

## Ansible: instalación de git en un servidor remoto

Ansible es una herramienta que permite administrar de manera masiva todos nuestros servidores remotos. La historia de Ansible y su *Colectivo* o *Galaxia* (comunidad de usuarios) está [muy bien explicado en este artículo](#): esencialmente podemos contribuir de manera sistematizada a compartir nuestro trabajo con otros usuarios y viceversa. *Esto es poderoso, sí.* Es muy bueno, sin embargo nuestras necesidades específicas pueden ser muy singulares: por ejemplo [instalar un programa como Pandora FMS, su Agente Software](#) (el artículo tiene un ejemplo práctico para ello, incluidos **curl** y **figlet**). En esta entrada damos la alta y bienvenida a nuestros propios guiones Ansible para poder cumplir [con la premisa del administrador de sistemas perezoso](#): hacer un esfuerzo inicial para escribir guiones para tareas repetitivas. **Esta entrada va de eso.**

Para evitar el reinventar la rueda, hemos hecho una bifurcación del repositorio de [libretos Ansible de la Comunidad de la empresa DigitalOcean](#). Está regida por la licencia del MIT y fue creado para las versión LTS 18.04 de Ubuntu, *pero acá está nuestra contribución*: la traducción al castellano, probar y comprobar esos guiones y agregar nuevo material e incluir más aplicaciones y distribuciones GNU/Linux.

Al momento de escribir estas líneas estamos bien alojados (relación entre precio y servicio) con DigitalOcean pero también existen otros proveedores como [UpCloud](#) y [Linode](#) los cuales estamos evaluando. **Mientras tanto estimadas y estimados lectores, pueden crear una cuenta nueva en DigitalOcean con nuestro código de referido para que tengan un crédito de USD 100\$ ¡y poder así practicar con todos estos artículos!**

## Estilo de repositorio

El estilo de repositorio -aparte de las traducciones correspondientes- lo pueden observar -y utilizar- en el capítulo de instalación de [Docker y Docker-compose](#). Justo ahora al escribir esto, solo tenemos una variable en el archivo **vars.yml** que contiene el nombre del usuario que va a utilizar el docker-compose (o la aplicación deseada, aquí en esta entrada **git**). Para nosotros siempre será "ks7000", para ustedes coloquen vuestro nombre con el cual administran los servidores remotos.

## Artículos consultados

Acostumbramos colocar al final de la entrada la bibliografía y/o fuentes consultadas, sin embargo como el 90% de todo esto es un trabajo original (en idioma inglés) y un 10% de personalización es nuestro deber colocarlos por delante:

- Los conceptos básicos para escribir y administrar libretos Ansible:
  - <https://www.digitalocean.com/community/tutorials/configuration-management-101-writing-ansible-playbooks> .
- De suma importancia la correcta configuración del servidor remoto para recibir Ansible, la sra. [Erika Heidi](#) con una simplicidad admirable lo enseña en su blog:
  - <http://www.erikaheidi.com/blog/configuring-ssh-server-access-for-ansible/> .
- Para instalar el legendario servidor web Apache en Ubuntu 14:
  - <https://www.digitalocean.com/community/tutorials/how-to-configure-apache-using-ansible-on-ubuntu-14-04> .
- Drupal es un manejador de contenido, una alternativa a WordPress, es complejo ya que necesita varios componentes tales como base de datos, servidor web y entorno de programación en Ubuntu 14:
  - <https://www.digitalocean.com/community/tutorials/how-to-create-an-ansible-playbook-to-automate-drupal-installation-on-ubuntu-14-04> .

Este último artículo presenta un estilo de repositorio más complejo, con el uso intensivo de variables, lo dejamos a futuro para tocar instalaciones parecidas, que contengan sistemas completos.

## Instalando git y figlet

Agregamos a **figlet**, un sencillo programa al cual le podemos pasara cualquier texto y lo mostrará desplegado como una pancarta (a esto creo que le llaman "Arte ASCII").

## Libreto Ansible para git y figlet

Destacamos el uso de una lista de renglones con la instrucción **with\_items**:

```
- hosts: all    become: yes    vars_files:    - vars.yml    gather_facts: f
alse    tasks:    - name: Instalación de git y figlet con apt en Ubuntu 20.
04 .    apt:    name: "{{ item }}"    state: present    update_
cache: yes    with_items:    - git    - figlet    tags:    - g
it    - figlet    - name: Agregando usuario al grupo "git" .    user:
name: "{{ ans_usuario }}"    group: "git"    append: yes
tags:    - git
```

Observen que también le ordenamos a **apt** que actualice el catálogo de los repositorios con la instrucción **update\_cache: yes** .

Y el comando para instalar (fíjense en cuál directorio estén ubicados, el indicador de comando muestra la ruta):

```
ansible-playbook playbook.yml -l flor.fcia.com.ve -u ks7000
```

A futuro le agregaremos otras opciones como el verificar si la distro es Debian o sucedánea. Para ello es útil la instrucción **gather\_facts** que por consumir más tiempo de ejecución debemos buscar la manera de "preguntar" de dicho estado una sola vez y almacenar dichos valores en nuestro nodo controlador.

Noten también que usamos el parámetro **-l** para especificar un solo servidor remoto, sino Ansible utilizaría el valor almacenado en **/etc/ansible/hosts** . Con el parámetro **-u** indicamos con cuál usuario conectamos con el servidor remoto, esto es independiente del usuario que usamos en **vars.yml** .

En la imagen anterior mostramos nuestro resultado práctico , tenemos una advertencia y un error.

- La advertencia es el ciclo que usamos para instalar **git** y **figlet** ya que a partir de la versión Ansible 2.11 será desactivada para adaptarse al uso de listas en Python (hagan clic en la imagen anterior para ver de manera ampliada en una ventana nueva).
- El error es que no existe el grupo **git**, ese es el usuario que me refiero en el archivo **vars.yml**.

Aunque también podemos hacer un guión para mostrar la versión **git** que fue instalada, simplemente colocamos la captura de conexión normal:

git version (salida para figlet)