



Administración de Jenkins

Instalación

- Instalar minikube
- minikube --memory 4096 --cpus 2 start --vm-driver=virtualbox -p jenkins
- minikube status
- minikube dashboard
- brew install kubectl
- brew install kubernetes-helm
- kubectl get ns
- kubectl create -f minikube/jenkins-namespace.yaml
- kubectl create -f minikube/jenkins-volume.yaml
- minikube mount ./data:/data/jenkins-volume
- helm init
- helm repo add jenkins https://charts.jenkins.io
- helm repo update
- helm search repo jenkins
- helm pull jenkins/jenkins

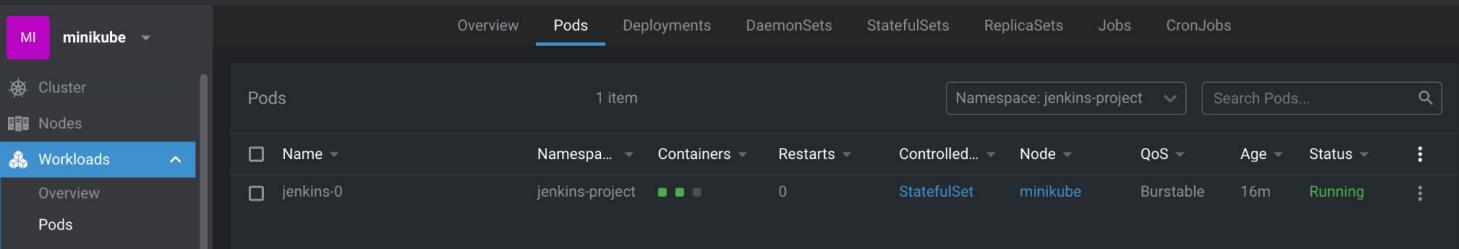
Instalación (II)

- helm show values jenkins/jenkins > values.yaml
 - serviceType: ClusterIP → NodePort (línea 129)
 - nodePort: 32000 (descomentar línea ~146)
 - volumes [] →
 - volumes: (línea ~647)
 - type: HostPath
 - hostPath: /var/run/docker.sock
 - mountPath: /var/run/docker.sock
 - storageClass: → jenkins-pv (línea 818)
- helm upgrade --install jenkins jenkins/jenkins -n jenkins -f helm/jenkins-values.yaml --namespace jenkins-project
- helm list --namespace jenkins-project

OpenLens

Instalar OpenLens

<https://github.com/MuhammedKalkan/OpenLens/releases>



The screenshot displays the OpenLens application interface for managing Kubernetes workloads. The left sidebar includes navigation links for 'Cluster', 'Nodes', and 'Workloads' (which is currently active), along with 'Overview' and 'Pods'. The main content area is titled 'Pods' and shows a single item: 'jenkins-0'. The pod details are as follows:

Name	Namespace	Containers	Restarts	Controlled By	Node	QoS	Age	Status
jenkins-0	jenkins-project	3	0	StatefulSet	minikube	Burstable	16m	Running

Desinstalar Jenkins

- helm uninstall jenkins --namespace=jenkins-project
- kubectl delete pv jenkins-pv --grace-period=0 --force --namespace=jenkins-project

Troubleshooting

- Los pods no resuelven dns (<https://github.com/kubernetes/minikube/issues/1340>)

```
$kubectl edit configmap coredns -n kube-system
```

Cambiar

```
forward . /etc/resolv.conf {  
    max_concurrent 1000  
}
```

Por

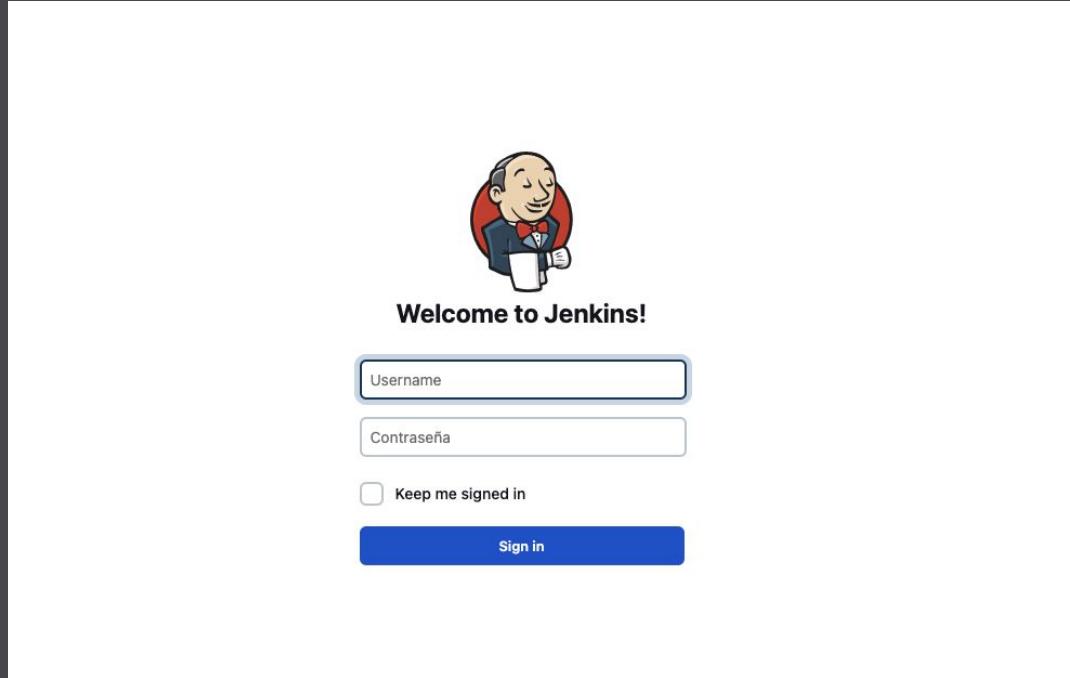
```
forward . 8.8.8.8 8.8.4.4
```

Troubleshooting

The screenshot shows the MiKube UI interface for troubleshooting. The left sidebar has a 'Config' section with 'ConfigMaps' selected. The main area is titled 'ConfigMaps' and shows a list of 6 items in the 'kube-system' namespace. Each item has columns for Name, Namespace, Keys, Age, and three-dot options. The first item, 'coredns', has an 'Edit' button highlighted with a red box.

Name	Namespace	Keys	Age	Actions
coredns	kube-system	Corefile	19d	Edit
extension-apiserver-authentication	kube-system	client-ca-file, requestheader-allowed-names, requestheader... ...ntermediate-ca-file	19d	Delete
kube-proxy	kube-system	config.conf, kubeconfig.conf	19d	⋮
kube-root-ca.crt	kube-system	ca.crt	19d	⋮
kubeadm-config	kube-system	ClusterConfiguration	19d	⋮
kubelet-config	kube-system	kubelet	19d	⋮

Instalación



Credenciales

- Get your 'admin' user password by running:

```
kubectl exec --namespace jenkins-project -it svc/jenkins -c jenkins -- /bin/cat  
/run/secrets/additional/chart-admin-password && echo
```

- Get the Jenkins URL to visit by running these commands in the same shell:

```
export NODE_PORT=$(kubectl get --namespace jenkins-project -o jsonpath=".spec.ports[0].nodePort"  
services jenkins)
```

```
export NODE_IP=$(kubectl get nodes --namespace jenkins-project -o  
jsonpath=".items[0].status.addresses[0].address")
```

```
echo http://$NODE_IP:$NODE_PORT/login
```

- Login with the password from step 1 and the username: admin

Panel Control

Panel de Control > Administrar Jenkins

+ Nueva Tarea

Personas

Historial de trabajos

Administrador Jenkins

Mis vistas

Trabajos en la cola

No hay trabajos en la cola

Estado del ejecutor de construcciones

Administrar Jenkins

Parece que la configuración de proxy inverso está mal

[Más información](#)

[Ocultar](#)

System Configuration



Configurar el Sistema

Configurar variables globales y rutas.



Global Tool Configuration

Configure tools, their locations and automatic installers.



Administrar Plugins

Añadir, borrar, desactivar y activar plugins que extienden la funcionalidad de Jenkins.



Administrar Nodos

Añadir, borrar, gestionar y monitorizar los nodos sobre los que Jenkins ejecuta tareas.



Configuration as Code

Reload your configuration or update configuration source

Panel Control

Tools and Actions



Actualizar configuración desde el disco duro.

Descartar todos los datos cargados en memoria y actualizar todo nuevamente desde los ficheros del sistema. Útil cuando se modifican ficheros de configuración directamente en el disco duro.



Consola de scripts

Ejecutar script para la administración, diagnóstico y solución de problemas.



Jenkins CLI

Accede y administra Jenkins desde la consola, o desde scripts



Preparar Jenkins para apagar el contenedor

Detener la ejecución de nuevas tareas para que el sistema pueda apagarse de manera segura.

Jenkins 2.361.3

Usuario Jenkins

↑ Volver al Panel de control

⚙️ Administrar Jenkins

👤 Users

+ Crear un usuario

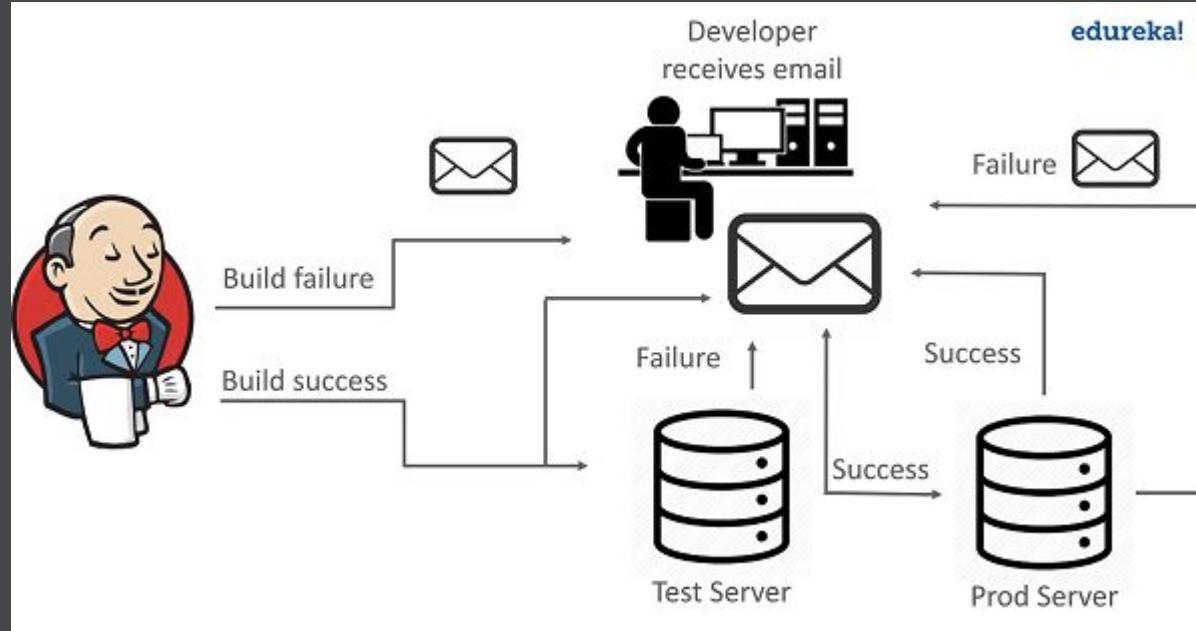
Usuarios

Estos usuarios pueden entrar en Jenkins. Este es un subconjunto de [esta lista](#), que tambien incluyen usuarios creados automáticamente porque hayan hecho 'commits' a proyectos. Los usuarios creados automáticamente no tienen acceso directo a Jenkins.

User ID ↓	Nombre	
admin	Jenkins Admin	
miguel	Miguel	

Email notification

- Jenkins proporciona un informe enviado por email donde se indicará el estado de finalización del job



Email notification

Notificación por correo electrónico

Servidor de correo saliente (SMTP)

Sufijo de email por defecto [?](#)

Use SMTP Authentication [?](#)

Usar seguridad SSL [?](#)

Usar seguridad TLS (STARTTLS)

Puerto de SMTP [?](#)

Dirección para la respuesta

Plugin: email-ext ([Email Extension | Jenkins plugin](#))

- Añade versatilidad y capacidad para enviar emails desde cualquier pipeline.



The screenshot shows the Jenkins Manage Plugins interface. The top navigation bar has tabs for 'Updates', 'Available', 'Installed' (which is selected and highlighted in dark blue), and 'Advanced'. A search bar at the top right is labeled 'Filter:' with a magnifying glass icon and contains the text 'email'. Below the tabs is a table with the following columns: 'Enabled', 'Name', 'Version', 'Previously installed version', and 'Uninstall'. The table lists one plugin:

Enabled	Name	Version	Previously installed version	Uninstall
<input checked="" type="checkbox"/>	Email Extension Plugin	2.68		Uninstall

The 'Email Extension Plugin' row contains a detailed description: 'This plugin is a replacement for Jenkins's email publisher. It allows to configure every aspect of email notifications: when an email is sent, who should receive it and what the email says.'

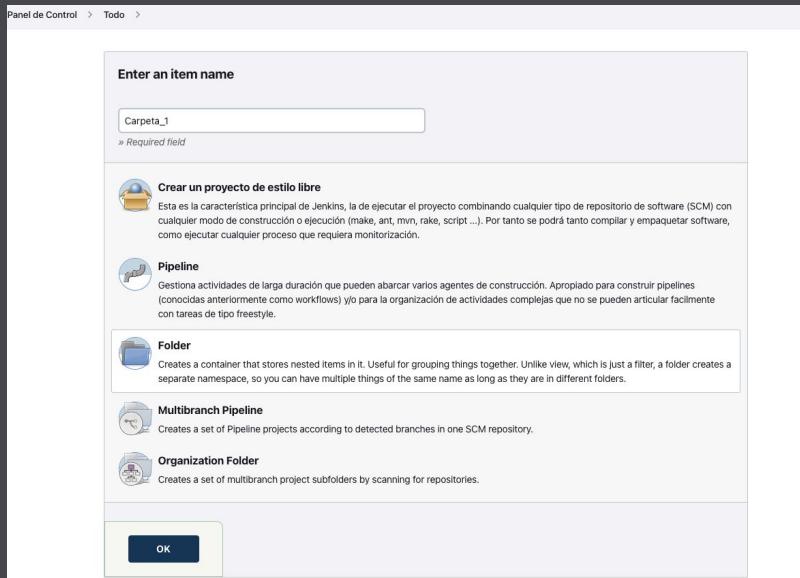
Plugin: email-ext

```
agent any
stages {
    stage('Ok') {
        steps {
            echo "Ok"
        }
    }
}
post {
    always {
        emailext body: 'A Test EMail', recipientProviders: [[${class: 'DevelopersRecipientProvider'}, ${class: 'RequesterRecipientProvider'}]], subject: 'Test'
    }
}
}
```

- recipientProviders: añadimos destinatarios adicionales. Por ejemplo para incluir a los desarrolladores implicados en los cambios del repositorio y a la persona que inició el job.

Plugin: folders (<https://docs.cloudbees.com/docs/admin-resources/latest/plugins/folder>)

- Permite añadir carpetas a nuestro Jenkins, con el fin de organizar los proyectos
- Genera una estructura de carpetas y subcarpetas



Credenciales

- La gran mayoría de los pipelines que ejecutamos requieren de algún tipo de credencial para poder hacer su trabajo.
- Todos los secretos deben quedar fuera del repositorio de código, por tanto, se necesita de un lugar donde guardar todos los secretos.
- Jenkins proporciona un sistema de almacenamiento de credenciales. No es seguro. Cualquiera que tenga acceso a Jenkins podría tener acceso a todas las credenciales guardadas.
- No se pueden usar algoritmos de encriptación one-way como bcrypt, ya que los pipelines necesitan obtener y utilizar la clave desencriptada. Por tanto, Jenkins debe encriptar las credenciales y asegurarse que será capaz de desencriptarlas en el futuro. Guarda la clave de encriptado en su sistema.
- ```
println
hudson.util.Secret.decrypt(" {AQAAABAAAAAgPT7JbBVgyWiivobt0CJEduLyP0lB3uyTj+D5WBvVk6jyG6BQFPYGN4Z3V
JN2JLDm} ")
```

# Credenciales

---

Jenkins guarda la información relativa a las credenciales en los siguiente ficheros:

```
$JENKINS_HOME/credentials.xml (incluye la lista de credenciales existentes)
$JENKINS_HOME/secrets/master.key (Clave que desencripta el fichero hudson.util.Secret)
$JENKINS_HOME/secrets/hudson.util.Secret (desencripta el fichero credentials.xml)
```

# Credenciales

Jenkins 2 search ?

Jenkins > Credentials > System > Global credentials (unrestricted) >

Back to credential domains Add Credentials

Kind: Username with password

Username with password

- Docker Host Certificate Authentication
- SSH Username with private key
- Secret file
- Secret text
- Certificate
- Password

ID:

Description:

OK

A screenshot of the Jenkins Global credentials configuration interface. The top navigation bar shows 'Jenkins' with a count of 2 items, a search bar, and a help icon. Below it, the breadcrumb path is 'Jenkins > Credentials > System > Global credentials (unrestricted)'. On the left, there are links for 'Back to credential domains' and 'Add Credentials'. The main area has a 'Kind' dropdown set to 'Username with password', which is highlighted with a blue selection bar and has a red circle around its dropdown arrow. Below the dropdown are several other credential types: Docker Host Certificate Authentication, SSH Username with private key, Secret file, Secret text, Certificate, and Password. There are also fields for 'ID' and 'Description' with help icons, and a large 'OK' button at the bottom.

# Credenciales: username/password

```
stage('usernamePassword') {
 steps {
 script {
 withCredentials[
 usernamePassword(credentialsId: 'gitlab',
 usernameVariable: 'username',
 passwordVariable: 'password')
]) {
 print 'username=' + username + 'password=' + password

 print 'username.collect { it }=' + username.collect { it }
 print 'password.collect { it }=' + password.collect { it }
 }
 }
 }
}
```

## Credenciales: secret

```
stage('string (secret text)') {
 steps {
 script {
 withCredentials[
 string(
 credentialsId: 'joke-of-the-day',
 variable: 'joke')
]) {
 print 'joke=' + joke
 print 'joke.collect { it }=' + joke.collect { it }
 }
 }
 }
}
```

## Credenciales: secret

```
stage('sshUserPrivateKey') {
 steps {
 script {
 withCredentials[
 sshUserPrivateKey(
 credentialsId: 'production-bastion',
 keyFileVariable: 'keyFile',
 passphraseVariable: 'passphrase',
 usernameVariable: 'username')
]) {
 print 'keyFile=' + keyFile
 print 'passphrase=' + passphrase
 print 'username=' + username
 print 'keyFile.collect { it }=' + keyFile.collect { it }
 print 'passphrase.collect { it }=' + passphrase.collect { it }
 print 'username.collect { it }=' + username.collect { it }
 print 'keyFileContent=' + readFile(keyFile)
 }
 }
 }
}
```

## Credenciales: Buenas prácticas

---

- Mantener actualizado Jenkins. :D
- Seguir el principio de "Menor privilegio". ReadOnly / ReadWrite
- Limitar el acceso. Cuando un pipeline solo necesite un subconjunto de credenciales, crear las credenciales dentro del Folder/Carpeta. El plugin Folders nos ayuda en este sentido.
- Intentar evitar el uso de credenciales. Los proveedores cloud permiten el acceso a los recursos asignando un rol o identidad (Azure Managed Identity) a una máquina (VM, Cluster Kubernetes, SQL Server, ...).
- Rotar los secretos. Los proveedores cloud proporcionan servicios para albergar secretos. Estos servicios son capaces de rotar las claves automáticamente cada cierto tiempo. Los secretos podrían filtrarse igual pero tendrían una vigencia limitada.

## Plugins: Azure Key Vault (<https://plugins.jenkins.io/azure-keyvault/>)

Este plugin recupera los secretos existentes en un Azure Key Vault. Una vez finalizada la configuración, los secretos aparecerán en nuestro Jenkins, indicando mediante un icono diferente, que se han recuperado de un Key Vault de Azure.

The screenshot shows the Jenkins Plugin Manager interface. At the top, there are tabs: 'Actualizaciones disponibles' (Updates available), 'Todos los plugins' (All plugins), 'Plugins instalados' (Installed plugins), and 'Configuracion avanzada' (Advanced configuration). Below the tabs is a search bar containing the text 'azure key'. The main area displays a list of plugins. One plugin is highlighted with a checkmark icon: 'Azure Key Vault 146.v0df546b\_c98a\_5'. To the right of the plugin name are the labels 'Released' and 'Hace 23 días' (23 days ago). Below the plugin name, its description is visible: 'Azure Key Vault integration, provides a build wrapper, declarative pipeline step, credential provider and configuration-as-code integration.'

# Plugins: Azure Key Vault

Panel de Control > Administrar Jenkins > Credentials

## Credentials

| T | P Store ↓ | Domain | ID |
|---|-----------|--------|----|
|---|-----------|--------|----|

**Stores scoped to Jenkins**

| P Store ↓                                                                                | Domains  |
|------------------------------------------------------------------------------------------|----------|
|  Azure  | (global) |
|  System | (global) |

# Credenciales

Panel de Control > Administrar Jenkins > Configurar el sistema >

## Azure Key Vault Plugin

Key Vault URL [?](#)

Credential ID [?](#)

- none -

▼

[+ Add](#)

[Test Connection](#)

[Reload cache](#)

# Plugins: Azure Key Vault

```
pipeline {
 agent any
 stages {
 stage('Secret') {
 options {
 azureKeyVault([[envVariable: 'MY_SECRET', name: 'my-secret', secretType: 'Secret']])
 }
 steps {
 sh "echo $SECRET"
 }
 }
 stage('Certificate') {
 options {
 azureKeyVault([[envVariable: 'CERT_LOCATION', name: 'my-cert-name', secretType: 'Certificate']])
 }
 steps {
 sh "openssl pkcs12 -in $CERT_LOCATION -nodes -password 'pass:' -out keyvault.pem"
 }
 }
 }
}
```

## **Plugins: Azure Key Vault**

- Este tipo de credenciales son Read-Only. Los metadatos se sincronizan cada ~10min.
- Los secretos que se creen en el KV no aparecerán inmediatamente.
- Podemos forzar el refresco de la caché desde el panel de configuración del sistema.

## Plugins: Azure AD (<https://plugins.jenkins.io/azure-ad/>)

- Autenticación basada en Azure Active Directory
- Autorización basada en Grupos/Roles de Active Directory
- Existen alternativas para utilizar LDAP (<https://plugins.jenkins.io/active-directory/>)
- 

The screenshot shows the Jenkins Plugin Manager interface. At the top, there are tabs: 'Actualizaciones disponibles', 'Todos los plugins', 'Plugins instalados' (which is highlighted in blue), and 'Configuracion avanzada'. Below the tabs is a search bar containing the text 'active directory'. The main list area has two columns: 'Nombre' (sorted by name) and 'Activados' (status). A single plugin entry is shown:

| Nombre                                                                                                                                                | Activados                                                            |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| Azure AD Plugin 267.v5b_dfb_514d9fd<br>A Jenkins authentication & authorization plugin for Azure Active Directory<br>Report an issue with this plugin | <input checked="" type="checkbox"/> <input type="button" value="X"/> |

# Plugins: Azure AD (<https://plugins.jenkins.io/azure-ad/>)

## Configuración conexión AD

Panel de Control > Administrar Jenkins > Configuración global de la seguridad

Seguridad

Azure Active Directory

Client ID ?  
22f1d1a3-24ff-4231-b44b-8e1497e98b39

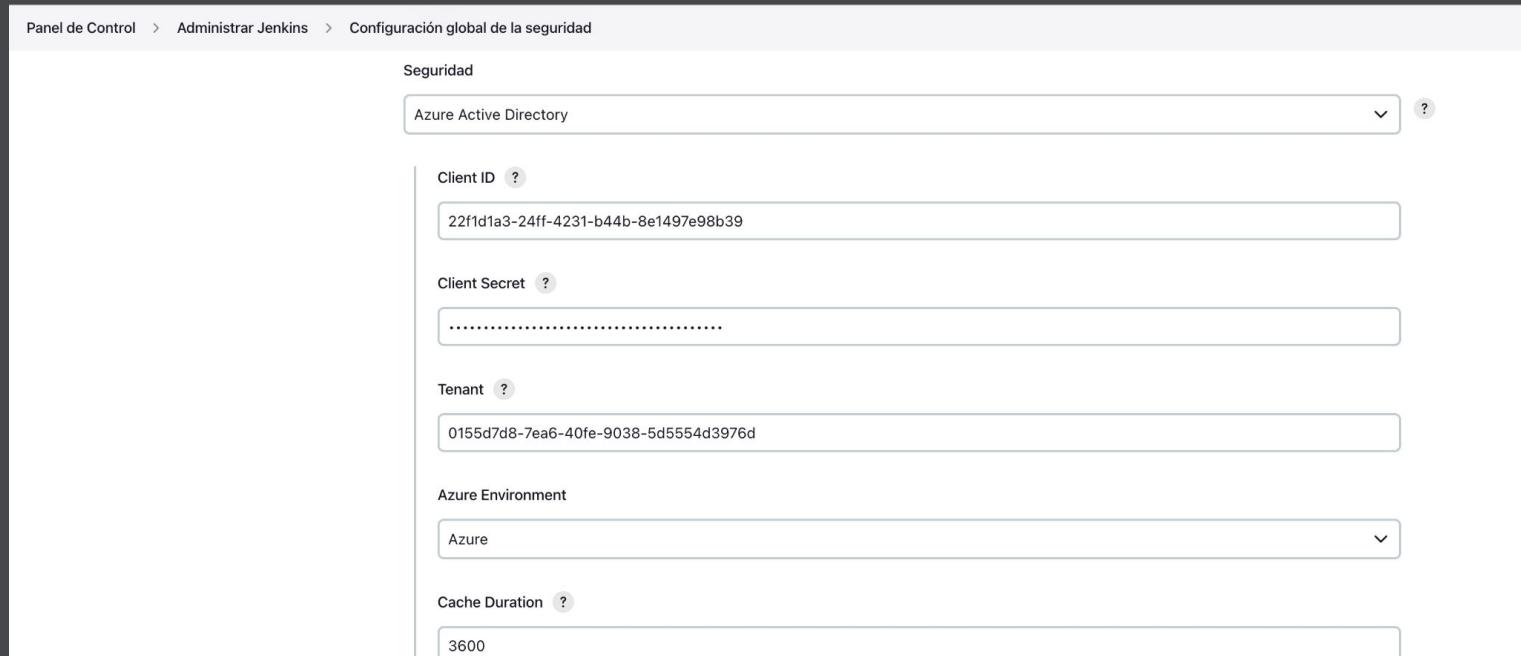
Client Secret ?  
.....

Tenant ?  
0155d7d8-7ea6-40fe-9038-5d5554d3976d

Azure Environment

Azure

Cache Duration ?  
3600



# Plugins: Role-based Authorization Strategy (<https://plugins.jenkins.io/role-strategy/>)

Plugin Manager

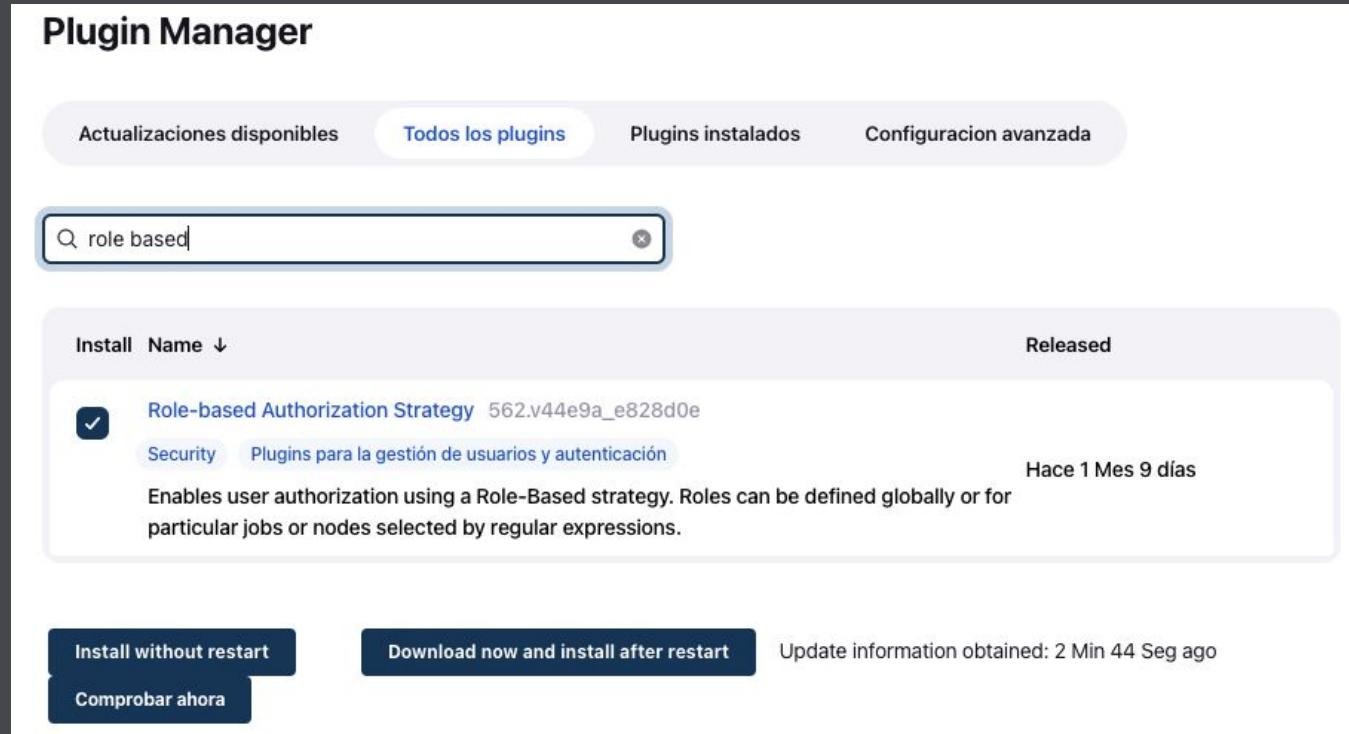
Actualizaciones disponibles Todos los plugins Plugins instalados Configuracion avanzada

Q role based

Install Name ↓ Released

Role-based Authorization Strategy 562.v44e9a\_e828d0e  
Security Plugins para la gestión de usuarios y autenticación Hace 1 Mes 9 días  
Enables user authorization using a Role-Based strategy. Roles can be defined globally or for particular jobs or nodes selected by regular expressions.

Install without restart Download now and install after restart Update information obtained: 2 Min 44 Seg ago  
Comprobar ahora



# Plugins: Role-based Authorization Strategy (<https://plugins.jenkins.io/role-strategy/>)

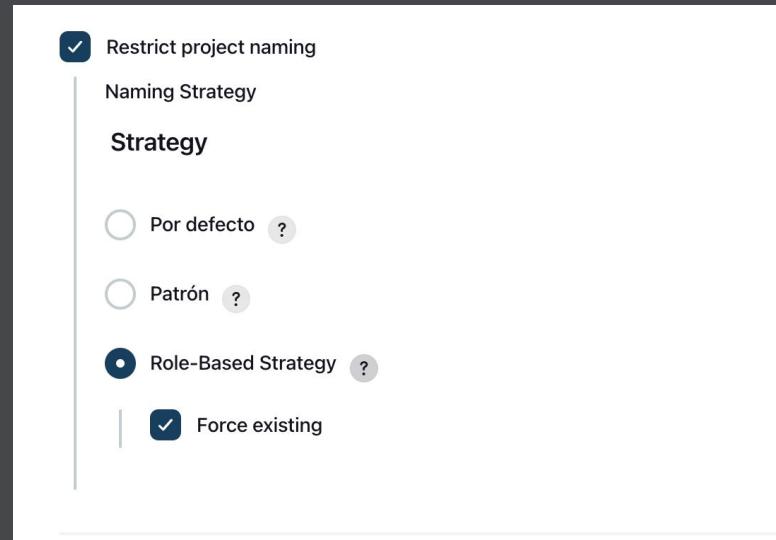
The screenshot shows the Jenkins interface with the "Manage and Assign Roles" plugin installed. The top navigation bar includes the Jenkins logo, a search bar with placeholder "búsqueda (⌘+K)", and the current path: "Panel de Control > Administrar Jenkins > Manage and Assign Roles". On the left, there's a sidebar with links: "+ Nueva Tarea", "Personas", "Historial de trabajos", "Administrar Jenkins", and "Mis vistas". Below this is a dropdown menu for "Trabajos en la cola" which says "No hay trabajos en la cola". At the bottom is a dropdown menu for "Estado del ejecutor de construcciones". The main content area is titled "Manage and Assign Roles" and contains three sections: "Manage Roles" (with a fingerprint icon), "Assign Roles" (with a user icon), and "Role Strategy Macros" (with a puzzle piece icon). Each section has a link to its documentation.

Manage and Assign Roles

- [Manage Roles](#)  
Manage Roles
- [Assign Roles](#)  
Assign Roles
- [Role Strategy Macros](#)  
Provides info about macro usage and available macros

# Plugins: Role-based Authorization Strategy (<https://plugins.jenkins.io/role-strategy/>)

- Global roles: Aplican a cualquier item en Jenkins y sobreescriben cualquier permiso que se especifique a nivel de Item Role. Por tanto, cuando se da un permiso para leer Jobs a nivel global, ya no importa lo que se indique a nivel de Item, el usuario tendrá acceso al Job.
- En los roles de Item o Agent, se pueden utilizar expresiones regulares para especificar el "ámbito" de aplicación de los permisos.
- Crear permisos a nivel de Item solo funciona correctamente si el campo Naming Strategy, de la configuración global de Jenkins, se ha establecido a 'Role-Based'. Únicamente podrán crearse items que cumplan las reglas de nombrado.



# Plugins: Role-based Authorization Strategy

Creación de roles globales de Jenkins

| Manage Roles |                                     | Global roles                        |                          |                          |                          |                          |                          |                                     |                                     |                                     |                                     |
|--------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Role         | Overall                             | Credentials                         |                          | Manage ownership         |                          | Agent                    |                          | Job                                 |                                     | Run                                 |                                     |
|              |                                     | SystemRead                          | SystemWrite              | View                     | Update                   | Create                   | Delete                   | Discover                            | Delete                              | Replay                              | Delete                              |
| admin        | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            |
| builder      | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| creator      | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

# Plugins: Role-based Authorization Strategy

Creación de roles de Jenkins

## Manage Roles

### Global roles

| Role         | Global                              | Credentials                         | Nodo                     | Tarea                               | Ejecutar                            | Vistas                              | Repositorio de software (SCM) |                          | Metrics                             |                                     |                                     |                                     |
|--------------|-------------------------------------|-------------------------------------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
|              |                                     |                                     |                          |                                     |                                     |                                     | Read                          | Delete                   | Create                              | Configure                           | Update                              | Replay                              |
| Rol Alumno   | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>      | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| Rol Profesor | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>      | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input checked="" type="checkbox"/> |
| admin        | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>      | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            |

Role to add

# Plugins: Role-based Authorization Strategy

Creación de roles de Jenkins aplicables sobre objetos (carpetas)

**Item roles**

| Role                                                    | Pattern                                                                                          | Credentials              |                          | Manage ownership         |                          | Job                      |                                     | Run                                 |                                     | Create                              |                                     |                                     |
|---------------------------------------------------------|--------------------------------------------------------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
|                                                         |                                                                                                  | View                     | Update                   | Nodes                    | Jobs                     | Discover                 | ExtendedRead                        | Move                                | Replay                              | Delete                              | Configure                           | Update                              |
| <input checked="" type="checkbox"/> folder-access       |  "(?i)folder"   | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            |
| <input checked="" type="checkbox"/> folder-builder      |  "(?i)folder/*" | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| <input checked="" type="checkbox"/> folder-creator      |  "(?i)folder/*" | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| <input checked="" type="checkbox"/> @CurrentUserIsOwner |  "*"            | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

**Role to add**

@CurrentUserIsOwner

**Pattern**

\*

# Plugins: Role-based Authorization Strategy

Creación de roles de Jenkins aplicables sobre objetos (carpetas)

Item roles

| Role            | Pattern          | Credentials              |                          | Tarea                               |                                     | Ejecutar                            | Vistas                              | Repository de software (SCM)        | Metrics                             |
|-----------------|------------------|--------------------------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
|                 |                  | Read                     | Delete                   | Configure                           | Update                              |                                     |                                     |                                     |                                     |
| item-alumnos    | "CarpetaAlumnos" | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| item-profesores | "Profesor*"      | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

Role to add

Pattern [?](#)

# Plugins: Role-based Authorization Strategy

Asignación de Roles (AD - Jenkins)

## Assign Roles

Global roles

| User/group                                      | Rol Alumno               | Rol Profesor             | admin                                                                                                                   |
|-------------------------------------------------|--------------------------|--------------------------|-------------------------------------------------------------------------------------------------------------------------|
| Alumnos                                         | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/>  |
| Profesores                                      | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/>  |
| Tornasol (5f3b38ff-ae3e-4b60-881a-1a76a28e8f3c) | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/>  |
| Anonymous                                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>             |

User/group to add 

Add

# Plugins: Role-based Authorization Strategy

Asignación de Roles (AD - Jenkins)

Item roles

| User/group | item-alumnos                        | item-profesores                     |
|------------|-------------------------------------|-------------------------------------|
| Alumnos    | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| Profesores | <input type="checkbox"/>            | <input checked="" type="checkbox"/> |
| Anonymous  | <input type="checkbox"/>            | <input type="checkbox"/>            |

User/group to add ?

Add

# Scripts Jenkins

Jenkins permite ejecutar scripts directamente en el nodo master mediante su consola de scripts.

The screenshot shows the Jenkins Script Console interface. At the top, there's a navigation bar with links to 'Panel de Control' > 'Administrar Jenkins' > 'Consola de scripts'. On the left, a sidebar lists 'Nueva Tarea', 'Personas', 'Historial de trabajos', 'Administrar Jenkins', and 'Mis vistas'. Below the sidebar, there are two dropdown menus: 'Trabajos en la cola' (which says 'No hay trabajos en la cola') and 'Estado del ejecutor de construcciones'. The main area is titled 'Consola de scripts' and contains a text input field with the Groovy script: `println(Jenkins.instance.pluginManager.plugins)`. A note below the input field states: 'Escribe un "script" [Groovy script](#) y ejecutalo en el servidor. Es útil para depurar e investigar problemas. Usa "println" para ver la salida (si usas System.out, se escribirá en la salida "stdout" del servidor, lo que es más difícil de visualizar). Ejemplo:'. To the right of the input field, a large text area displays the output of the script: '1'. At the bottom right of the output area is a blue 'Ejecutar' button.

# Scripts Jenkins

Por seguridad, Jenkins tiene deshabilitado el uso de una buena cantidad de funciones y librerías que debemos desbloquear para poder usarlas en nuestros scripts

The screenshot shows the Jenkins ScriptApproval interface. At the top, there's a navigation bar with the Jenkins logo, a search bar containing "búsqueda (⌘+K)", and user information for "Miguel". Below the navigation bar, the URL is "Panel de Control > Administrar Jenkins > ScriptApproval".

The main content area has a sidebar on the left with links: "+ Nueva Tarea", "Personas", "Historial de trabajos", "Administrador Jenkins", and "Mis vistas". A dropdown menu "Trabajos en la cola" is open, showing "No hay trabajos en la cola". Another dropdown menu "Estado del ejecutor de construcciones" is also open.

The central part of the screen displays a dialog box for approving a Groovy script from "miguel". The dialog contains two buttons: "Approve" and "Deny". The text "Groovy script from miguel" is followed by a large empty text area for the script content. Below this area, a message says "You can also remove all previous script approvals: [Clear Approvals](#)".

At the bottom of the dialog, it says "No pending signature approvals." and "Signatures already approved:" followed by another large empty text area. A footer note at the very bottom states "Signatures already approved assuming permission check".

# Scripts de Jenkins

- Podeis encontrar gran cantidad de scripts en  
<https://www.jenkins.io/doc/book/managing/script-console/>

# Scripts Jenkins

```
import hudson.model.*
def counter = 0
// Create a ref for closure
def delClos
// Define closure
delClos = {
 it.eachDir(delClos);
 if(it.getName().contains("workspace-files")) {
 it.eachFile {
 if(it.getName().endsWith(".tmp")) {
 println "Deleting file ${it.canonicalPath}";
 it.delete()
 counter++;
 }
 }
 }
}

// Apply closure
for(item in Hudson.instance.items) {
 println "Applying on " + item.rootDir
 delClos(item.rootDir)
}
println counter + " files deleted"
```

## Plugins: Sonarqube (<https://plugins.jenkins.io/sonar/>)

- Integración con servidor SonarQube
- Permite centralizar la configuración de la conexión a distintos servidores Sonar existentes, por ejemplo, podemos mantener la configuración de uno para entornos de desarrollo y otro adicional para producción.
- La configuración de este plugin se encuentra dentro de la Configuración Global de Jenkins.
- Existen distintas formas de lanzar el análisis:
  - Sonarqube CLI (sonar-scanner-cli)
  - Maven (Plugin de sonar para maven)
    - Configuración en pom.xml y en settings.xml

# Plugins: Sonarqube (<https://plugins.jenkins.io/sonar/>)

Panel de Control > Administrar Jenkins > Configurar el sistema >

## SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.



**Environment variables** Enable injection of SonarQube server configuration as build environment variables

## Instalaciones de SonarQube

Listado de instalaciones SonarQube

Add SonarQube

# Plugins: Sonarqube (<https://plugins.jenkins.io/sonar/>)

La credencial debe ser añadida a Jenkins Credentials como una credencial de tipo Secret Text

Panel de Control > Administrar Jenkins > Configurar el sistema >

Instalaciones de SonarQube  
Listado de instalaciones SonarQube

Name  ×

● Esta propiedad es obligatoria.

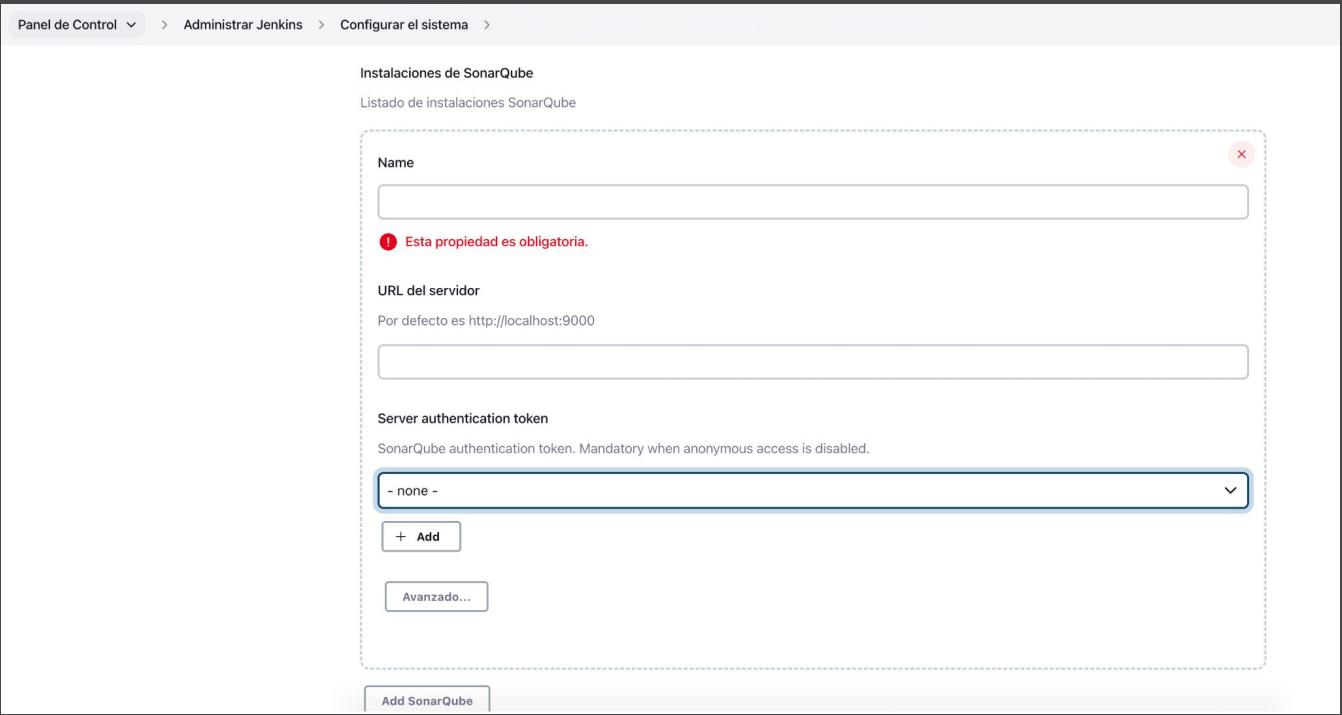
URL del servidor  
Por defecto es http://localhost:9000

Server authentication token  
SonarQube authentication token. Mandatory when anonymous access is disabled.  
 ▼

+ Add

Avanzado...

Add SonarQube



## Plugins: Sonarqube (<https://plugins.jenkins.io/sonar/>)

Desde nuestro pipeline, usaremos la función proporcionada por el plugin 'withSonarQubeEnv', indicando el nombre que le dimos a la configuración al darla de alta en Jenkins.

```
withSonarQubeEnv('My SonarQube Server', envOnly: true) {
 // This expands the environment variables SONAR_CONFIG_NAME, SONAR_HOST_URL,
 // SONAR_AUTH_TOKEN that can be used by any script.

 println ${env.SONAR_HOST_URL}
}

withSonarQubeEnv('SONAR_DEV') {
 sh 'mvn org.sonarsource.scanner.maven:sonar-maven-plugin:3.7.0.1746:sonar'
}
```

## Jenkins: Estructura de directorios

- La ruta principal de Jenkins viene indicada en la variable de entorno \$JENKINS\_HOME.
- La clave del controlador es utilizada para encriptar los datos existentes en el directorio de secretos, el cual contiene las credenciales. Se almacena en el fichero \$JENKINS\_HOME/secrets/hudson.util.Secret. Si necesitamos recuperar el sistema desde una copia de seguridad, necesitaremos este fichero. Cualquiera que posea este fichero, podrá desencriptar nuestros secretos desde un backup.
- Por tanto, el fichero master.key debe ser respaldado en una sitio distinto al donde se respalde el resto del sistema.
- En caso de que debamos restaurar el sistema al completo, primero restauraremos el resto del sistema y por último la clave

# Jenkins: Estructura de directorios

- JENKINS\_HOME: Respaldando este directorio completamente estaremos guardando toda la instancia de Jenkins. Para restaurarlo simplemente tendremos que copiar el backup en el nuevo sistema.
- No obstante, la carpeta JENKINS\_HOME contiene gran cantidad de ficheros que no son necesarios respaldar.
- Los ficheros de configuración (config.xml y \*.xml) se encuentran directamente en \$JENKINS\_HOME. Podemos hacer una copia de todos ellos. Una práctica común consiste en respaldarlos en un repositorio SCM.
- Dentro de JENKINS\_HOME encontramos el directorio 'jobs', donde se guarda toda la información de los jobs ejecutados en el sistema.
  - ./builds – Contains build records
  - ./builds/archive – Contains archived artifacts!!!!
  - ./workspace – Contiene fichero obtenidos a partir de SCM, probablemente no necesitemos respaldar esta carpeta.
  - ./plugins/\*.hpi – Plugin packages with specific versions used on your system
  - ./plugins/\*.jpi – Plugin packages with specific versions used on your system

# Jenkins: Estructura de directorios

---

No necesitamos respaldar las siguientes carpetas:

**./war** – Exploded war file. To restore a system, download the latest war file.

**./cache** – Downloaded tools. To restore a system, download the current version of the tools.

**./tools** – Extracted tools. To restore a system, extract the tools again.

**./plugins/xxx** – Subdirectories of installed plugins. These will be automatically populated on the next resta

## Jenkins: Estructura de directorios

---

Para chequear que hemos realizado correctamente la copia de seguridad del sistema, basta con apuntar la variable de entorno JENKINS\_HOME a la carpeta donde se encuentra nuestro backup.

A continuación debemos lanzar de nuevo la instancia de Jenkins.

```
java -jar jenkins.war --httpPort=9999
```

# Jenkins: Troubleshooting

- Usaremos los logs del master de Jenkins como fuente de información a la hora de diagnosticar un problema. Una forma sencilla de consultar estos logs es usando la herramienta OpenLens.
- También podemos iniciar un shell en el nodo master y ejecutar los comandos que nos resulten útiles para diagnosticar.
- En sistemas Jenkins, es habitual que los volúmenes de datos usados para persistir la configuración, los jobs, etc se queden sin espacio o sin 'inodos'.
- ¿Qué son inodos en sistemas unix?
- df -hi
- Borrado de ficheros temporales. Cuando lanzamos un job, a menudo se descargan las librerías utilizadas para su ejecución. Estas librerías ocupan un espacio en el sistema y son perfectamente prescindibles ya que se volverán a descargar en la próxima ejecución.

# Plugins: GitHub Branch Source (<https://plugins.jenkins.io/github-branch-source/>)

- Escanea automáticamente en busca de nuevos repositorios y/o ramas en los mismos y los incorpora a Jenkins

The screenshot shows the Jenkins Plugin Manager interface. The top navigation bar includes 'Panel de Control' > 'Administrar Jenkins' > 'Gestor de plugins'. On the left sidebar, there are links for 'Volver al Panel de control', 'Administrador Jenkins', and 'Centro de actualizaciones'. The main area is titled 'Plugin Manager' with tabs for 'Actualizaciones disponibles', 'Todos los plugins' (which is selected), 'Plugins instalados', and 'Configuración avanzada'. A search bar at the top has 'github' typed into it. Below the search bar, a table lists three plugins:

| Install                             | Name                                                                                                                                | Released           |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|--------------------|
| <input type="checkbox"/>            | GitHub 1.36.0<br>Plugins de integración con sitios y herramientas externas   <a href="#">github</a>                                 | Hace 16 días       |
| <input type="checkbox"/>            | GitHub API 1.303-400.v35c2d8258028<br><a href="#">github</a>   Library plugins (for use by other plugins)                           | Hace 6 Mes 22 días |
| <input checked="" type="checkbox"/> | GitHub Branch Source 1696.v3a_7603564d04<br><a href="#">pipeline</a>   <a href="#">github</a>   Plugins de repositorios de software | Hace 4 días 23 Hor |

A note at the bottom of the list states: 'Multibranch projects and organization folders from GitHub. Maintained by CloudBees, Inc.'

# Plugins: GitHub Branch Source (<https://plugins.jenkins.io/github-branch-source/>)

- Creamos un nuevo item del tipo Multibranch Pipeline

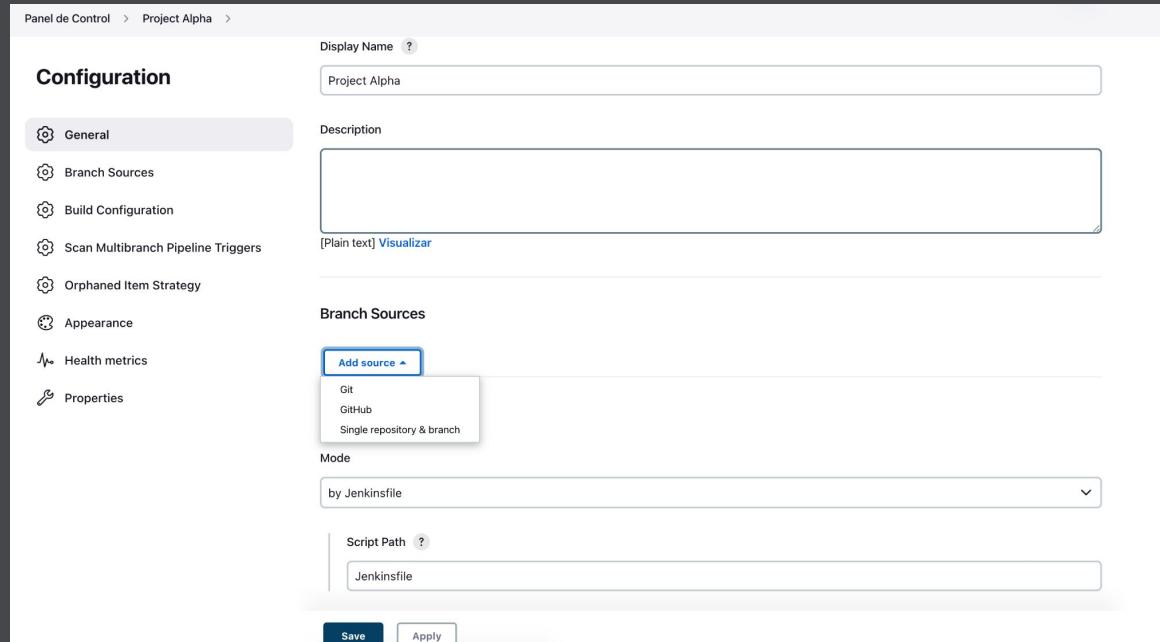
The screenshot shows the Jenkins 'Panel de Control > Todo' interface. A search bar at the top contains the text 'Project Alpha'. Below it, a list of project types is displayed:

- Crear un proyecto de estilo libre**: Describes a自由风格的项目, combining any type of software repository (SCM) with any build or execution mode (make, ant, mvn, rake, script...). It's suitable for both compilation and packaging.
- Pipeline**: Manages long-duration activities across multiple agents. Appropriate for building pipelines (formerly known as workflows) or organizing complex activities.
- Crear un proyecto multi-getConfiguración**: Suitable for projects requiring many different configurations, such as testing across multiple environments or executing on specific platforms.
- Folder**: Creates a container for nested items. Useful for grouping things together. Unlike a view, which is just a filter, a folder creates a separate namespace, allowing multiple items with the same name if they are in different folders.
- Multibranch Pipeline**: Creates a set of Pipeline projects based on detected branches in one SCM repository. This is the selected option, highlighted with a blue border.
- Organization Folder**: Creates a set of multibranch project subfolders by scanning for repositories.

A large blue 'OK' button is located at the bottom of the list.

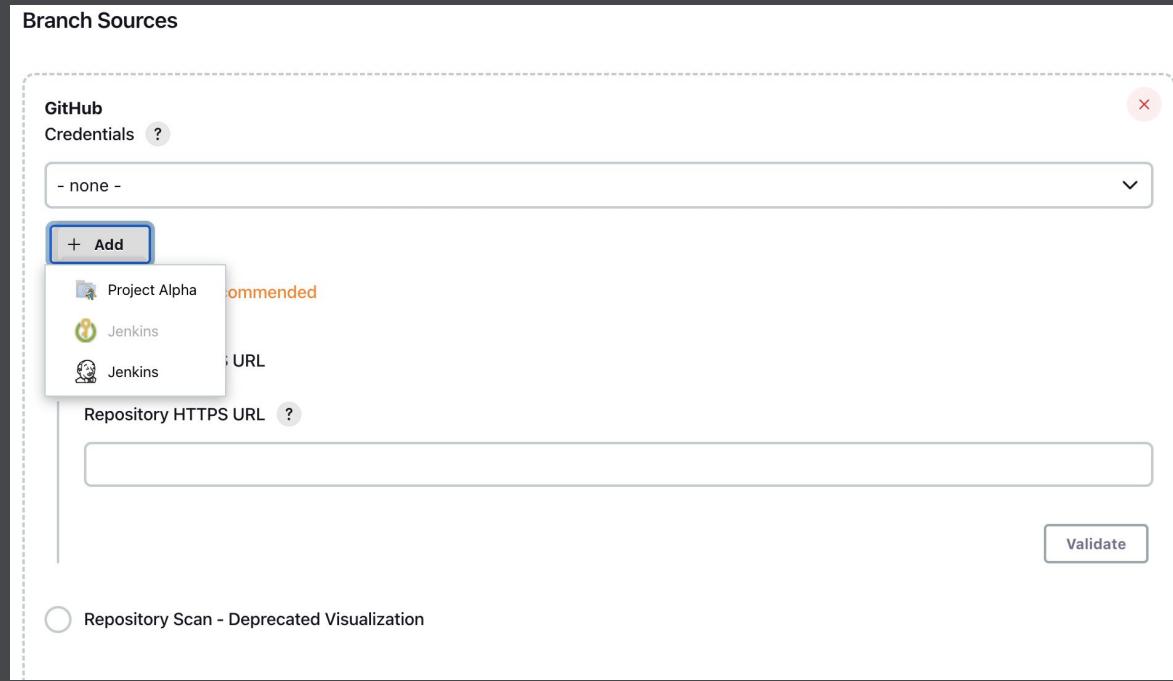
# Plugins: GitHub Branch Source (<https://plugins.jenkins.io/github-branch-source/>)

- En la configuración del item, seleccionamos GitHub como origen de las ramas.



# Plugins: GitHub Branch Source (<https://plugins.jenkins.io/github-branch-source/>)

- Creamos la credencial necesaria para poder conectar a GitHub.



# Plugins: GitHub Branch Source (<https://plugins.jenkins.io/github-branch-source/>)

- En GitHub creamos una App, obtenemos el PAT y lo usamos para crear una credencial de tipo Username with password.

The screenshot shows a modal dialog for creating a new credential. The 'Kind' dropdown is set to 'Username with password'. The 'Username' field contains 'Jenkins'. The 'Password' field is filled with a series of dots. The 'ID' field is set to 'JENKINS-GITHUB-APP'. The 'Description' field also contains 'JENKINS-GITHUB-APP'. At the bottom, there are 'Add' and 'Cancel' buttons.

Kind

Username with password

Username ?  
Jenkins

Treat username as secret ?

Password ?  
.....

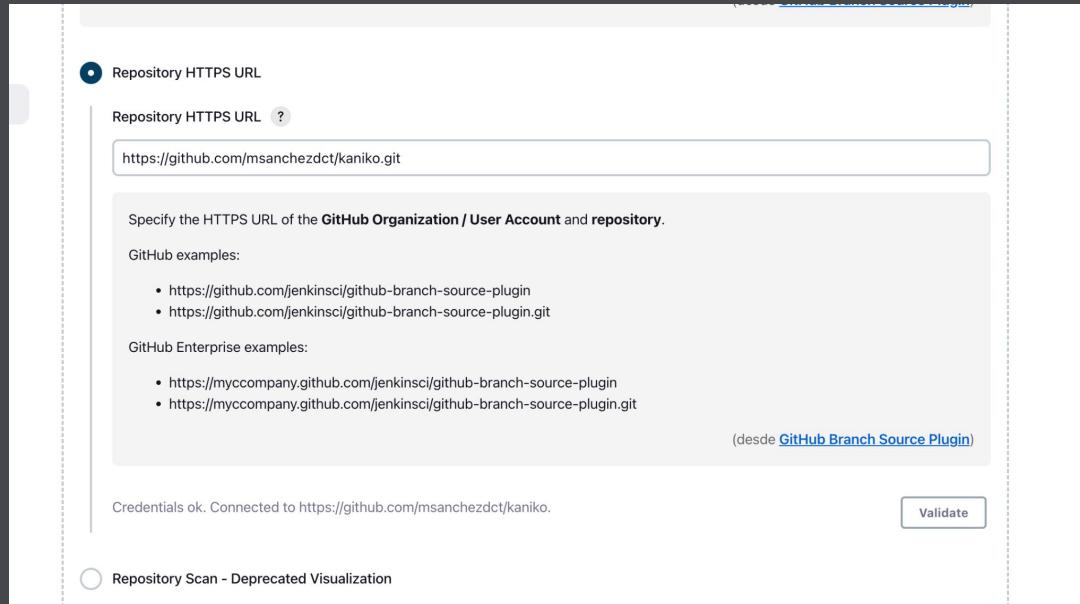
ID ?  
JENKINS-GITHUB-APP

Description ?  
JENKINS-GITHUB-APP

Add Cancel

# Plugins: GitHub Branch Source (<https://plugins.jenkins.io/github-branch-source/>)

- Incluimos la url del repositorio



# Jenkins: Configuración de librerías globales

- Concepto de sandbox.

Panel de Control > Administrar Jenkins > Configurar el sistema >

## Global Pipeline Libraries

Sharable libraries available to any Pipeline jobs running on this system. These libraries will be trusted, meaning they run without "sandbox" restrictions and may use @Grab.

**Library**

Name  X

**Required** You must enter a name.

Default version

Load implicitly ?

Allow default version to be overridden ?

Include @Library changes in job recent changes ?

Cache fetched versions on controller for quick retrieval ?

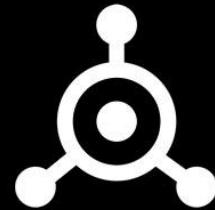
Retrieval method

**Guardar** **Apply**

## **Jenkins:**

---

-



KEEP  
CALM  
DEVCENTER  
IS  
WORKING  
ON IT