



# Instalar Wildfly (JBoss)

Eduardo Ostertag Jenkins, Ph.D.

OBCOM INGENIERIA S.A. (Chile)

[Eduardo.Ostertag@obcom.cl](mailto:Eduardo.Ostertag@obcom.cl)

# Descargar Server JRE 8u241

The screenshot shows a Microsoft Edge browser window with the URL [oracle.com/java/technologies/javase-server-jre8-downloads.html](http://oracle.com/java/technologies/javase-server-jre8-downloads.html) highlighted by a red box. The main content area displays the title "Server JRE (Java SE Runtime Environment) 8u241" also highlighted by a red box. Below it, a sub-header states: "This software is licensed under the Oracle Technology Network License Agreement for Oracle Java SE". A table lists four download options:

Product / File Description	File Size	Download
Linux x64	52.8 MB	<a href="#">server-jre-8u241-linux-x64.tar.gz</a>
Solaris SPARC 64-bit	63.97 MB	<a href="#">server-jre-8u241-solaris-sparcv9.tar.gz</a>
Solaris x64	61.75 MB	<a href="#">server-jre-8u241-solaris-x64.tar.gz</a>
Windows x64	48.92 MB	<a href="#">server-jre-8u241-windows-x64.tar.gz</a>

# Descargar Wildfly 19.0.0.Final

Screenshot of a web browser showing the Wildfly download page at [wildfly.org/downloads/](http://wildfly.org/downloads/). The browser address bar is highlighted with a red box.

The page displays a table of available Wildfly distributions:

Version	Date	Description	License	Size	Format	Checksum
19.0.0.Final	2020-03-18	Java EE Full & Web Distribution	LGPL	181 MB	<a href="#">ZIP</a>	SHA-1
				180 MB	<a href="#">TGZ</a>	SHA-1
		Servlet-Only Distribution	LGPL	43 MB	<a href="#">ZIP</a>	SHA-1
				42 MB	<a href="#">TGZ</a>	SHA-1
		Application Server Source Code	LGPL	36 MB	<a href="#">ZIP</a>	SHA-1
				23 MB	<a href="#">TGZ</a>	SHA-1
		Quick Start Source Code	AL		<a href="#">Source</a>	
		Release Notes			<a href="#">Notes</a>	

A yellow callout box points to the [TGZ](#) link for the "Servlet-Only Distribution" row, with the text "wildfly-19.0.0.Final.tar.gz".

On the right side of the page, there are two status icons: "Supported by Tools" (blue icon) and "Tested with Arquillian" (green icon).

# Extraer archivos \*.tar.gz

```
# cd /opt
#
# gtar zxf server-jre-8u241-linux-x64.tar.gz
# chown -R root.root jdk1.8.0_241
#
# gtar zxf wildfly-19.0.0.Final.tar.gz
# chown -R root.root wildfly-19.0.0.Final
#
# ls -l
total 238476
drwxr-xr-x.  7 root root      166 Dec 11 07:35 jdk1.8.0_241
-rw-r--r--.  1 root root 55367716 Apr  2 08:53 server-jre-8u241-linux-x64.tar.gz
drwxr-xr-x. 11 root root     239 Mar 17 20:00 wildfly-19.0.0.Final
-rw-r--r--.  1 root root 188826408 Apr  2 08:43 wildfly-19.0.0.Final.tar.gz
#
```

# Editar standalone.conf

```
# cd /opt/wildfly-19.0.0.Final/bin
#
# vi standalone.conf
...
# Specify the location of the Java home directory.
JAVA_HOME="/opt/jdk1.8.0_241"
...
# Specify options to pass to the Java VM.
JAVA_OPTS="$JAVA_OPTS -Xms128m -Xmx1024m ..."
...
# Change Wildfly listen ports.
JAVA_OPTS="$JAVA_OPTS -Djboss.socket.binding.port-offset=100"
...
#
```

8080 + 100 = 8180	http
8443 + 100 = 8543	https
9990 + 100 = 10090	management-http
9993 + 100 = 10093	management-https
8009 + 100 = 8109	ajp
4712 + 100 = 4812	txn-recovery-env
4713 + 100 = 4813	txn-status-manager

# Editar standalone.xml

```
# cd /opt/wildfly-19.0.0.Final/standalone/configuration
#
# vi standalone.xml
...
<interfaces>
    <interface name="management">
        <inet-address value="${jboss.bind.address.management:127.0.0.1}">
            <any-address/>
        </interface>
    <interface name="public">
        <inet-address value="${jboss.bind.address:127.0.0.1}">
            <any-address/>
        </interface>
    </interfaces>
...
#
```

# Iniciar servidor Wildfly

```
# cd /opt/wildfly-19.0.0.Final/bin  
#  
# ./standalone.sh  
09:34:37,325 INFO  [org.jboss.modules] (main) JBoss Modules version 1.10.0.Final  
...  
09:34:46,182 INFO  [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: ...  
...consola bloqueada con Wildfly ejecutado. Para detenerlo presionar ^C...
```

# Definir cuenta administrador

```
# cd /opt/wildfly-19.0.0.Final/standalone/configuration  
#  
# /opt/wildfly-19.0.0.Final/bin/add-user.sh  
  
Username : admin  
  
Password : adminadmin  
  
What groups do you want this user to belong to? blank  
  
Is this new user going ... AS process to connect to another AS process? no  
#
```

# Consola HTML Wildfly

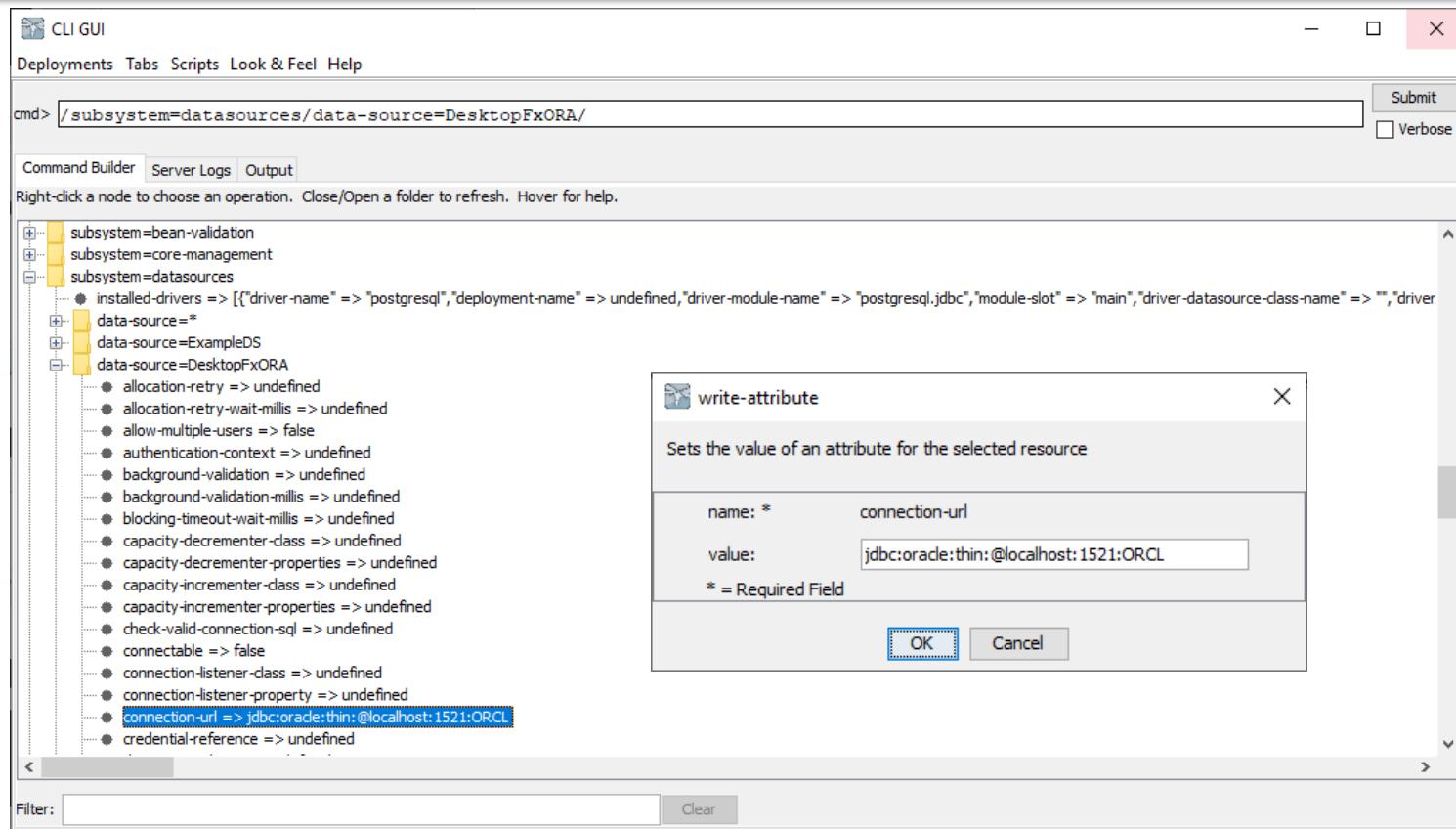
The screenshot shows the HAL Management Console interface at [localhost:9990/console/index.html](http://localhost:9990/console/index.html). The top navigation bar includes links for Home, Deployments, Configuration, Runtime, Patching, and Access Control. The main content area features six cards: Deployments, Configuration, Runtime, Access Control, Patching, and Need Help.

- Deployments**: Add and manage deployments. Includes a 'Deploy an Application' section with steps 1. Use the 'Add Deployment' wizard to deploy the application and 2. Enable the deployment.
- Configuration**: Configure subsystem settings. Includes a 'Create a Datasource' section with steps 1. Select the Datasources subsystem, 2. Add a Non-XA or XA datasource, and 3. Use the 'Create Datasource' wizard to configure the datasource settings.
- Runtime**: Monitor server status. Includes a 'Monitor the Server' section with steps 1. Select the server and 2. View log files or JVM usage.
- Access Control**: Manage user and group permissions for management operations. Includes an 'Assign User Roles' section with steps 1. Add a new user or group and 2. Assign one or more roles to that user or group.
- Patching**: Manage WildFly Full patches. Includes an 'Apply a Patch' section with steps 1. Download the patch file to the local machine and 2. Use the 'Apply Patch' wizard to select and apply the patch.
- Need Help?**: A summary of resources for help. It lists General Resources (WildFly Home, WildFly Documentation, Model Reference Documentation, Browse Issues, Latest News) and Get Help resources (Access tutorials and quickstarts, User Forums, IRC, Developers Mailing List).

At the bottom, there are links for 3.2.1.Final, Tools, and Settings.

# Consola avanzada Wildfly

```
# JAVA_HOME=/usr/java/jdk1.8.0_241 \
/opt/wildfly-19.0.0.Final/bin/jboss-cli.sh \
--gui --connect --controller=localhost:9990
```



# Driver JDBC (MySQL)

```
# JAVA_HOME=/usr/java/jdk1.8.0_241 \
/opt/wildfly-19.0.0.Final/bin/jboss-cli.sh \
--connect --controller=localhost:9990 --file=add_mysql_driver.txt
```

## add\_mysql\_driver.txt

```
batch
```

```
module add \
--name=mysql.jdbc \
--slot=main --resource-delimiter=; \
--resources=mysql-connector-java-8.0.19.jar \
--dependencies=javax.api,javax.transaction.api

/subsystem=datasources/jdbc-driver=mysql:add( \
    driver-name="mysql", \
    driver-module-name="mysql.jdbc", \
    driver-class-name=com.mysql.cj.jdbc.Driver \
)

run-batch
```

# Driver JDBC (Oracle)

```
# JAVA_HOME=/usr/java/jdk1.8.0_241 \
/opt/wildfly-19.0.0.Final/bin/jboss-cli.sh \
--connect --controller=localhost:9990 --file=add_oracle_driver.txt
```

## add\_oracle\_driver.txt

```
batch

module add \
--name=oracle.jdbc \
--slot=main --resource-delimiter=; \
--resources=ojdbc8.jar;orai18n.jar \
--dependencies=javax.api,javax.transaction.api

/subsystem=datasources/jdbc-driver=oracle:add( \
    driver-name="oracle", \
    driver-module-name="oracle.jdbc", \
    driver-class-name=oracle.jdbc.OracleDriver \
)

run-batch
```

# Driver JDBC (PostgreSQL)

```
# JAVA_HOME=/usr/java/jdk1.8.0_241 \
/opt/wildfly-19.0.0.Final/bin/jboss-cli.sh \
--connect --controller=localhost:9990 --file=add_postgresql_driver.txt
```

## add\_postgresql\_driver.txt

```
batch

module add \
--name=postgresql.jdbc \
--slot=main --resource-delimiter=; \
--resources=postgresql-42.2.11.jar \
--dependencies=javax.api,javax.transaction.api

/subsystem=datasources/jdbc-driver=postgresql:add( \
    driver-name="postgresql", \
    driver-module-name="postgresql.jdbc", \
    driver-class-name=org.postgresql.Driver \
)

run-batch
```

# Driver JDBC (MS-SQL)

```
# JAVA_HOME=/usr/java/jdk1.8.0_241 \
/opt/wildfly-19.0.0.Final/bin/jboss-cli.sh \
--connect --controller=localhost:9990 --file=add_mssql_driver.txt
```

## add\_mssql\_driver.txt

```
batch

module add \
--name=sqlserver.jdbc \
--slot=main --resource-delimiter=; \
--resources=mssql-jdbc-8.2.1.jre8.jar \
--dependencies=javax.api,javax.transaction.api

/subsystem=datasources/jdbc-driver=sqlserver:add( \
    driver-name="sqlserver", \
    driver-module-name="sqlserver.jdbc", \
    driver-class-name=com.microsoft.sqlserver.jdbc.SQLServerDriver \
)

run-batch
```

# DataSources en la consola

The screenshot shows the HAL Management Console interface. The top navigation bar includes tabs for Homepage, Deployments, Configuration (which is selected), Runtime, Patching, and Access Control. The left sidebar lists various subsystems: Subsystems, Interfaces, Socket Bindings, Paths, and System Properties. Under Subsystems, 'Datasources & Drivers' is expanded, showing options like Batch, JBeret, Core Management, and others. The main content area is titled 'Datasources & Drivers' and shows a list of configured data sources. A specific datasource named 'DesktopFxSQL' is highlighted. The right panel displays detailed information for 'DesktopFxSQL', including its main attributes:

JNDI Name:	java:/jdbc/obcomDesktopFxDataSource
Driver Name:	sqlserver
Connection URL:	jdbc:sqlserver://localhost:1433
Enabled:	true
Statistics:	false
Enabled:	(checkbox)

At the bottom of the page, there are links for 'Tools' and 'Settings'.

# DataSource JDBC (MySQL)

## add\_mysql\_datasource.txt

```
batch
```

```
data-source add \
--name=MyDataSource \
--driver-name=mysql \
--jndi-name=jdbc:/jdbc/MyDataSource \
--connection-url=jdbc:mysql://localhost:3306/schema \
--user-name=<usuario> --password=<contraseña> \
--min-pool-size=0 --max-pool-size=10 \
--initial-pool-size=0 --idle-timeout-minutes=30
```

```
/subsystem=datasources/data-source=MyDataSource \
  /:write-attribute(name=enabled,value=true)
```

```
/subsystem=datasources/data-source=MyDataSource \
  /connection-properties=sslMode:add(value="DISABLED")
```

```
run-batch
```

# DataSource JDBC (Oracle)

## add\_oracle\_datasource.txt

```
batch
```

```
data-source add \
  --name=MyDataSource \
  --driver-name=oracle \
  --jndi-name=java:/jdbc/MyDataSource \
  --connection-url=jdbc:oracle:thin:@localhost:1521:ORCL \
  --user-name=<usuario> --password=<contraseña> \
  --min-pool-size=0 --max-pool-size=10 \
  --initial-pool-size=0 --idle-timeout-minutes=30

/subsystem=datasources/data-source=MyDataSource \
  /:write-attribute(name=enabled,value=true)
```

```
run-batch
```

# DataSource JDBC (PostgreSQL)

## add\_postgresql\_datasource.txt

```
batch
```

```
data-source add \
    --name=MyDataSource \
    --driver-name=postgresql \
    --jndi-name=jdbc:/jdbc/MyDataSource \
    --connection-url=jdbc:postgresql://localhost:5432/schema \
    --user-name=<usuario> --password=<contraseña> \
    --min-pool-size=0 --max-pool-size=10 \
    --initial-pool-size=0 --idle-timeout-minutes=30

/subsystem=datasources/data-source=MyDataSource \
    /:write-attribute(name=enabled,value=true)
```

```
run-batch
```

# DataSource JDBC (MS-SQL)

## add\_mssql\_datasource.txt

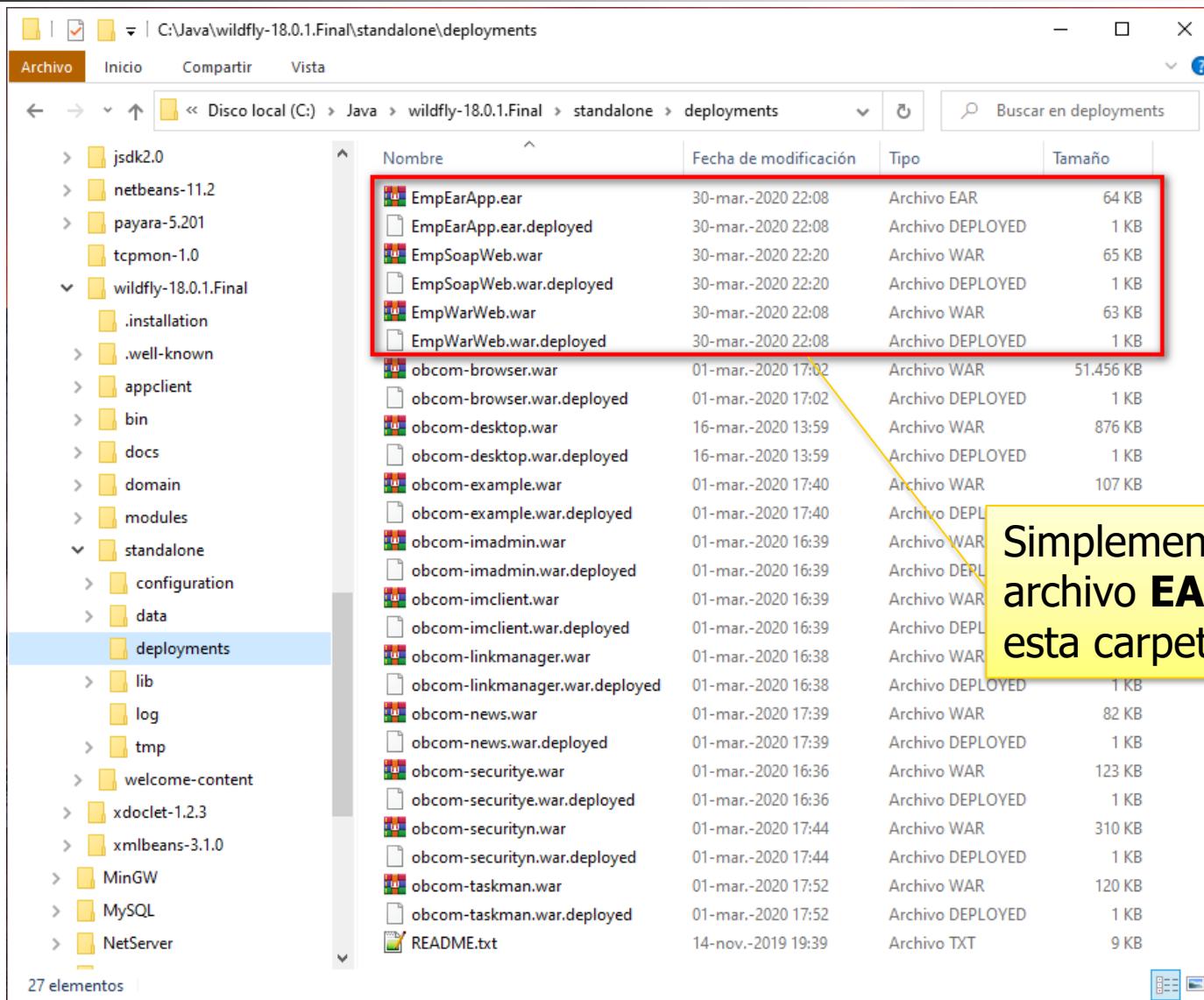
```
batch
```

```
data-source add \
--name=MyDataSource \
--driver-name=sqlserver \
--jndi-name=java:/jdbc/MyDataSource \
--connection-url=jdbc:sqlserver://localhost:1433;database=DBNAME \
--user-name=<usuario> --password=<contraseña> \
--min-pool-size=0 --max-pool-size=10 \
--initial-pool-size=0 --idle-timeout-minutes=30

/subsystem=datasources/data-source=MyDataSource \
/:write-attribute(name=enabled,value=true)
```

```
run-batch
```

# Auto instalación en Wildfly



# Configurar HTTPS (1)

- Crear archivo “**application.keystore**” (JKS)
  - Debe contener un solo certificado público “alias”
  - Debe contener la llave privada (key) de “alias”
- Copiar “**application.keystore**” al directorio
  - /opt/wildfly-19.0.0.Final/standalone/configuration
- Editar el archivo “**standalone.xml**”
  - *...ver detalles en la próxima diapositiva...*

# Configurar HTTPS (2)

...

```
<security-realm name="ApplicationRealm">  
    <server-identities>  
        <ssl>  
            <keystore path="application.keystore"  
                relative-to="jboss.server.config.dir"  
                keystore-password="<contraseña-del-archivo>"  
                alias="<alias-del-certificado>"  
                key-password="<contraseña-de-la-llave>"/>  
        </ssl>  
    </server-identities>  
</security-realm>
```

...

**standalone.xml**

# Definir servicio Wildfly (1)

- Copiar archivo “**wildfly.service**” a...
  - /usr/lib/systemd/system/wildfly.service
- Administrar servicio Wildfly con...
  - # systemctl start wildlfy                   *Iniciar*
  - # systemctl stop wildlfy                   *Detener*
  - # systemctl status wildlfy               *Ver estado*
  - # systemctl enable wildlfy               *Habilitar*
  - # systemctl disable wildlfy              *Deshabilitar*

# Definir servicio Wildfly (2)

```
[Unit]
Description=WildFly 19.0.0 service

[Service]
Type=forking
WorkingDirectory=/opt/wildfly-19.0.0.Final/standalone
ExecStart=/opt/wildfly-19.0.0.Final/bin/standalone.sh -b=0.0.0.0 -bmanagement=0.0.0.0
ExecStop=/opt/wildfly-19.0.0.Final/bin/jboss-cli.sh --connect controller=localhost:9990
command=/:shutdown
User=wildfly
Group=wildfly

[Install]
WantedBy=multi-user.target
```

wildfly.service

```
# firewall-cmd --get-active-zones  
  
# firewall-cmd --get-default-zone  
  
# firewall-cmd --zone=public --list-ports  
  
#  
  
# firewall-cmd --zone=public --add-port=8080/tcp --permanent  
  
# firewall-cmd --zone=public --add-port=8443/tcp --permanent  
  
# firewall-cmd --zone=public --add-port=9990/tcp --permanent  
  
# firewall-cmd --zone=public --add-port=9993/tcp --permanent
```

Muchas gracias

Muchas  
gracias