

Deployit File Plugin Manual

Version 3.8.1

Table of Contents

Table of Contents	2
Preface	3
Overview	3
Features	3
Requirements	3
Usage in Deployment Packages	3
Using the deployables and deployed	3
Deployable vs. Container Table	3
Deployed Actions Table	3
CI Reference	4
Configuration Item Overview	4
Deployables	4
Deployeds	4
Containers	4
Other Configuration Items	4
Configuration Item Details	4
file.Archive	5
file.DeployedArchive	5
file.DeployedArtifactOnHost	5
file.DeployedFile	6
file.DeployedFolder	6
file.File	7
file.Folder	7
overthere.CifsHost	8
overthere.Host	9
overthere.Jumpstation	10
overthere.LocalHost	10
overthere.RemoteHost	10
overthere.SshHost	11
overthere.SshJumpstation	13

Preface

This document describes the functionality provided by the File Plugin.

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

Overview

In many cases, an application depends on external resources for its configuration. The application accesses these resources from a predefined location or using a predefined mechanism. In the simplest of forms, a resource can be described as a file, an archive or a folder (collection of files). The File Plugin enables the definition of such resources in a deployment package and subsequently managing them on a target host. The resources can contain placeholders that the plugin will replace when targeting to the specific host, thus allowing resources to be defined independent of their environment.

Features

- Deploy, upgrade, and undeploy of a file based resource on a [Host](#).
 - [File](#)
 - [Folder](#)
 - [Archive](#)

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 file, folder and archive resource

```
Manifest-Version: 1.0
Deployit-Package-Format-Version: 1.3
CI-Application: FilePluginSample
CI-Version: 1.0

Name: sampleFile.txt
CI-Name: sampleFile
CI-Type: file.File

Name: sampleArchive.zip
CI-Name: sampleArchive
CI-Type: file.Archive

Name: sampleFolder
CI-Name: sampleFolder
CI-Type: file.Folder
```

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
file.File file.Archive	overthere.Host	file.DeployedFile file.DeployedArchive
file.Folder	overthere.Host	file.DeployedFolder

Deployed Actions Table

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

Deployed	Create	Destroy	Modify
file.DeployedFile file.DeployedArchive	<ul style="list-style-type: none"> • Create target path on host, if needed 	<ul style="list-style-type: none"> • Delete file from host 	<ul style="list-style-type: none"> • Delete old file from host • Copy modified file to target path on host

	<ul style="list-style-type: none"> Copy file to target path on host 		
file.DeployedFolder	<ul style="list-style-type: none"> Create target folder on host, if needed Copy folder content to target folder on host 	<ul style="list-style-type: none"> Delete folder content from host If folder is not a shared folder, the folder itself is deleted from host 	<ul style="list-style-type: none"> Perform actions as described by <i>Destroy</i> for old folder Perform actions as described by <i>Create</i> for modified folder

CI Reference

Configuration Item Overview

Deployables

CI	Description
file.Archive	An archive (zip, jar, tar, etc) artifact that can be packaged in a Deployment Package
file.File	A single file artifact that can be packaged in a Deployment Package
file.Folder	A folder artifact that can be packaged in a Deployment Package

Deployeds

CI	Description
file.DeployedArchive	The archive as deployed on the Host
file.DeployedArtifactOnHost	Abstract deployed that can target any DeployableArtifact to a Host
file.DeployedFile	The file as deployed on the Host
file.DeployedFolder	The folder as deployed on the Host

Containers

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.Host	Machine that runs middleware, on which scripts can be executed, etc
overthere.Jumpstation	Base class for jumpstations
overthere.LocalHost	Machine on which the Deployit Server is running
overthere.RemoteHost	Description unavailable
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

Other Configuration Items

CI	Description
file.Archive	An archive (zip, jar, tar, etc) artifact that can be packaged in a Deployment Package
file.DeployedArchive	The archive as deployed on the Host
file.DeployedArtifactOnHost	Abstract deployed that can target any DeployableArtifact to a Host
file.DeployedFile	The file as deployed on the Host
file.DeployedFolder	The folder as deployed on the Host
file.File	A single file artifact that can be packaged in a Deployment Package
file.Folder	A folder artifact that can be packaged in a Deployment Package
overthere.CifsHost	Machine that can be connected to using either WinRM or Telnet and can perform file manipulation via the CIFS protocol
overthere.Host	Machine that runs middleware, on which scripts can be executed, etc
overthere.Jumpstation	Base class for jumpstations
overthere.LocalHost	Machine on which the Deployit Server is running
overthere.RemoteHost	Description unavailable
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

Configuration Item Details

file.Archive

Type Hierarchy	udm.BaseDeployableArchiveArtifact >> udm.BaseDeployableFileArtifact >> udm.BaseDeployableArtifact >> udm.BaseDeployable >> udm.BaseConfigurationItem
Interfaces	udm.Taggable, udm.Deployable, udm.SourceArtifact, udm.ArchiveArtifact, udm.Artifact, udm.DeployableArtifact, udm.ConfigurationItem, udm.FileArtifact

An archive (zip, jar, tar, etc) artifact that can be packaged in a Deployment Package

Public Properties	
createTargetPath : STRING	Create the targetPath on the host if it does not exist.
excludeFileNamesRegex : STRING	Regular expression that matches file names that must be excluded from scanning
placeholders : SET_OF_STRING	Placeholders detected in this artifact
scanPlaceholders : BOOLEAN = false	Whether to scan this artifact for placeholders when it is imported
tags : SET_OF_STRING	If set, this deployable will only be mapped automatically to containers with the same tag.
targetFileName : STRING	Name of the artifact on the host.
targetPath : STRING	Path to which artifact must be copied to when being deployed.
targetPathShared : STRING	Is the targetPath shared by others on the host. When true, the targetPath is not deleted during undeployment; only the artifacts copied to it.
Hidden Properties	
* textFileNamesRegex : STRING = .+\. (cfg conf config ini properties props txt asp aspx htm html jsf jsp xht xhtml sql xml xsd xsl xslt)	Regular expression that matches file names of text files

file.DeployedArchive

Type Hierarchy	file.DeployedArtifactOnHost >> udm.BaseDeployedArtifact >> udm.BaseDeployed >> udm.BaseConfigurationItem
Interfaces	udm.Artifact, udm.Deployed, udm.ConfigurationItem, udm.DerivedArtifact

The archive as deployed on the Host.

Parent	
* container : CI <udm.Container>	The container on which this deployed runs.
Public Properties	
* targetPath : STRING	Path to which artifact must be copied to on the host.
createTargetPath : BOOLEAN = false	Create the targetPath on the host if it does not exist.
deployable : CI <udm.Deployable>	The deployable that this deployed is derived from.
placeholders : MAP_STRING_STRING	A Map containing all the placeholders mapped to their values. Special values are <ignore> or <empty>
targetFileName : STRING	Name of the artifact on the host.
targetPathShared : BOOLEAN = true	Is the targetPath shared by others on the host. When true, the targetPath is not deleted during undeployment; only the artifacts copied to it.

file.DeployedArtifactOnHost

Virtual Type

Type Hierarchy `udm.BaseDeployedArtifact >> udm.BaseDeployed >> udm.BaseConfigurationItem`

Interfaces `udm.Artifact, udm.Deployed, udm.ConfigurationItem, udm.DerivedArtifact`

Abstract deployed that can target any DeployableArtifact to a Host.

Parent	
* container : <code>CI<udm.Container></code>	The container on which this deployed runs.
Public Properties	
* targetPath : <code>STRING</code>	Path to which artifact must be copied to on the host.
createTargetPath : <code>BOOLEAN = false</code>	Create the targetPath on the host if it does not exist.
deployable : <code>CI<udm.Deployable></code>	The deployable that this deployed is derived from.
placeholders : <code>MAP_STRING_STRING</code>	A Map containing all the placeholders mapped to their values. Special values are <ignore> or <empty>
targetFileName : <code>STRING</code>	Name of the artifact on the host.
targetPathShared : <code>BOOLEAN = true</code>	Is the targetPath shared by others on the host. When true, the targetPath is not deleted during undeployment; only the artifacts copied to it.

file.DeployedFile

Type Hierarchy `file.DeployedArtifactOnHost >> udm.BaseDeployedArtifact >> udm.BaseDeployed >> udm.BaseConfigurationItem`

Interfaces `udm.Artifact, udm.Deployed, udm.ConfigurationItem, udm.DerivedArtifact`

The file as deployed on the Host.

Parent	
* container : <code>CI<udm.Container></code>	The container on which this deployed runs.
Public Properties	
* targetPath : <code>STRING</code>	Path to which artifact must be copied to on the host.
createTargetPath : <code>BOOLEAN = false</code>	Create the targetPath on the host if it does not exist.
deployable : <code>CI<udm.Deployable></code>	The deployable that this deployed is derived from.
placeholders : <code>MAP_STRING_STRING</code>	A Map containing all the placeholders mapped to their values. Special values are <ignore> or <empty>
targetFileName : <code>STRING</code>	Name of the artifact on the host.
targetPathShared : <code>BOOLEAN = true</code>	Is the targetPath shared by others on the host. When true, the targetPath is not deleted during undeployment; only the artifacts copied to it.

file.DeployedFolder

Type Hierarchy `file.DeployedArtifactOnHost >> udm.BaseDeployedArtifact >> udm.BaseDeployed >> udm.BaseConfigurationItem`

Interfaces `udm.Artifact, udm.Deployed, udm.ConfigurationItem, udm.DerivedArtifact`

The folder as deployed on the Host.

Parent	
* container : <code>CI<udm.Container></code>	The container on which this deployed runs.

Public Properties	
* targetPath : STRING	Path to which artifact must be copied to on the host.
createTargetPath : BOOLEAN = false	Create the targetPath on the host if it does not exist.
deployable : CI<udm.Deployable>	The deployable that this deployed is derived from.
placeholders : MAP_STRING_STRING	A Map containing all the placeholders mapped to their values. Special values are <ignore> or <empty>
targetPathShared : BOOLEAN = true	Is the targetPath shared by others on the host. When true, the targetPath is not deleted during undeployment; only the artifacts copied to it.
Hidden Properties	
targetFileName : STRING	Not applicable for this type.

file.File

Type Hierarchy	udm.BaseDeployableFileArtifact >> udm.BaseDeployableArtifact >> udm.BaseDeployable >> udm.BaseConfigurationItem
Interfaces	udm.Taggable, udm.Deployable, udm.SourceArtifact, udm.Artifact, udm.DeployableArtifact, udm.ConfigurationItem, udm.FileArtifact

A single file artifact that can be packaged in a Deployment Package

Public Properties	
createTargetPath : STRING	Create the targetPath on the host if it does not exist.
excludeFileNamesRegex : STRING	Regular expression that matches file names that must be excluded from scanning
placeholders : SET_OF_STRING	Placeholders detected in this artifact
scanPlaceholders : BOOLEAN = true	Whether to scan this artifact for placeholders when it is imported
tags : SET_OF_STRING	If set, this deployable will only be mapped automatically to containers with the same tag.
targetFileName : STRING	Name of the artifact on the host.
targetPath : STRING	Path to which artifact must be copied to when being deployed.
targetPathShared : STRING	Is the targetPath shared by others on the host. When true, the targetPath is not deleted during undeployment; only the artifacts copied to it.
Hidden Properties	
* textFileNamesRegex : STRING = .*\.(cfg conf config ini properties props txt asp aspx htm html jsf jsp xht xhtml sql xml xsd xsl xslt)	Regular expression that matches file names of text files

file.Folder

Type Hierarchy	udm.BaseDeployableFolderArtifact >> udm.BaseDeployableArtifact >> udm.BaseDeployable >> udm.BaseConfigurationItem
Interfaces	udm.Taggable, udm.Deployable, udm.SourceArtifact, udm.Artifact, udm.DeployableArtifact, udm.ConfigurationItem, udm.FolderArtifact

A folder artifact that can be packaged in a Deployment Package

Public Properties	
createTargetPath : STRING	Create the targetPath on the host if it does not exist.
excludeFileNamesRegex : STRING	Regular expression that matches file names that must be excluded from scanning
placeholders : SET_OF_STRING	Placeholders detected in this artifact
scanPlaceholders : BOOLEAN = true	Whether to scan this artifact for placeholders when it is imported
tags : SET_OF_STRING	If set, this deployable will only be mapped automatically to containers with the same tag.
targetFileName : STRING	Name of the artifact on the host.
targetPath : STRING	Path to which artifact must be copied to when being deployed.
targetPathShared : STRING	Is the targetPath shared by others on the host. When true, the targetPath is not deleted during undeployment; only the artifacts copied to it.
Hidden Properties	
* textFileNamesRegex : STRING = .\.(cfg conf config ini properties props txt asp aspx htm html jsf jsp xht xhtml sql xml xsd xsl xslt)	Regular expression that matches file names of text files

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

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
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:\IBM\WebSphere -> 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 certifiacte 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	
winrmKerberosAddPortToSpn : BOOLEAN = false	Add the port number (e.g. 5985) to the service principal name (SPN) for which a Kerberos ticket is requested	
winrmKerberosDebug : BOOLEAN = false	Enable Kerberos debug messages	
winrmKerberosUseHttpSpn : BOOLEAN = 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		
checkConnection	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	
* os : ENUM [WINDOWS, UNIX]	Operating system
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	
* 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

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	
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

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	
* os : ENUM [WINDOWS , UNIX]	Operating system
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	
* 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.RemoteHost

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

Description unavailable

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.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	
* 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 >	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 = <code>.*Password:[]?</code>	Regular expression to look for in keyboard-interactive authentication before sending the password
* protocol : STRING = <code>ssh</code>	Protocol
* sudoCommandPrefix : STRING = <code>sudo -u {0}</code>	Sudo command to prefix to the original command. The placeholder {0} is replaced with the sudoUsername
* sudoPasswordPromptRegex : STRING = <code>.*[Pp]assword.*:</code>	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 = <code>false</code>	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 = <code>true</code>	If true, permissions are explicitly changed with <code>chmod -R go+rX</code> after uploading a file or directory
sudoPreserveAttributesOnCopyFromTempFile : BOOLEAN = <code>true</code>	If true, files are copied from the connection temporary directory using the <code>-p</code> flag to the <code>cp</code> command
sudoPreserveAttributesOnCopyToTempFile : BOOLEAN = <code>true</code>	If true, files are copied to the connection temporary directory using the <code>-p</code> flag to the <code>cp</code> command
sudoQuoteCommand : BOOLEAN = <code>false</code>	If true, the original command is quoted when it is prefixed with sudoCommandPrefix
tmpDeleteOnDisconnect : BOOLEAN = <code>true</code>	Whether to delete the temporary connection directory when the connection is closed

Control Tasks	
checkConnection	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	
* 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