temporary file with a unique name	
tmpDeleteOnDisconnect	: BOOLEAN = true
Whether to delete the temporary connection directory when the connection is closed	

Control Tasks	
checkConnection	
Check connection	

overthere.SshHost

Hierarchy [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	
address	: STRING
Address of the host	
connectionType	: ENUM [SFTP, SFTP_CYGWIN, SFTP_WINSSHD, SCP, SUDO, INTERACTIVE_SUDO, TUNNEL] = SFTP
Type of SSH connection to create	
os	: ENUM [WINDOWS, UNIX]
Operating system	
port	: INTEGER = 22
Port on which the SSH server runs	
username	: STRING
Username to connect with	
deploymentGroup	: INTEGER
If the group-orchestrator is enabled, all containers with the same deployment group number will be deployed to at the same time. The groups are ordered by this number.	
jumpstation	: CI <overthere.Jumpstation>
Jumpstation that should be used to reach this host	
passphrase	: STRING
Optional passphrase for the private key in the private key file	
password	: STRING
Password to use for authentication	
privateKeyFile	: STRING
Private key file to use for authentication	
sudoUsername	: STRING
Username to sudo to when accessing files or executing commands	
tags	: SET_OF_STRING

If set, only deployables with the same tag will be automatically mapped to this container.

temporaryDirectoryPath : **STRING**

Directory into which temporary files are stored. Will be cleaned up when the connection is closed.

Hidden Properties

connectionTimeoutMillis : **INTEGER** = 1200000

Number of milliseconds Overthere waits for a connection to a remote host to be established

interactiveKeyboardAuthRegex : **STRING** = .*Password:[]?

Regular expression to look for in keyboard-interactive authentication before sending the password

protocol : **STRING** = ssh

Protocol

sudoCommandPrefix : **STRING** = sudo -u {0}

Sudo command to prefix to the original command. The placeholder {0} is replaced with the sudoUsername

sudoPasswordPromptRegex : **STRING** = .*[Pp]assword.*:

Regular expression to look for in interactive sudo before sending the password

tmpFileCreationRetries : **INTEGER** = 1000

Number of times Overthere attempts to create a temporary file with a unique name

allocateDefaultPty : **BOOLEAN** = false

If true, a default PTY (dummy:80:24:0:0) is allocated when executing a command

allocatePty : **STRING**

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

sudoOverrideUmask : **BOOLEAN** = true

If true, permissions are explicitly changed with chmod -R go+rX after uploading a file or directory

sudoPreserveAttributesOnCopyFromTempFile : **BOOLEAN** = true

If true, files are copied from the connection temporary directory using the -p flag to the cp command

sudoPreserveAttributesOnCopyToTempFile : **BOOLEAN** = true

If true, files are copied to the connection temporary directory using the -p flag to the cp command

sudoQuoteCommand : **BOOLEAN** = false

If true, the original command is quoted when it is prefixed with sudoCommandPrefix

tmpDeleteOnDisconnect : **BOOLEAN** = true

Whether to delete the temporary connection directory when the connection is closed

Control Tasks

checkConnection

Check connection

overthere.SshJumpstation

Hierarchy [overthere.Jumpstation](#) >> [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

address : **STRING**

Address of the host

port : **INTEGER** = 22

Port on which the SSH server runs

username : **STRING**

Username to connect with

deploymentGroup : **INTEGER**

If the group-orchestrator is enabled, all containers with the same deployment group number will be deployed to at the same time. The groups are ordered by this number.

jumpstation : **CI**<[overthere.Jumpstation](#)>

Jumpstation that should be used to reach this host

passphrase : **STRING**

Optional passphrase for the private key in the private key file

password : **STRING**

Password to use for authentication

privateKeyFile : **STRING**

Private key file to use for authentication

tags : **SET_OF_STRING**

If set, only deployables with the same tag will be automatically mapped to this container.

Hidden Properties

connectionTimeoutMillis : **INTEGER** = 1200000

Number of milliseconds Overthere waits for a connection to a remote host to be established

connectionType : **ENUM** [SFTP, SFTP_CYGWIN, SFTP_WINSSHD, SCP, SUDO, INTERACTIVE_SUDO, TUNNEL] = TUNNEL

Connection Type

interactiveKeyboardAuthRegex : **STRING** = .*Password:[]?

Regular expression to look for in keyboard-interactive authentication before sending the password

os : **ENUM** [WINDOWS, UNIX] = UNIX

Os

portAllocationRangeStart : **INTEGER** = 1025

Port from where to start looking for freely available ports to use as the local part of an SSH port forward

protocol : **STRING** = ssh

Protocol

tmpFileCreationRetries : **INTEGER** = 1000

Number of times Overthere attempts to create a temporary file with a unique name

temporaryDirectoryPath : **STRING**

The default platform value (/tmp) suffices as no temporary files will be placed on the jumpstation

tmpDeleteOnDisconnect : **BOOLEAN** = true

Whether to delete the temporary connection directory when the connection is closed

Control Tasks

checkConnection

Check connection