

# Informe de Servidor ETCD

---

Nelson Ricardo Baquero

10 de septiembre de 2021

## 1. SERVIDOR DE CONFIGURACIÓN ETCD



Figura 1.1: Logo ETCD.

El etcd es un almacén de clave-valor open source, distribuido y uniforme que permite la configuración compartida, la detección de servicios y la coordinación del programador de clústeres o sistemas distribuidos de máquinas. Facilita la ejecución de actualizaciones automáticas de forma más segura, coordina el trabajo que se programa para los hosts e interviene en la configuración de las redes superpuestas para los contenedores.

## 2. INSTRUCCIONES DE INSTALACIÓN

Las instrucciones de instalación son válidas para la mayoría de distribuciones de Linux. Se utilizó **Fedora** en las capturas.

### 2.1. Crear un script de instalación

Crear un archivo llamado *etcd-install.sh* con las siguientes instrucciones:

---

```
1 ETCD_VER=v3.5.0
2 ETCD_BIN=/tmp/test-etcd
3
4 GOOGLE_URL=https://storage.googleapis.com/etcd
5 GITHUB_URL=https://github.com/etcd-io/etcd/releases/download
6
7 DOWNLOAD_URL=${GITHUB_URL}
8
9 rm -f /tmp/etcd-${ETCD_VER}-linux-amd64.tar.gz
10 rm -rf $ETCD_BIN
11 mkdir -p $ETCD_BIN
12
13 curl -L ${DOWNLOAD_URL}/${ETCD_VER}/etcd-${ETCD_VER}-linux-amd64.tar.gz \
14     -o /tmp/etcd-${ETCD_VER}-linux-amd64.tar.gz
15
16 tar xzvf /tmp/etcd-${ETCD_VER}-linux-amd64.tar.gz -C $ETCD_BIN \
17     --strip-components=1
```

---

Listing 1: Contenido del archivo etcd-install.sh

Finalmente, otorgarle permiso de ejecución `chmod +x etcd-install.sh`

## 2.2. Ejecutar el script

---

```
$ ./etcd-install.sh
```

---

Listing 2: Ejecución del script de instalación.

Si la instalación resultó exitosa, ya es posible levantar el servidor.

## 2.3. Desplegar el servidor ETCD

Para desplegar el servidor, basta con ejecutar el binario (etcd):

---

```
$ ETCD_BIN=/tmp/test-etcd
$ $ETCD_BIN/etcd
```

---

Listing 3: Ejecución del binario.

## 2.4. Asignación de atributos clave/valor

Finalmente, para agregar atributos clave/valor, se debe iniciar otra sesión de terminal y ejecutar el siguiente comando:

---

```
$ $ETCD_BIN/etcdctl put [clave] [valor]
```

---

Listing 4: Ingreso de un nuevo atributo.

Para obtener un valor a partir de su clave en la misma terminal, ejecutar el siguiente comando:

```
$ $ETCD_BIN/etcdctl get [clave]
```

Listing 5: Obtención de un atributo.

### 3. CONFIGURACIÓN EN JAVA

Utilizaremos un proyecto con Gradle como gestor de dependencias.

#### 3.1. Agregar las dependencias necesarias

Se debe agregar las siguientes dependencias:

```
ext {
    helidonVersion = '2.3.2'
}

dependencies {
    implementation platform("io.helidon:helidon-dependencies:${project.helidonVersion}")

    implementation 'io.helidon.micromprofile.server:helidon-micromprofile-server'
    implementation 'org.glassfish.jersey.media:jersey-media-json-binding'
    implementation group: 'org.micromprofile-ext.config-ext', name: 'configsource-etcd', version: '2.0.1'

    runtimeOnly 'org.jboss:jandex'
}
```

Figura 3.1: Sección de dependencias en build.gradle

Además, debido a la antigüedad de una dependencia de la librería de configuración, se debe forzar a gradle a utilizar una versión anterior de la dependencia utilizando el siguiente código:

```
configurations.all {
    resolutionStrategy {
        force 'io.grpc:grpc-core:1.15.1'
        force 'io.grpc:grpc-netty:1.15.1'
    }
}
```

Figura 3.2: Sección de configuración en build.gradle

#### 3.2. Crear un archivo de configuración para ETCD

El archivo debe tener como nombre `micromprofile-config.properties` y estar ubicado en el directorio META-INF del proyecto.

Dentro del archivo se pueden colocar opciones de configuración como host, puerto, activar o desactivar, entre otras opciones.

#### 3.3. Utilizar los atributos de ETCD en el código

Podemos utilizar CDI para inyectar los atributos. Por lo tanto necesitamos anotar la clase con un ámbito y anotar la propiedad que deseemos inyectar.

---

```
1 @ApplicationScoped // Anotación de ámbito
2 @Path("/")
3 public class HolaMundoRest {
4
5     @Inject // Anotación para inyectar una propiedad.
6     @ConfigProperty(name = "app.mensaje2") // Anotación con la propiedad ETCD
7     private String msg;
8
9     @GET
10    @Path("/hola")
11    public String hola() {
12
13        // Config providers
14        ConfigProvider.getConfig().getConfigSources().forEach(s -> {
15            System.out.printf("%3d: %s\n", s.getOrdinal(), s.getName());
16        });
17
18        return String.format("%s %s", msg, LocalDateTime.now());
19    }
20 }
```

---

Listing 6: Ejemplo de Hola Mundo utilizando configuración de ETCD.