

## 目录

目录	1
安装配置Terraform	2
Terraform介绍	2
1. 下载Terraform	2
2. 配置环境变量	2
3. 安装金山云插件	2
4. 配置金山云provider文件	3

# 安装配置Terraform

## Terraform介绍

Terraform是一种开源工具，用于安全高效地预配和管理云基础设施，支持多家云服务提供商。以代码的形式将所要管理的资源定义在模板中，通过解析并执行模板来自动化完成所定义资源的创建，变更和管理，进而达到自动化运维的目标。

本项目已经发布在[Github](#)，如本文中覆盖不到的示例可以参考github example内容。

注意： terraform v0.12 版本的配置文件与v0.11 版本的配置文件格式不同。该example下的配置文件是基于v0.11.13开发的。若想使用v0.12版本的terraform需对配置文件进行修改，该修改不需手动修改，terraform支持自动修改。可在配置文件目录下直接执行：

```
# terraform 0.12upgrade
```

## 1. 下载Terraform

官方提供了最新版本的[Terraform](#)，用户可以选择适合自己开发环境的下载包。若要安装其它的Terraform版本，请自行更改下载链接。

本文以Linux系统为例，输入下载安装命令。

```
// 下载Terraform
wget https://releases.hashicorp.com/terraform/0.12.5/terraform_0.12.5_linux_amd64.zip
```

```
// 安装Terraform
# unzip terraform_0.12.5_linux_amd64.zip
```

## 2. 配置环境变量

新建目录downloads，将安装好的terraform文件保存在该目录下。

```
// Move terraform
# mkdir downloads
# mv terraform downloads/
```

进入配置文件~/.bash\_profile添加Terraform的环境变量。

```
# vim ~/.bash_profile
# export PATH="$PATH:~/downloads"
# export TF_LOG=DEBUG
# export TF_LOG_PATH=/var/log/terraform.log //打印日志信息，方便问题定位
# export OS_DEBUG=1
```

重新加载~/.bash\_profile文件

```
# source ~/.bash_profile
```

检查Terraform配置完成，若显示可用的Terraform选项列表，表示安装配置完成。

```
#terraform
```

## 3. 安装金山云插件

当前金山云部分产品线Terraform providers已经开发完成，且正在与Terraform官方进行认证，认证完成以前需要用户手动完成插件安装配置。

插件安装支持多平台，包括linux、windows和mac，不同平台的安装包下载链接如下。

[下载](#)

- linux、mac平台，下载后将其拷贝到terraform的plugins目录（~/terraform.d/plugins）即可完成配置工作（需要解压缩）。

注意：首次安装需要在~/terraform.d 目录下手动创建plugins文件夹。

- Windows平台，下载后将其拷贝到terraform的plugins目录（%APPDATA%\terraform.d\plugins）即可完成配置工作。

本文以linux平台为例，

```
cd ~/.terraform.d
mkdir plugins
```

解压安装包完成配置。

```
# cd ~/.terraform.d/plugins
# tar -xzf terraform-provider-ksyun_linux-amd64.tgz
```

## 4. 配置金山云provider文件

**【Provider】** 是一个与Open API直接交互的后端驱动，Terraform就是通过Provider来完成对基础设施资源的管理的。不同的基础设施提供商都需要提供一个Provider来实现对自家基础设施的统一管理。

登录控制台获取账号ak/sk密钥。

因为每个Terraform项目都需要一个独立的执行目录，所以我们可以先创建一个新的目录，目录名称Terraform-ksyun。

```
# mkdir Terraform-ksyun
# cd Terraform-ksyun
```

进入新建目录，创建provider配置文件。Terraform在运行时，会读取该目录空间下所有.tf和.tfvars文件。因此，您可以按照实际用途将配置信息写入到不同的文件中。

创建 provider.tf 文件，需要按以下格式配置您的身份认证信息：

```
# vim provider.tf

provider "ksyun" {
  access_key = "AKLT7Nj7p*****"
  secret_key = "0Cd8XJrX+*****"
  region = "cn-beijing-6" //机房信息
}
```

初始化工作目录，检查是否配置正确。

```
# terraform init
```