

Deployit Remoting Plugin Manual

Version 3.7.1

Table of Content

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	4
overthere.CifsHost	4
overthere.Host	5
overthere.HostContainer	6
overthere.Jumpstation	6
overthere.LocalHost	7
overthere.SshHost	8
overthere.SshJumpstation	10

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 eachother, 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
overthere.CifsHost	A machine that can be connected to using either WinRM or Telnet and can perform file manipulation via the CIFS protocol
overthere.LocalHost	The machine on which the Deployit Server is running on
overthere.SshHost	A machine that can be connected to using ssh
overthere.SshJumpstation	A machine that can be used to create a tunneled connection to the destination host

Virtual Topology Configuration Items

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

Configuration Item Details

overthere.CifsHost

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

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

A 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_HTTP, WINRM_HTTPS] = TELNET	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
jumpstation : CI<overthere.Jumpstation>	If this host is not directly reachable, specify a jumpstation here which can be used to reach this host.
port : INTEGER	Port on which the Telnet or WinRM server runs
tags : SET_OF_STRING	The tags to map deployables to containers.
temporaryDirectoryPath : STRING	Directory into which temporary files are stored. Will be cleaned up when the connection is closed.

Hidden Properties**connectionTimeoutMillis** : *INTEGER = 1200000*

Connection Timeout Millis

protocol : *STRING = cifs*

Protocol

tmpFileCreationRetries : *INTEGER = 1000*

Tmp File Creation Retries

winrmContext : *STRING = /wsman*

Winrm Context

winrmEnvelopSize : *INTEGER = 153600*

Winrm Envelop Size

winrmLocale : *STRING = en-US*

Winrm Locale

winrmTimeout : *STRING = PT60.000S*

Winrm Timeout

tmpDeleteOnDisconnect : *BOOLEAN = true*

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

Control Tasks**checkConnection**

Check connection

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

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

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

Operating system

jumpstation : *CI<overthere.Jumpstation>*

If this host is not directly reachable, specify a jumpstation here which can be used to reach this host.

tags : *SET_OF_STRING*

The tags to map deployables to containers.

temporaryDirectoryPath : *STRING*

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

Hidden Properties

connectionTimeoutMillis : **INTEGER** = *1200000*

Connection Timeout Millis

protocol : **STRING**

Protocol to use when connecting to this host

tmpFileCreationRetries : **INTEGER** = *1000*

Tmp File Creation Retries

tmpDeleteOnDisconnect : **BOOLEAN** = *true*

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

Control Tasks

checkConnection

Check connection

overthere.HostContainer

null

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](#)>

If this host is not directly reachable, specify a jumpstation here which can be used to reach this host.

tags : **SET_OF_STRING**

The tags to map deployables to containers.

Hidden Properties**connectionTimeoutMillis** : INTEGER = 1200000

Connection Timeout Millis

os : ENUM [WINDOWS, UNIX] = UNIX

Os

protocol : STRING

Protocol to use when connecting to this host

tmpFileCreationRetries : INTEGER = 1000

Tmp File Creation Retries

temporaryDirectoryPath : STRING

Temporary Directory Path

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](#)

The machine on which the Deployit Server is running on.

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

Operating system

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

If this host is not directly reachable, specify a jumpstation here which can be used to reach this host.

tags : SET_OF_STRING

The tags to map deployables to containers.

temporaryDirectoryPath : STRING

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

Hidden Properties

connectionTimeoutMillis : **INTEGER** = *1200000*

Connection Timeout Millis

protocol : **STRING** = *local*

Protocol

tmpFileCreationRetries : **INTEGER** = *1000*

Tmp File Creation Retries

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](#)

A 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

jumpstation : *CI<overthere.Jumpstation>*

If this host is not directly reachable, specify a jumpstation here which can 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*

The tags to map deployables to containers.

temporaryDirectoryPath : *STRING*

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

Hidden Properties

connectionTimeoutMillis : **INTEGER** = *1200000*

Connection Timeout Millis

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*

Tmp File Creation Retries

allocateDefaultPty : **BOOLEAN** = *false*

If true, a default pty is allocated when executing a command. All sudo implementations require it for interactive sudo, some even require it for normal sudo. Some SSH server implementations (notably the one on AIX 5.3) crash when it is allocated.

sudoOverrideUmask : **BOOLEAN** = *false*

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

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](#)

A machine that can be used to create a tunneled connection to the 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>*

If this host is not directly reachable, specify a jumpstation here which can 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*

The tags to map deployables to containers.

Hidden Properties

connectionTimeoutMillis : *INTEGER = 1200000*

Connection Timeout Millis

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

protocol : *STRING = ssh*

Protocol to use when connecting to this host

tmpFileCreationRetries : *INTEGER = 1000*

Tmp File Creation Retries

temporaryDirectoryPath : *STRING*

Temporary Directory Path

tmpDeleteOnDisconnect : *BOOLEAN = true*

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

Control Tasks**checkConnection**

Check connection