

Vagrant Repositories

Overview

On top of [general support](#) for advanced artifact management, Artifactory support for [Vagrant](#) provides:

1. Distribution and sharing of Vagrant boxes within your organization.
2. Calculation of Metadata for Vagrant boxes hosted in Artifactory's local repositories
3. Extensive security features that give you fine-grained access control over boxes.
4. Support for flexible repository layouts that allow you to organize your boxes and assign access privileges according to projects or development teams.
5. Smart searches for boxes.

Configuration

Local Repositories

To create a local Vagrant repository to host your Vagrant boxes, from the **Administration** module go to **Repositories | Repositories | Local** and create a new Local Repository and set **Vagrant** as the **Package Type**.

New Local Repository

Basic

Package Type *


VAGRANT
Vagrant

Repository Key *

Page Contents

- [Overview](#)
- [Configuration](#)
 - [Local Repositories](#)
- [Deploying Vagrant Boxes](#)
 - [Deploying a package using the UI](#)
 - [Set Me Up](#)
 - [Deploying a package using Matrix Parameters](#)
 - [Setting the Target Path](#)
- [Provisioning Vagrant Boxes](#)
- [Authenticated Access to Servers](#)
- [Watch the Screencast](#)

Deploying Vagrant Boxes

Deploying a package using the UI

To deploy a Vagrant box to Artifactory, in the **Application** module, under **Artifactory | Artifacts** select the repository to which you want to deploy your Vagrant box and click **Deploy**.

The **Deploy** dialog is displayed with your selected repository as the **Target Repository** and a default **Target path**.

Deploy



Target Repository

vagrant-local

Package Type:  Vagrant

Repository Layout:

[orgPath]/[module]/[module]-[baseRev].[ext]

Single

Multi



Drop file here or Select file

Target Path 

Deploy

You can add properties you wish to attach to your box as parameters to the target path.

For example, to upload the box **precise64-virtualbox-1.0.0.box**, and specify that its name is **precise64**, with a provider of **virtualbox** and the version is **1.0.0**, you would enter:

Specifying the Target Path

```
/precise64-virtualbox-1.0.0.box;box_name=precise64;box_provider=virtualbox;box_version=1.0.0
```

Set Me Up

You can also select your repository and click **Set Me Up** to view the cURL command you can use to upload your box.

Set Me Up ✕

Tool

Repository

🔒 Type password to insert your credentials to the code snippets



Deploy

To deploy Vagrant boxes to this Artifactory repository using an explicit URL with Matrix Parameters use:

```
1 curl -u<USERNAME>:<PASSWORD> -T <PATH_TO_FILE> "http://10.70.30.65:8081/artifactory/vagrant-local/{vagrantBoxName.box};box_name={name};box_provider={provider};box_version={version}"
```

Resolve

To provision a Vagrant box, all you need is to construct it's name in the following manner.

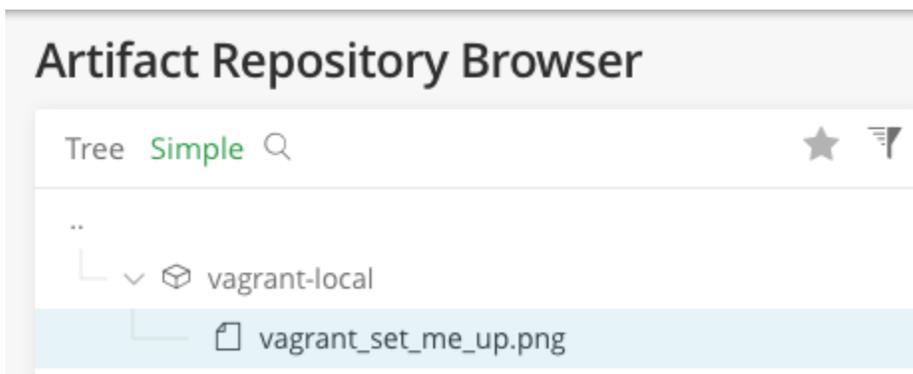
```
1 vagrant box add "http://10.70.30.65:8081/artifactory/api/vagrant/vagrant-local/{boxName}"
```



Be careful with spaces

Make sure you don't enter any superfluous spaces in the Target Path specification.

Once you have deployed your Vagrant box, and Artifactory has recalculated the repository index, your repository should be organized as displayed below:



Deploying a package using Matrix Parameters

You can also deploy Vagrant boxes to Artifactory with an explicit URL using [Matrix Parameters](#).

The URL is built similarly to the [Target Path](#) format as follows:

Deploying a package using Matrix Parameters

```
PUT "http://{Artifactory URL}/{vagrantRepoKey}/{vagrantBoxName.box};box_name={name};box_provider={provider};box_version={version}"
```

For example, to upload the box **precise64-virtualbox-1.0.0.box**, and specify that it's name is **precise64**, with a provider of **virtualbox** and the version is **1.0.0**, you would enter:

Example

```
PUT "http://localhost:8080/artifactory/vagrant-local/precise64-virtualbox-1.0.0.box;box_name=precise64;box_provider=virtualbox;box_version=1.0.0"
```

Setting the Target Path

The **Target Path** can be anywhere in the repository, but it has to contain the 3 mandatory matrix parameters: **box_name**, **box_provider** and **box_version** and the file name must end with **.box**. The format is as follows:

Target Path Format

```
PUT "http://{Artifactory URL}/{vagrantRepoKey}/{path/to/vagrantBoxName.box};box_name=[name];box_provider=[provider];box_version=[version]"
```

name	The value to assign to the <code>box_name</code> property used to specify the Vagrant box name.
provider	The value to assign to the <code>box_provider</code> property used to specify the Vagrant box provider (virtualbox/lxc or others).
version	The value to assign to the <code>box_version</code> property used to specify the Vagrant box version (must comply with Vagrant's versioning schema)

Provisioning Vagrant Boxes

Vagrant boxes are available through the following URL:

Vagrant box URL

```
vagrant box add "http://{Artifactory URL}/api/vagrant/{vagrantRepoKey}/{boxName}"
```



Specifying the path to the box

With Vagrant client commands, make sure you don't specify the path to a box in the command. The path should be specified using properties.

For example, to provision a Vagrant box called **precise64** from a repository called **vagrant-local**, you would construct it's name in the following manner:

Provisioning a Vagrant box

```
vagrant box add "http://localhost:8080/artifactory/api/vagrant/vagrant-local/precise64"
```

You can select the repository from which you want to provision your box, and click [Set Me Up](#) to get the specific URL for the selected repository.

You can also, optionally, pass parameters to specify a specific box version or provider. For example:

Provisioning a Vagrant box by version

```
vagrant box add "http://localhost:8080/artifactory/api/vagrant/vagrant-local/precise64 --provider virtualbox --box-version 1.0.0"
```

In addition, boxes can be provisioned using properties; this is useful when you want to download the latest box tagged by a specific property. The properties query parameter value should comply with [Using Properties in Deployment and Resolution](#).

Examples:

Provisioning a Vagrant box by version

```
vagrant box add "http://localhost:8080/artifactory/api/vagrant/vagrant-local/precise64?properties=box_version%2B=3.0.0"
```

Note the '%2B' [encoding](#) on the command for the '+' symbol (which is for [Mandatory](#) properties: key+=value)

The following example downloads a box with `box_name=trusty64`, `box_version=3.0.0` from `path="folder"`.

It uses an optional "path" property (in addition to the mandatory properties) to specify the path where the box is stored in Artifactory. We will use this property for resolution of the box.

```
vagrant box add "http://localhost:8080/artifactory/api/vagrant/vagrant-local/trusty64?properties=box_version%2B=3.0.0;path%2B=folder"
```

Note the format for resolution of multiple properties: `key1+=value1;key2+=value2....`

4 Properties

 Delete < Page of 1 >

<input checked="" type="checkbox"/>	Property	Value(s)
<input type="checkbox"/>	box_name	trusty64
<input type="checkbox"/>	box_version	3.0.0
<input type="checkbox"/>	box_provider	aws
<input type="checkbox"/>	path	folder

Authenticated Access to Servers

If you need to access a secured Artifactory server that requires a username and password, you need to specify 2 environment variables:

1. ATLAS_TOKEN

- This token is a [Base64](#) encoded string of the user credentials (formatted `username:password`).
- It can be generated by any user from Artifactory with the following command:

```
$curl -uadmin:password "http://localhost:8080/artifactory/api/vagrant/auth"  
YWRtaW46QVZzWGhzWmlDU29NVmtaQ2dCZEY3XXXXXXXX
```

- It is not related to the user's access token, API key or encrypted password
- It is only used for downloading Vagrant boxes from private repositories in Artifactory
- The token will change if the user's password changes
- For more details on the ATLAS_TOKEN, please refer to the [Vagrant documentation](#).

2. VAGRANT_SERVER_URL - The base URL for the Artifactory server.

Setting ATLAS_TOKEN and VAGRANT_SERVER_URL

```
export ATLAS_TOKEN={token}
export VAGRANT_SERVER_URL=http://{Artifactory URL}/api/vagrant/{vagrantRepoKey}
For example:
export ATLAS_TOKEN=YWRtaW46QVAzWGhzWmlDU29NVmtaQ2dCZEY3XXXXXXXXX
export VAGRANT_SERVER_URL=http://localhost:8081/api/vagrant/vagrant-local
```



Both environment variables are required

When using Artifactory with authenticated access (i.e. anonymous access is disabled), both of these environment variables are required. If either of them is not set, the Vagrant client will not be able to access Artifactory and a 401 error message will be generated.

Watch the Screencast