

# Deployit Remoting Plugin Manual

Version 3.7.5

## Table of Contents

Table of Contents	2
Preface	3
Overview	3
Features	3
Requirements	3
Examples	3
Connecting through a tunnel	3
CI Reference	3
Configuration Item Overview	3
Topology Configuration Items	3
Virtual Topology Configuration Items	3
Configuration Item Details	3
overthere.CifsHost	3
overthere.Host	5
overthere.Jumpstation	5
overthere.LocalHost	6
overthere.SshHost	6
overthere.SshJumpstation	7

## Preface

This document describes the functionality provided by the Remoting plugin.

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

## Overview

The Remoting plugin is a Deployit plugin that allows Deployit to manipulate files and execute commands on remote hosts. It does so by using the **Overthere** framework. Overthere is a Java library to manipulate files and execute processes on remote hosts, i.e. do stuff "over there". See the [Overthere repository](#) for more information.

## Features

- Define remote host CIs to use as deployment targets.
- Define jump stations to connect to remote hosts.

## Requirements

- **Deployit requirements**
  - **Deployit:** version 3.5+
  - **Other Deployit Plugins:** None

## Examples

### Connecting through a tunnel

When Deployit cannot reach a Host directly, but that Host can only be reached through an SSH tunnel, you need to create a so-called Jumpstation Host. This can be set up as follows.

Two Infrastructure items need to be created, ie. the target 'overthere.Host' and the 'overthere.SshJumpstation' that will actually be used to connect to the target machine. Once these are created, they can be hooked up to each other, by pointing the 'jumpstation' property of the target machine to the created 'overthere.Jumpstation'.

Once Deployit starts a deployment to the target host, it will see that it needs to connect through the jumpstation, and will first open a connection to that machine, and then setup a dynamic ssh tunnel to the target machine.

## CI Reference

### Configuration Item Overview

#### Topology Configuration Items

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.LocalHost</a>	Machine on which the Deployit Server is running
<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

#### Virtual Topology Configuration Items

CI	Description
<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

### Configuration Item Details

#### overthere.CifsHost

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

**Interfaces** `udm.Taggable`, `udm.ConfigurationItem`, `udm.Container`,

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

Public Properties	
<b>address</b>	: <b>STRING</b> Address of the host
<b>connectionType</b>	: <b>ENUM [TELNET, WINRM, WINRM_HTTP, WINRM_HTTPS] = WINRM</b> Connection Type
<b>os</b>	: <b>ENUM [WINDOWS, UNIX]</b> Operating system
<b>password</b>	: <b>STRING</b> Password to use for authentication
<b>username</b>	: <b>STRING</b> Username to connect with
<b>cifsPort</b>	: <b>INTEGER = 445</b> Port on which the CIFS server runs
<b>jumpstation</b>	: <b>CI&lt;overthere.Jumpstation&gt;</b> Jumpstation that should be used to reach this host
<b>pathShareMappings</b>	: <b>MAP_STRING_STRING</b> Mapping from Windows paths to Windows share names, e.g. C:\IBMWebSphere -> WebSphereShare
<b>port</b>	: <b>INTEGER</b> Port on which the Telnet or WinRM server runs
<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.
<b>winrmEnableHttps</b>	: <b>BOOLEAN = false</b> Enable SSL communication to the WinRM server
Hidden Properties	
<b>connectionTimeoutMillis</b>	: <b>INTEGER = 1200000</b> Number of milliseconds Overthere waits for a connection to a remote host to be established
<b>protocol</b>	: <b>STRING = cifs</b> Protocol
<b>tmpFileCreationRetries</b>	: <b>INTEGER = 1000</b> Number of times Overthere attempts to create a temporary file with a unique name
<b>winrmContext</b>	: <b>STRING = /wsman</b> Context used by the WinRM server
<b>winrmEnvelopSize</b>	: <b>INTEGER = 153600</b> Envelop size for WinRM messages
<b>winrmHttpsCertificateTrustStrategy</b>	: <b>ENUM [STRICT, SELF_SIGNED, ALLOW_ALL] = STRICT</b> HTTPS certificate trust strategy for WinRM over HTTPS
<b>winrmHttpsHostnameVerificationStrategy</b>	: <b>ENUM [STRICT, BROWSER_COMPATIBLE, ALLOW_ALL] = STRICT</b> HTTPS host name verification strategy for WinRM over HTTPS
<b>winrmLocale</b>	: <b>STRING = en-US</b> Locale to use for WinRM messages
<b>winrmTimeout</b>	: <b>STRING = PT60.000S</b> Timeout to use for WinRM messages in XML schema duration format
<b>tmpDeleteOnDisconnect</b>	: <b>BOOLEAN = true</b> Whether to delete the temporary connection directory when the connection is closed
<b>winrmDebugKerberosAuth</b>	: <b>BOOLEAN = false</b> If true, enabled Kerberos debug messages
Control Tasks	
<b>checkConnection</b>	Check connection

**overthere.Host**

**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**

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

Operating system

**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**

**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	
<b>os</b>	: ENUM [WINDOWS, UNIX] Operating system
<b>jumpstation</b>	: Cl< <a href="#">overthere.Jumpstation</a> > Jumpstation that should be used to reach this host
<b>tags</b>	: SET_OF_STRING If set, only deployables with the same tag will be automatically mapped to this container.
<b>temporaryDirectoryPath</b>	: STRING Directory into which temporary files are stored. Will be cleaned up when the connection is closed.
Hidden Properties	
<b>connectionTimeoutMillis</b>	: INTEGER = 1200000 Number of milliseconds Overthere waits for a connection to a remote host to be established
<b>protocol</b>	: STRING = local Protocol
<b>tmpFileCreationRetries</b>	: INTEGER = 1000 Number of times Overthere attempts to create a temporary file with a unique name
<b>tmpDeleteOnDisconnect</b>	: BOOLEAN = true Whether to delete the temporary connection directory when the connection is closed
Control Tasks	
<b>checkConnection</b>	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	
<b>address</b>	: STRING Address of the host
<b>connectionType</b>	: ENUM [SFTP, SFTP_CYGWIN, SFTP_WINSSHD, SCP, SUDO, INTERACTIVE_SUDO, TUNNEL] = SFTP Type of SSH connection to create
<b>os</b>	: ENUM [WINDOWS, UNIX] Operating system
<b>port</b>	: INTEGER = 22 Port on which the SSH server runs
<b>username</b>	: STRING Username to connect with
<b>jumpstation</b>	: Cl< <a href="#">overthere.Jumpstation</a> > Jumpstation that should be used to reach this host
<b>passphrase</b>	: STRING Optional passphrase for the private key in the private key file
<b>password</b>	: STRING Password to use for authentication
<b>privateKeyFile</b>	: STRING Private key file to use for authentication
<b>sudoUsername</b>	: STRING Username to sudo to when accessing files or executing commands
<b>tags</b>	: 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

**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