Instrucciones de Despliegue Turnero Desarrollo

Proyecto: Nuevo Turnero Etapa 2

Revisiones

Revisión	Autor	Fecha	Descripción
1.0	Julián Gigena	10-03-2021	Versión inicial

Índice

Proyecto: Nuevo Turnero Etapa 2 1

Índice 2

Detalle del ambiente 2

Servidores 3

Desarrollo 3

Pre-Producción 4

Producción 4

Usuario de servicio 5

Configuración previa 6

Prerrequisito 6

Validaciones de entorno 7

Configuración de proxy 7

Instalación de Docker-Compose 8

Configuración de los Nombres de Dominio (Opcional) 8

Configuración Variables de Entorno de Compilación de Imágenes 9

Configuración de cluster de Swarm 9

Desplegar aplicaciones de soportes 10

Monitoreo - Métricas 12

Ejecución de entorno 14

Pos-Validación de Entorno 15

Consolidar las Imágenes en Registry 15

Integración Continua 16

Configurar Jenkins 16

Desplegar Aplicaciones Turnero 23

Despliegue Manual 24

Procedimientos Pos-Implementación 25

Agregar o Reemplazar un Nuevo Nodo 25

Actualizar Certificados HTTPS por vencimientos 26

Detalle del ambiente

Servidores

Desarrollo

IP / Name	Destino	Puertos expuestos	FQDN
d250lxcde34.gobiernocba.gov .ar (10.250.52.1)	Swarm Cluster Nodo 1	443/https - Internet 22/ssh - Interno	turnero-dev.test.cba.gov.ar monitor-turnero-dev.test.cba.gov . ar
d250lxcde35.gobiernocba.gov .ar (10.250.52.2)	Swarm Cluster Nodo 2		
d250lxcde36.gobiernocba.gov.ar (10.250.52.3)	Nodo Monitoreo (ex Soporte)		trace-turnero-dev.test.cba.gov.a r metrics-turnero-dev.test.cba.gov . ar

Testing

IP / Name	Destino	Puertos expuesto	FQDN
d250lxcte01.gobiernocba.go v.ar (10.250.10.170)	Swarm Cluster Nodo 1	443/https - Internet 22/ssh - Interno	turnero-test.test.cba.gov.ar (balanceado) monitor-turnero-test.test.cba. gov.ar
d250lxcte02.gobiernocba.go v.ar (10.250.10.171)	Swarm Cluster Nodo 2		
d250lxcte03.gobiernocba.go v.ar (10.250.10.172)	Nodo Soporte		registry-turnero-test.test.cba.g ov.ar build-turnero-test.test.cba.gov .ar trace-turnero-test.test.cba.gov .ar metrics-turnero-test.test.cba.g ov.ar

Pre-Producción

IP / Name	Destino	Puertos expuesto	FQDN
d250lxcpp01.gobiernocba.gov.ar (10.250.2.48)	Swarm Cluster Nodo 1	443/https - Internet 22/ssh - Interno	turnero.test.cba.gov.ar (balanceado) monitor-turnero.test.cba.gov.ar
d250lxcpp02.gobiernocba.gov.ar (10.250.2.49)	Swarm Cluster Nodo 2		
d250lxcpp03.gobiernocba.gov.ar (10.250.2.30)	Nodo Soporte		registry-turnero.test.cba.gov.ar build-turnero.test.cba.gov.ar trace-turnero.test.cba.gov.ar metrics-turnero.test.cba.gov.ar

Producción

IP / Name	Destino	Puertos expuesto	FQDN
d250lxcpr02.gobiernocba.gov.ar (10.250.68.12)	Swarm Cluster Nodo 1	443/https - Internet 22/ssh - Interno	turnero.cba.gov.ar (balanceado) monitor-turnero.cba.gov.ar
d250lxcpr03.gobiernocba.gov.ar (10.250.68.13)	Swarm Cluster Nodo 2		
d250lxcpr04.gobiernocba.gov.ar (10.250.68.14)	Swarm Cluster Nodo 3		
d250lxcpr05.gobiernocba.gov.ar (10.250.68.15)	Swarm Cluster Nodo 4		
d250lxcpr26.gobiernocba.gov.ar (10.250.68.47)	Swarm Cluster Nodo 5		
d250lxcpr27.gobiernocba.gov.ar (10.250.68.48)	Swarm Cluster Nodo 6		

d250lxcpr28.gobiernocba.gov.ar (10.250.68.49)	Swarm Cluster Nodo 7		
d250lxcpr29.gobiernocba.gov.ar (10.250.68.50)	Swarm Cluster Nodo 8		
d250lxcpr01.gobiernocba.gov.ar (10.250.68.11)	Nodo Soporte		registry-turnero.test.cba.gov.ar build-turnero.test.cba.gov.ar trace-turnero.test.cba.gov.ar metrics-turnero.test.cba.gov.ar

Usuario de servicio

Los siguientes usuarios deben ser creados y configurados como pre-requisitos:

Usuario	Nodo	Permisos	Propósito
devuser	d250lxcde34 d250lxcde35 d250lxcde36	debe pertenecer al grupo de sudo y docker, srv_build, sa_turnero en d250lxcde36 (soporte) en el resto de los nodos debe ser solo docker y sudo.	Usuario para el proceso de despliegue en este procedimiento.
srv_deploy (1002)	d250lxcde34 d250lxcde35	acceso a /opt/turnero en d250lxcde34 y d250lxcde35 a través de ssh passwordless rsa. Debe pertenecer a los grupos docker, sshusers y sa_turnero	Usuario de servicio del servicio de despliegue Ansible para la publicación de la aplicación en el Swarm del entorno
GOBIERNOCBA\svc_Turnero_Proje	N/A	Acceso a TFS para el proyecto turnero con permisos de	Usuario de servicio que permite extraer el código del repositorio TFS dentro del proceso de compilación.

ct		reader.	
ctr_jenkins (10101)	d250lxcde36	Usuario regular	Usuario proxy para mapear usuarios de contenedor y host en al volumen
sa_turnero (10102)	d250lxcde34 d250lxcde35 d250lxcde36	Usuario regular, debe pertenecer al grupo docker en d250lxcde36 (soporte)	Usuario proxy para mapear usuario de contenedor y host en los volúmenes de componentes de proyecto. Usuario de servicio del proceso de Integración continua para la conexión ssh entre Jenkins Master y Jenkins Agente.
reg_turnero	N/A	Usuario de registry	Usuario de registry para acceso para la aplicación de turnero

En caso de no existir, crearlos y configurarlos como se indica en la tabla previa en el nodo especificado (Debe ser ejecutado con permisos de superusuario), por ejemplo:

```
useradd -ms /bin/bash -u 10102 -d /home/sa_turnero
sa_turnero gpasswd -a sa_turnero docker
```

Configuración previa

Nota: Esta configuración debe repetirse en todos los nodos del ambiente. Los usuarios mencionados previamente se consideran pre-requisitos para la instalación. Los comandos en los pasos del procedimiento que aquí se describe se suponen ejecutados sobre el usuario devuser a menos que se indique lo contrario:

```
sudo su - devuser
```

Prerrequisito

1. Requerir servidores para cada nodo de imagen Docker DGCIT-Debian9-1.0.2.
2. Los certificados wildcard de dominio deben ser copiados en el directorio /home/sysoper
3. Los usuarios listados en la tabla del comienzo del documento deben ser creados y passwords deben serle asignados. El equipo de aplicaciones no necesita conocer esas contraseñas.

Validaciones de entorno

Configuración de proxy

1. Validar versión de docker instalado debe ser al menos Docker versión 18.09.6.

```
sudo docker version
```

De no ser la versión especificada o superior se debe actualizar la distribución. Aun siéndolo se recomienda actualizar a la última.

```
sudo apt update; sudo apt upgrade
```

Validar nuevamente que la versión instalada sea la esperada.

2. Configurar el proxy a nivel de docker en todos los nodos:
   a. De no existir el directorio crearlo

```
sudo mkdir -p /etc/systemd/system/docker.service.d
```

   b. Editar el archivo

```
sudo vi /etc/systemd/system/docker.service.d/http-proxy.conf
```

   e incluir

```
[Service]
Environment="HTTP_PROXY=http://proxydesa.gobiernocba.gov.
ar:8080"
Environment="HTTPS_PROXY=http://proxydesa.gobiernocba.gov
.ar:8080"
```

   c. Excluir las urls locales que no deban ir por proxy agregando este bloque al mismo archivo:

```
Environment="NO_PROXY=registry-turnero-dev.test.cba.gov.ar,bu
ild-turne
ro-dev.test.cba.gov.ar,metrics-turnero-dev.test.cba.gov.ar,tr
ace-turne ro-dev.test.cba.gov.ar,127.0.0.1,localhost"
```

   d. Reiniciar servicio de docker para que tome los cambios anteriores

```
sudo systemctl daemon-reload;
sudo service docker stop;
sudo service docker start;
```

3. Configurar el proxy del sistema operativo en todos los nodos. Esta configuración debe ir en el .profile del usuario que ejecuta comandos docker/docker-compose (`devuser` y `sa_turnero` en `d250lxcde34, d250lxcde35 y d250lxcde36`):

```
cd ~
vi .profile
```

Incluir el siguiente contenido al final del archivo:

```
export http_proxy=http://proxydesa.gobiernocba.gov.ar:8080
export https_proxy=http://proxydesa.gobiernocba.gov.ar:8080
export
no_proxy=registry-turnero-dev.test.cba.gov.ar,build-turnero-dev.test.cba.gov
.ar,metrics-turnero-dev.test.cba.gov.ar,trace-turnero-dev.test.cba.gov.ar,*.
gobiernocba.gov.ar,*.cba.gov.ar,*.test.cba.gov.ar,127.0.0.1,localhost
```

Instalación de Docker-Compose

Los siguientes pasos se deben ejecutar en todos los nodos.

1. Docker compose se usa para desplegar imágenes correlacionadas de servicios. En nuestro caso para desplegar las apps de soporte, registry, jenkins, elk:

```
sudo apt install docker-compose
```

2. Validar versión de docker compose instalada para que sea superior a 1.7.0

```
docker-compose version
```

Configuración de los Nombres de Dominio (Opcional)

1. Validar la resolución DNS de los FQDN de los dominios de la solución :

```
registry-turnerodev.test.cba.gov.ar
build-turnero-dev.test.cba.gov.ar
trace-turnero-dev.test.cba.gov.ar
metrics-turnero-dev.test.cba.gov.ar
monitor-turnero-dev.test.cba.gov.ar
```

El siguiente dominio es validado luego de configurar el balancer para la aplicación

```
turnero-dev.test.cba.gov.ar
```

Configuración Variables de Entorno de Compilación de Imágenes

1. Incluir las siguientes variables en el .profile del usuario que se usa para ejecutar docker-compose. Estas variables son utilizadas por docker compose como parte de las definiciones del ambiente `devuser` y `sa_turnero` en `d250lxcde34, d250lxcde35, d250lxcde36` y `srv_build` en `d250lxcde36`.

```
export DOCKER_REGISTRY=registry-turnero-dev.test.cba.gov.ar
export PROXY_SERVER=proxydesa.gobiernocba.gov.ar
exportENVIRONMENT=Desarrollo
export ENVIRONMENT_SUFFIX=-dev.test
export CURRENT_IP=`host \`hostname\`  | cut -d" " -f4`
```

Configuración de cluster de Swarm

Los siguientes pasos indican en qué nodo deben ser ejecutados.

1. Todos los nodos existentes se crean como master:
   a. En el nodo maestro `d250lxcde34` inicializar el swarm

```
docker swarm init --advertise-addr `echo $CURRENT_IP`
```

   b. En el mismo nodo solicitar un token de join para ser ejecutado en el resto los nodos

```
docker swarm join-token manager
```

   c. Usar el link provisto por el comando anterior en `d250lxcde35` asegurando que se hace sudo sobre el resultado.

```
sudo docker swarm join --token
SWMTKN-1-07zsm0ibeph7hofcf5h84fdn75b37h0paqwiyzs5kqi95248uj-5
vghr81vla oe9ph0s3u3xgk6j 10.250.52.1:2377
```

   El token resaltado es lo que varía en cada ejecución y esta determinado del paso anterior.

   d. Verificar que todos los nodos estén disponibles:

```
docker node ls
```

Desplegar aplicaciones de soportes

El siguiente despliegue corresponde al nodo de soporte exclusivamente (`d250lxcde36.gobiernocba.gov.ar/10.250.52.3`). Este nodo se despliega sobre docker pero por fuera del cluster swarm. Ver en la tabla de servidores el nodo designado de soporte según corresponda.

1. Crear el directorio base de instalación

```
sudo mkdir -p /opt/turnero

sudo chown devuser.devuser /opt/turnero
```

2. Crear el directorio de logs

```
sudo mkdir -p /var/log/turnero/nginx;
```

3. Crear el directorio de volúmenes de aplicaciones

```
sudo mkdir -p /mnt/cdata/volumes/turnero/build-server;
sudo mkdir -p /mnt/cdata/volumes/turnero/monitoring/elk/elasticsearch;
sudo mkdir -p /mnt/cdata/volumes/turnero/monitoring/prometheus;
sudo mkdir -p /mnt/cdata/volumes/turnero/monitoring/grafana;
sudo chown -R sa_turnero.sa_turnero /mnt/cdata/volumes/turnero;
```

4. Ajustar permisos de directorios de volúmenes de datos

```
cd /mnt/cdata/volumes/turnero
sudo chown -R sa_turnero:sa_turnero monitoring
sudo chown -R nobody:nogroup monitoring/prometheus;
sudo chown -R sa_turnero:sa_turnero monitoring/grafana;
sudo chown -R ctr_jenkins:ctr_jenkins build-server;
sudo chown -R sa_turnero:sa_turnero monitoring/elk/elasticsearch;
```

5. Descargar y descomprimir en /opt/turnero el paquete de configuración solución-aplicativa-turnero-desarrollo-fs-opt-support-YYYY-MM-DD-VV.tar.tgz. Ejecutar como root:

```
cd /home/sysoper/
mv
solucion-aplicativa-turnero-desarrollo-fs-opt-support-YYYY-MM-DD-VV.tar.gz
/opt
cd/tar xvzf
/opt/solucion-aplicativa-turnero-desarrollo-fs-opt-support-YYYY-MM-DD-VV.tar
.gz
```

```
chown -R sa_turnero:sa_turnero /opt/turnero
chown srv_build:srv_build /opt/turnero/build-server/agent/ssh/ansible_rsa
```

El resultado final debe quedar conformado en sus primeros niveles de la siguiente forma:

```
/opt
└── turnero
    ├── build-server
    ├── gateway
    └── monitoring
```

6. Ajustar permisos en directorios y archivos descomprimidos

```
cd /opt/turnero
find . -type d -exec sudo chmod 755 {} \;
find . -type f -exec sudo chmod 644 {} \;
```

Instalación de Registry (Opcional)

El despliegue del registry de docker es opcional y solo requerido en caso de no existir otro registry disponible para este propósito.

1. Generar archivo de password para validación de usuarios de registry. Se debe seleccionar una contraseña que debe ser utilizada más adelante en el proceso (el nombre de usuario a utilizar es registry):

```
sudo su - sa_turnero

docker run --rm --entrypoint htpasswd registry:2 -Bbn reg_turnero {password}
> /opt/turnero/gateway/auth/registry.passwd;
```

2. Al utilizar certificate `*.test.cba.gov.ar` se debe copiar el certificado (provisto por infraestructura al momento del despliegue: `wildcard_test_cba_pkey.key` y `wildcard_test_cba_chain.pem`, en el nodo de soporte `d250lxcde36.gobiernocba.gov.ar`) a la carpeta de autoridades y actualizar la lista. Esto es necesario para evitar validaciones de certificados. Muchos servicios consideran como error fatal. Ejecutar como root:

```
cp /home/sysoper/wildcard_test_cba*
/opt/turnero/gateway/nginx/ssl/ cd
/opt/turnero/gateway/nginx/ssl
chown sa_turnero.sa_turnero wildcard_test_cba*
sudo su - sa_turnero
cd /opt/turnero/gateway/nginx/ssl
ln -s wildcard_test_cba_pkey.key
domain.key ln -s
wildcard_test_cba_chain.pem
domain.crt chmod 640
```

```
wildcard_test_cba*
```

3. Crear link simbólicos entre los archivos de configuración genéricos a sus equivalentes por entorno. Ejecutar como root:

```
cd /opt/turnero/gateway/nginx/conf.d;
ln -s ../sites-enabled/001_registry-turnero.conf
001_registry-turnero-dev.test.cba.gov.ar.conf; ln -s
../sites-enabled/002_build-turnero.conf
002_build-turnero-dev.test.cba.gov.ar.conf;
ln -s ../sites-enabled/003_metrics-turnero.conf
003_metrics-turnero-dev.test.cba.gov.ar.conf; ln -s
../sites-enabled/004_trace-turnero.conf
004_trace-turnero-dev.test.cba.gov.ar.conf;
ln -s ../sites-enabled/005_prometheus-turnero.conf
005_prometheus-turnero-dev.test.cba.gov.ar.conf;
```

Monitoreo - Métricas

Las capacidades de monitoreo se dividen en métricas y traza. Al igual que en el caso anterior este despliegue debe ser realizado en el nodo de soporte. Ejecutar como root:

1. Cambiar el password de admin de grafana por defecto. Editar

```
vi /opt/turnero/monitoring/prometheus/grafana/config.monitoring
```

y modificar para asignar una contraseña de administración del portal

```
GF_SECURITY_ADMIN_PASSWORD={password}
```

2. Enlazar el archivo del entorno de prometheus. Este archivo es el que debe editarse en caso de modificación de nodos.

```
cd /opt/turnero/monitoring/prometheus/prometheus ln -s prometheus.Desarrollo.yml
prometheus.yml
```

3. Configurar el acceso desde la receta de despliegue al nodo maestro principal de swarm. Este usuario debería existir específicamente para esta tarea. El procedimiento debe ser ejecutado en el Nodo 1 del cluster 10.250.52.1 `d2501xcde34.gobiernocba.gov.ar`.
   a. Impersonar al usuario de `srv_build`. Este usuario es sobre el que corren los procesos de agente de build.

```
sudo su - srv_deploy
```

b. Crear clave pública y privada de acceso ssh (no incluir passphrase ya que ansible no lo soporta).

```
mkdir ~/.ssh cd ~/.ssh/;
ssh-keygen -t rsa -C "Ansible Agent Key" -f "ansible_rsa" mv

ansible_rsa.pub authorized_keys
```

c. Copiar el certificado de clave privada al nodo de soporte para su uso en el build agent.

```
scp ansible_rsa
devuser@d250lxcde36.gobiernocba.gov.ar:/home/devuser; rm
ansible_rsa
```

d. En el nodo de soporte `d250lxcde36.gobiernocba.gov.ar` mover el certificado a la carpeta destino como root.

```
cd /opt/turnero/build-server/agent/ssh/
mv /home/devuser/ansible_rsa
/opt/turnero/build-server/agent/ssh/ chown
srv_build.srv_build ansible_rsa
chmod 444 ansible_rsa
```

Creación de imágenes

1. Las imágenes a utilizar se han configurado de formas que todas contengan el prefijo turnero. Esto facilita el despliegue, reconocimiento de componentes en el registry y customizaciones futuras de ser necesario. Por lo tanto, previo ejecutar los containers del entorno se deben construir y desplegar en el registry aun cuando esto no es completamente necesario. Como usuario sa_turnero ejecutar en el nodo de soporte:

```
cd /opt/turnero/

docker-compose -f
monitoring/prometheus/docker-compose.yml -p
supporting-services build --force-rm --pull

docker-compose -f monitoring/elk/docker-compose.yml -p
supporting-services build --force-rm --pull

docker-compose -f gateway/docker-compose.yml -p supporting-services build
--force-rm --pull

docker-compose -f build-server/docker-compose.yml
-p supporting-services build --force-rm --pull
```

Ejecución de entorno

2. Todas las imagenes deben ser levantadas sobre el mismo nombre de proyecto ya que la mayoría de los componentes depende de poder acceder a

```
cd /opt/turnero/

docker-compose -f monitoring/prometheus/docker-compose.yml -p turnero up -d
--force-recreate

docker-compose -f monitoring/elk/docker-compose.yml -p turnero up -d
--force-recreate

docker-compose -f gateway/docker-compose.yml -p turnero up -d
--force-recreate

docker-compose -f build-server/docker-compose.yml -p turnero up -d
--force-recreate
```

Pos-Validación de Entorno

1. Validar que el registro está funcionando utilizando el usuario sa_turnero:
   a. Probar logeo al registry

```
docker login -u reg_turnero registry-turnero-dev.test.cba.gov.ar
```

   b. Probar obtener lista de imágenes (tiene que devolver una lista vacía). Obtener el basic de autenticación de ~/.docker/config.json.

```
{
 "auths": {
  "registry-turnero-dev.test.cba.gov.ar
          ": { "auth":
          "cmVnaXN0cnk6c3VwZXJzZWNyZXQ=
          "
  }
}
```

luego ejecutar la consulta de forma estándar para un basic auth utilizando curl.

Ejemplo:

```
curl -X GET
https://registry-turnero-dev.test.cba.gov.ar/v2/_catalog
-H "Authorization: basic cmVnaXN0cnk6c3VwZXJzZWNyZXQ="
```

C. En ciertas circunstancias es posible que docker solicite el login para `registry-turnero-dev.test.cba.gov.ar:443` dado que las imágenes deben ser desplegadas con esa firma de registry y docker en algunos momentos puede no reconocer que son equivalentes. Por lo tanto es conveniente hacer el logueo también en

```
docker login -u reg_turnero registry-turnero-dev.test.cba.gov.ar:443
```

Para generar el registro en "/home/sa_turnero/.docker/config.json" con esa entrada también (O duplicarla a mano).

Consolidar las Imágenes en Registry

1. Subir las imágenes creadas en el registry global

```
cd /opt/turnero/

docker-compose -f monitoring/prometheus/docker-compose.yml
-p supporting-services push

docker-compose -f monitoring/elk/docker-compose.yml
-p supporting-services push
```

```
docker-compose -f gateway/docker-compose.yml -p supporting-services
push

docker-compose -f build-server/docker-compose.yml
-p supporting-services push
```

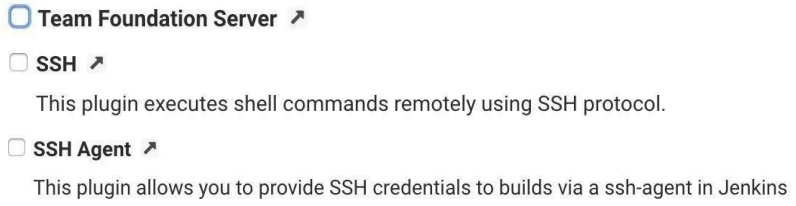
Integración Continua

Configurar Jenkins

4. Configuración inicial. Acceder a Jenkins e ingresar las credenciales temporales del usuario administrador según se indica en la pantalla inicial.
5. En la pantalla inicial elegir instalar los plugins sugeridos.
6. Configurar proxy de Plugin Manager (Jenkins->Administrar->Administrar Plugins->Tab:Configuración Avanzada).

```
Server: proxydesa.gobiernocba.gov.ar Port: 8080
User name:
Password:
No Proxy Host:
              *.test.cba.gov.ar
              *.cba.gov.ar
              *.gobiernocba.gov.ar localhost
              127.0.0.1
```

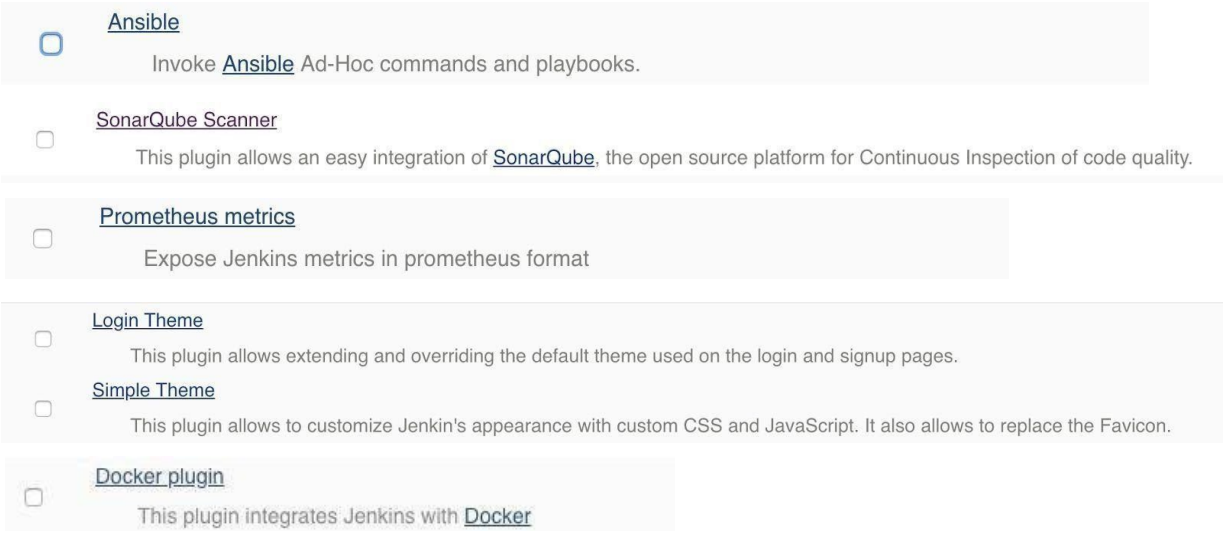
7. Ingresar a la opción “select plugins to install”. Seleccionar los siguientes plugins y luego instalar:



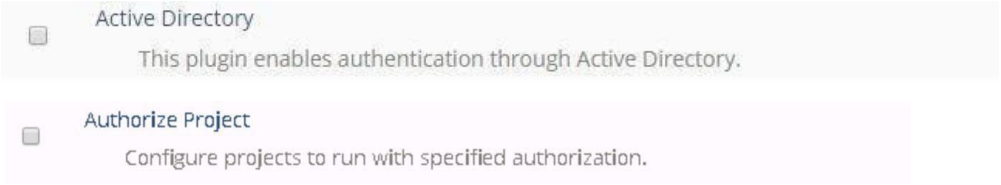
8. Reiniciar contenedor jenkins como usuario de grupo docker ejecutar el comando:

```
docker restart id_contenedor
```

9. Configurar Plugins.
    a. Ingresar nuevamente a Jenkins para instalar los siguientes plugins (Jenkins->Administrar->Administrar Plugins->Tab:Todos) y marcar el checkbox para reiniciar



*El nombre del plugin puede diferir (ej. Docker)



10. Agregar Nuevo Dominio de Credenciales (Jenkins->Credentials->System->Add Domain), nombrarla como `Turnero`.
11. Crear credenciales de registry.

```
Credentials:
      Domain: Turnero
      Kind: Username with
      password Scope:
      Global
      ID: docker-registry-credentials
      Description: Credenciales para publicar en
      Registry Username: reg_turnero
      Password: [ingresar la pwd del paso 15]
```

12. Crear credenciales de despliegue.

```
Credentials:
Domain: Turnero
Kind: SSH Username with private key Scope: Global
ID: jenkins-deploy-user
Description: Jenkins user used to deploy into containers
Username: srv_deploy
Enter Directly: checked
Private Key: pegar el contenido del certificado privado generado en
pasos anteriores ansible_rsa.
Passphrasse: Dejar vacio.
```

13. Configurar Conexion Nodo de Construcción en Docker (Jenkins→Administrar Jenkins→Configurar sitema)
    a. Sección SSH Remote host
       Configurar host con los datos provisto por gobierno
       Hostname
       port
       Credentials
       Probar conexión con el boton "Check Conexion"

14. Configurar los nodos de construcción (Jenkisn→Administrar Jenkins→Configurar Nodo→Nuevo Nodo)
    a. Colocar nombre del Nodo
    b. Tildar "Permanent Agent"
    c. "ok"
    d. Completar los datos solicitados en la pantalla
    i. Nombre: docker_builder_n°
    ii. Descripción: Docker builder
    iii. Directorio raíz: / *home / sa_turnero*
    iv. *Usar: "Utilizar este nodo tanto como sea posible"*
    v. *Método de ejecución: "Arrancar agentes remotos en maquinas Unix via SSH"*
       - *Nombre de maquina: d250lxcpr01.gobiernocba.gov.ar*
       - *Credentials: sa_turnero/******(Conex to Jenkins Builder)*
       - Host Key Verification Strategy: Known host file Verification Strategy
       - Click en "Avanzado"
       - Puerto: 2222

Disponibilidad: keep this agent online as much as possible

15. Configurar TFS (Jenkins->Manage Jenkins->Configure System)

h. Configurar TFS/Team Services. Agregar nueva colección.

```
Collection URL:
http://tfs.gobiernocba.gov.ar/tfs/Innovación%20-%20Proyectos%202017

Agregar Credenciales (si no permite la creación, aplicar el
cambio y refrescar jenkins). Del combo seleccionar la
opción Jenkins.
Domain: Turnero
Kind: Username with
      password
      Scope: Global
      Username:
      svc_turnero_project
      Password:
      ID: turnero-tfs-user
      Description: Usuario de servicio para acceso a
TFS en el proyecto
                    turnero con permisos de checkout únicamente.
```

16. Configurar Jobs de compilación (Jenkins->New Item)

a. Definir el nombre del job a crear como Turnero-Build-Application-DEV de tipo Pipeline project.

```
Description: Compila y despliega la solución de turnero en docker registry
y en el cluster de swarm.
Setear  Do not allow concurrent builds This project is parameterized:
checked
```

i. String Parameter 1: PROXY_SERVER
Default Value: http://proxydesa.gobiernocba.gov.ar:8080
Trim text: Checked

ii. String Parameter 2: DOCKER_REGISTRY
Default Value: registry-turnero-dev.test.cba.gov.ar
Trim text: Checked

iii. String Parameter 3: ENVIRONMENT
Default Value: Desarrollo
Trim text: Checked

iv. String Parameter 4: ENVIRONMENT_SUFFIX
Default Value: -dev.test
Trim text: Checked

v. String Parameter 5: SUPPORT_HOST
Default Value: 10.250.52.3

Trim

text:
Checked

Si Jenkins está en español, seleccionar "Parámetro de Cadena", NO "Texto" NI "Text" (Inglés).

Pipeline:

i. Definition: Pipeline script from SCM
ii. SCM: Team Foundation Version Control (TFVC)
iii. Collection URL: http://tfs.gobiernocba.gov.ar/tfs/Innovación%20-%20Proyectos%20 2017
iv. Project Path: $/Turnero v2 - Soporte/Turnero
v. Credentials: Automatic
vi. Script Path: Jenkinsfile
vii. Lightweight checkout: not checked

b. Definir el nombre del job a crear como Turnero-Monitoring-DEV de tipo Pipeline project.

Description: Compila y despliega la solución de traceo del turnero en docker registry.
Setear Do not allow concurrent builds This project is parameterized: checked

i. String Parameter 1: PROXY_SERVER
Default Value: http://proxydesa.gobiernocba.gov.ar:8080 Trim text: Checked
ii. String Parameter 2: DOCKER_REGISTRY
Default Value: registry-turnero-dev.test.cba.gov.ar Trim text: Checked
iii. String Parameter 3: ENVIRONMENT Default Value: Development Trim text: Checked
iv. String Parameter 4: ENVIRONMENT_SUFFIX Default Value: -dev.test Trim text: Checked

Si Jenkins está en español, seleccione "Parámetro de Cadena", NO "Texto" NI "Text" (Inglés).

```
Pipeline:
```

i. `Definition: Pipeline script from SCM`
ii. `SCM: Team Foundation Version Control (TFVC)`
iii. `Collection URL: http://tfs.gobiernocba.gov.ar/tfs/Innovación%20-%20Proyectos%20 2017`
iv. `Project Path: $/Turnero v2 - Soporte/Turnero`
v. `Credentials: Automatic`
vi. `Script Path: Jenkinsfile-Supporting-Monitoring`
vii. `Lightweight checkout: not checked`

17. Configurar Seguridad de Jenkins
    a. Configurar acceso a los usuarios por Active Directory.
    Activar la configuración de Active Directory (Jenkins->Manage Jenkins->Configure Global Security->Security).
    Seleccionar Active Directory y completar los siguientes datos:

```
Active Directory
        Domain Name:
        gobiernocba.gov.ar Domain
        controller:
        10.250.1.51:389 Site:
        Bind DN:
cn=svc_itop,cn=users,dc=gobiernocba,dc=
gov,dc=ar
        Bind Password:
        TLS Configuration: Confiar en todos los certificados
```

Repetir estos pasos para los siguientes Domain Controllers: 10.250.1.50 y 10.250.1.122.

Antes de aplicar/guardar cambios asegurarse de tener un usuario administrador global con todos los permisos. Chequear la opción “Cualquiera puede realizar cualquier operación” en sección “Autorización”.

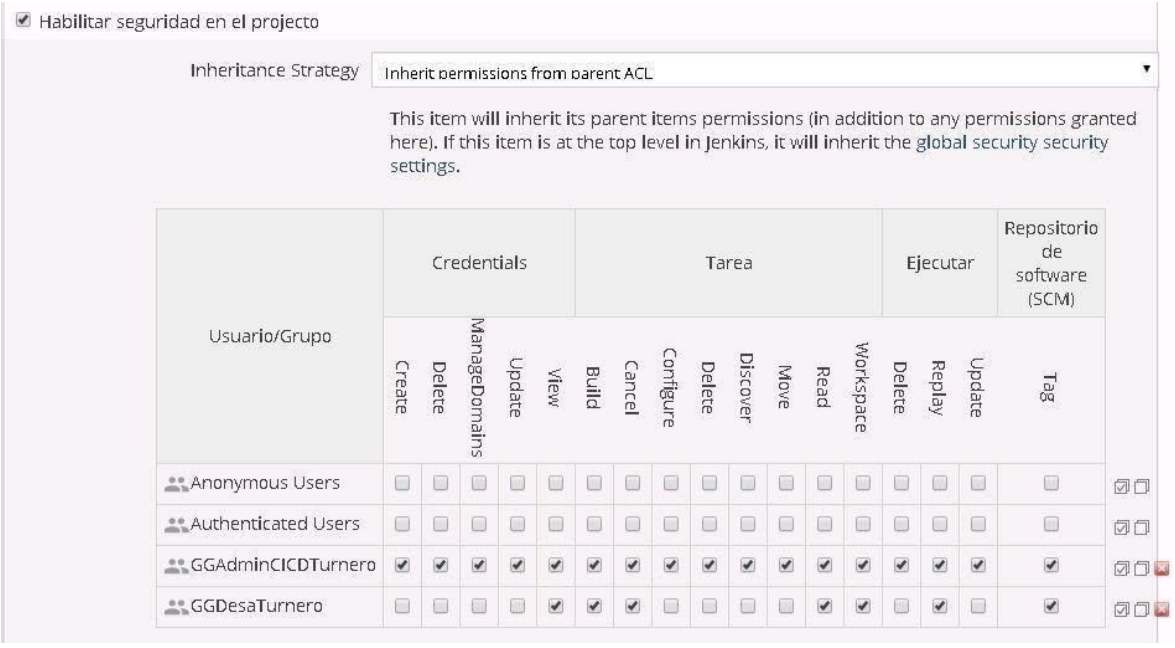
Activar Caché y recién después de esto aplicar los cambios.

b. Configurar Seguridad Basado en proyectos (Manage Jenkins->Configure Global Security->Authorization->Project-based Matrix Authorization Strategy)

Para el caso de turnero dos grupos son requeridos GGDesaTurnero y GGAdminCICDTurnero utilizando los permisos como se describe en la imagen de arriba.

Antes de aplicar asegurarse de que haya un usuario que sea administrador global. Preferentemente usar un usuario concreto que luego puede ser removido.

c. Actualizar los permisos de los Jobs de Jenkins. Entrar a la configuración del job (Jenkins->Turnero-Build-DEV->Configurar y Jenkins->Turnero-Monitoring-DEV) y habilitar la opción "Habilitar seguridad en el proyecto". Dos grupos son requeridos GGDesaTurnero y GGAdminCICDTurnero utilizando los permisos como se describe en la imagen más abajo.



Aplicar/guardar cambios.

d. Configurar ACL para Builds. Dentro de la configuración de Control de Acceso (Jenkins->Administrar Jenkins->Global Security->Access Control for Builds), agregar una nueva estrategia de tipo "Project default Build Authorization":

Strategy: Run as User who Triggered Build.

Aplicar/guardar cambios.

Desplegar Aplicaciones Turnero

Este proceso de despliegue incluye solo los nodos identificados como *swarm cluster node* (`d250lxcde34.gobiernocba.gov.ar, d250lxcde35.gobiernocba.gov.ar`). Con el usuario root:

1. Crear el subdirectorio de logs de la aplicación

```
mkdir -p
/var/log/turnero/web;
mkdir -p
/var/log/turnero/api;
mkdir -p
/var/log/turnero/redis;
mkdir -p
/var/log/turnero/hub;

chown -R sa_turnero /var/log/turnero/
```

2. Crear el subdirectorio de proyecto /opt/turnero. Ajustar los permisos de usuario al usuario de servicio del entorno de ser necesario.

```
mkdir -p /opt/turnero
chown sa_turnero.sa_turnero
/opt/turnero chmod 775
/opt/turnero
```

3. Crear el subdirectorio para la persistencia de Redis

```
mkdir -p /mnt/cdata/volumes;
mkdir -p /mnt/cdata/volumes/turnero;
mkdir -p /mnt/cdata/volumes/turnero/redis;
chown -R sa_turnero:sa_turnero /mnt/cdata/volumes;
```

4. Copiar el certificado wildcard (provisto por infraestructura al momento del despliegue, `wildcard_test_cba_pkey.key` y `wildcard_test_cba_chain.pem`):

```
cp /home/sysoper/wildcard_test_cba*
   /opt/turnero/ cd /opt/turnero
chown sa_turnero.sa_turnero
   wildcard_test_cba* sudo su -
   sa_turnero
cd /opt/turnero
ln -s wildcard_test_cba_pkey.key cba_Development_pkey.key
ln -s wildcard_test_cba_chain.pem
   cba_Development_chain.pem chmod 640
   wildcard_test_cba*
```

5. Probar logeo al registry en cada nodo de solución aplicativa con usuario devuser:

```
docker login -u reg_turnero registry-turnero-dev.test.cba.gov.ar
```

6. Hacer una compilación en Jenkins (Jenkins->Turnero-Build-DEV->Build with parameters) para que genere las imágenes docker de la solución aplicativa y las publique en la registry para posteriores despliegues. Este paso debe finalizar exitosamente antes de proseguir con cualquier despliegue.

Nota: En caso de fallar la compilación habrá que renombrar el workspace cambiando de Hudson a Jenkins
(Jenkins->Turnero-Build-DEV->Configurar->Pipeline->Avanzado->Workspace Name).

Despliegue Manual

1. Desplegar aplicaciones. Desde el primer nodo del cluster de swarm:

```
cd /opt/turnero

docker stack deploy --with-registry-auth --compose-file
docker-compose.yml turnero
```

2. Verificar que los componentes estén desplegados

```
docker service ps turnero
```

Procedimientos Pos-Implementación

Agregar o Reemplazar un Nuevo Nodo

A continuación se describen los pasos a seguir en caso que un nuevo nodo deba ser agregado al cluster de la aplicación:

a. Agregar el nuevo server al cluster siguiendo los pasos definidos en Desplegar Aplicaciones Turnero.

b. Actualizar la lista de servidores para el monitoreo. Edit

`/opt/turnero/monitoring/prometheus/prometheus/prometheus.yml`

para incluir el nombre del nuevo host. Los nombres de los jobs afectados son

- job_name: 'turnero.api'
Todos los nodos escuchan en el puerto 5000

- job_name: 'turnero.hub'
Todos los nodos escuchan en el puerto 5002

- job_name: 'turnero.cache'
Todos los nodos escuchan en el puerto 9121

- job_name: 'cadvisor'
Todos los nodos escuchan en el puerto 8080

- job_name: 'node-exporter'
Todos los nodos escuchan en el puerto 9100

Actualizar Certificados HTTPS por vencimientos

i. En el nodo de soporte (d250lxcde36). Ejecutar como root:

Opcionalmente para preservar el certificado wildcard vencido antes de copiar el nuevo (la extensión "old" es sólo a modo de ejemplo):

```
cd /opt/turnero/gateway/nginx/ssl
mv wildcard_test_cba_pkey.key
wildcard_test_cba_pkey.key.old mv
wildcard_test_cba_chain.pem
wildcard_test_cba_chain.pem.old
```

Luego copiar el certificado wildcard (provisto por infraestructura al momento del despliegue, `wildcard_test_cba_pkey.key` y `wildcard_test_cba_chain.pem`):