LOCAL LICENSE SERVER

Administrator Guide for Linux
2020.07

Copyright

Terms of Use

Proprietary Statement

This manual contains proprietary information of Zebra Technologies Corporation and its subsidiaries ("Zebra Technologies"). It is intended solely for the information and use of parties operating and maintaining the equipment described herein. Such proprietary information may not be used, reproduced, or disclosed to any other parties for any other purpose without the express, written permission of Zebra Technologies.

Product Improvements

Continuous improvement of products is a policy of Zebra Technologies. All specifications and designs are subject to change without notice.

Liability Disclaimer

Zebra Technologies takes steps to ensure that its published Engineering specifications and manuals are correct; however, errors do occur. Zebra Technologies reserves the right to correct any such errors and disclaims liability resulting therefrom.

Limitation of Liability

In no event shall Zebra Technologies or anyone else involved in the creation, production, or delivery of the accompanying product (including hardware and software) be liable for any damages whatsoever (including, without limitation, consequential damages including loss of business profits, business interruption, or loss of business information) arising out of the use of, the results of use of, or inability to use such product, even if Zebra Technologies has been advised of the possibility of such damages. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

Table of Contents

Introduction

The Zebra Technologies' Embedded Local License Server acts as on premise server to support node-locked site-wide licensing. The license server is designed to administer the licenses within a customer enterprise, report license usage to the back office, and provide served-license status information.

This guide describes how to administer the Zebra Technologies' Embedded Local License Server.

NOTE Screens and windows pictured in this guide are samples and can differ from actual screens.

Contact

Please feel free to send your queries on this guide to ZebraSWLicensingTeam@zebra.com

Getting Started

Local License Server Requirements

Hardware Requirements

Minimum hardware requirements for the license server:

- Hard drive: 500 MB
- RAM: 4 GB
- CPU: 2 GHz, two cores.

Supported Platforms

The Embedded local license server is supported on the following platforms:

- Linux x86/x86-64
    - Ubuntu 16.04 and higher
    - CentOS 6.0
    - CentOS 7.0
    - Red Hat Enterprise Linux 7.x.

Virtual Machine Support

** Will be included in the future releases based on the request from customers/partners. Virtualization functionality in the embedded local license server supports the following:

- VMware ESXi 6.0
- VMware Workstation 12.5
- Microsoft Hyper-V 6.3 on Windows Server 2016
- Microsoft Hyper-V 6.1 on Windows 8
- Citrix XenServer 7.0
- Oracle VirtualBox 5.1.12
- QEMU-KVM 2.7
- Parallels 12.1
- Google Compute Cloud
- Amazon EC2
- Microsoft Azure (Windows only).

Supported Browsers

The License Server Manager UI supports the following browsers:

- Mozilla Firefox version 43 or higher
- Google Chrome version 47 or higher
- Microsoft Internet Explorer version 10 or higher
- Microsoft Edge version 38.1.

Java Prerequisites

The following are the Java prerequisites for the machine where the embedded local license server is installed:

- Oracle Java SE 1.8 or Java SE 11, or OpenJDK 1.8/1.9 or OpenJDK 11 (must be 64-bit)
- The JAVA_HOME (or JRE_HOME) environment variable on your system set to the path for your default JDK (or JRE) installation.

NOTE: The license server requires only the JRE component. If JRE is your default Java installation, set the JRE_HOME environment variable; if JDK is your default installation, set JAVA_HOME.

License Server Manager Requirements

The License Server Manager requires installation of Apache Tomcat. Supported versions include:

- 7x (7.0.53 or higher)
- 8x (for 8.5.x, should be 8.5.16 or higher)
- 9x (for 9.0).

DNS Configuration Requirements (Optional)

Local License Server (LLS) exposed URL used by devices for license management (activate/return) needs to be constant and should not change after the LLS is setup and running.

By default the URL exposed by LLS is in the format `http://10.80.204.154:7070/request`, where 10.80.204.154 is the IP address of system on which the LLS is running.

Since the IP of a system can change depending on the network configuration, it is advised to setup a DNS server.

So, the URL format with DNS setup will be like `http://llicenseserver.zebra.com:7070/request`.

Connectivity Details

The Local License server is acquiring the licenses from the cloud (https://zebra-licensing.flexnetoperations.com) and the port 443 is open for the communication.

Default listening port for the LLS service on the host machine is 7070.

Download Packages

The Zebra Technologies Embedded local license server software package is provided as an executable file that can be downloaded from the zebra.com website under the **Support & Downloads** section. The software package supports both the 32-bit and 64-bit architectures.

- Linux: ZebraLLS_Linux_Installer_v2.0.sh

Administrator Experience on the Local License Server Overview

The following table summarizes the basic tasks to perform as a license server administrator on the LLS.

Table 1 LLS Overview

Phase	Task	Description
1	Un-Wrap the Local License Server Components	The Zebra provided executable script will load the necessary files for the Local Licenser Server and Local License Server Manager in the system.
2	Configuring the Local License Server	Following initial registration of the license server, you have the option to configure settings that define the local environment in which the license server will run. (These settings can be edited only after the service is initially registered as a Linux service.) Example settings: Active Host ID/Proxy Settings/Port Settings.
3	Registering the Local License Server	Register the server as a device in the End User Licensing Portal.
4	Preparing to Use the License Server Manager	Zebra Technologies has provided the License Server Manager as a tool with which to administer your license server. If so, before launching this tool, you must have an Apache Tomcat server installed and need to perform some additional configuration steps.
5	Start the Local License Server	This phase involves installing and starting up the Zebra local license server as a service.
6	Start the License Server Manager	The administrator tool that the producer provides with the license server enables you to manage and monitor the server and its operations.
7	Acquire Licenses on the License Server	A purchased set of product licenses needs to be acquired on the license server before it can distribute the licenses to client devices running the licensed products. The licenses can be assigned from the Zebra Portal or through the Offline Server Updates View.
--	Uninstalling the Local License Server	For various reasons, you might need to uninstall the LLS.

Un-Wrap the Local License Server Components

Running the executable shell script file with the root access or as a sudo user-provided by the Zebra Technologies (`ZebraLLS_Linux_Installer_v2.0.sh`) will un-wrap the components required for the Local Licensing Server and place it in a default or user defined directory.

Default Installation Location is drive: /usr/share. The following folders are available under the ZebraLLS directory:

- lib
- server
- service
- ui.

⚠️ **CAUTION** The server folder contains multiple producer-settings.xml files - these files are to never be tampered with as it is a Zebra generated and encrypted file. Any edits made to the file result in errors requiring a fresh install to fix.

Java Runtime Environment (JRE)

The LLS requires the Java Runtime Environment component to be installed in the system. If it is not installed, download one of the listed compatible Java versions on the machine to host the LLS.

After installing the Java on the Linux system, set the JAVA_HOME system variable by opening the file /etc/environment with any text editor and add the following line (example is for OpenJDK 1.8):

```
JAVA_HOME="/usr/lib/jvm/java-1.8.0-openjdk-amd64"
```

Once you save and exit the file, use the following to load the variables:

```
source /etc/environment
```

📝 **NOTE:** Usually most Linux systems source /etc/environment by default. If your system does not, then add the above line to ~/.bashrc to ensure that system variables are always loaded. Otherwise, you will need to source your environment variables per terminal session.

These commands are based on java being installed in its default directory (usr/lib/jvm).

Apache TomCat

To make use of the License Server Manager, Apache TomCat must be installed in the system. If it is not installed, download the latest version available for download online (all download versions of Apache can be found here).

After installing Apache TomCat in Linux, copy the **flsm.war** file from the **ui** folder of the LLS installation directory to the **/webapps** directory of TomCat. For example, if Apache TomCat version 8 is installed in the **/var/lib,** the directory is **/var/lib/tomcat{v#}/webapps**.

Configuring the Local License Server

NOTE: All the settings have their default values and it is not necessary to change any of the values unless it is required except for the active-hostid. Refer the end of this section to identify the correct HOSTID.

Configuring in Linux

To use systemd to configure and install the local license server as a Linux service:

1. Files Required for Installation Using systemd
2. Verify systemd-Enablement on Linux System
3. Run the Install Script
4. Manage the Service.

Files Required for Installation Using systemd

The following files, distributed with the LLS, support the installation and running of the license server service under systemd:

- zebrals.sh
- install-systemd.sh
- install-functions.sh
- producer-settings.xml.

Residing in the same directory as zebrals.jar, these files work together to generate the components required to install and start the license server service in a systemd configuration.

Verify systemd-Enablement on Linux System

Before proceeding with the installation, run the following command to determine whether the Linux system on which the license server is being installed is systemd-enabled:

```
systemctl
```

If the system is systemd-enabled, the command runs successfully and lists all active units.

Run the Install Script

Run the script `zebrals.sh` and follow the prompted commands for registration (command Z) to invoke the install-systemd.sh script to create and start the license server service using either the default configuration for the service or a configuration customized through command-line options.

See the following sections for details:

- About the Install Script on page 11
- Install and Start the Service with the Default Configuration on page 11
- Install and Start the Service with a Modified Configuration on page 11
- Configuration Values Editable from the Command Line on page 12.

The commands described in these following sections are run from the directory in which the systemd script files (see About the Install Script on page 11) and zebrals.jar reside.

Prerequisite

You must run the install script as a root or sudo user.

About the Install Script

Using the 'Z' command and following the prompted instructions in the LLS UI script `zebrals.sh` performs the following:

- Generates the service unit file called **/etc/systemd/system/localserver-zebra.service**.

- Generates the configuration unit file called **/etc/systemd/system/localserver-zebra.service.d/zebrals.conf**.

  This file contains the configuration settings needed to start the service. It is created with default values, but you can include options in the install-script command line to generate this file with custom values. See Install and Start the Service with a Modified Configuration on page 11 for more information.

You can also edit this configuration file post-installation as needed, as described in Editing the Local Settings Post - Installation in Linux on page 14.

- Generates a local-configuration.yaml file in the /opt/zebra directory. This file contains optional settings specific to the local service environment. Generally, these settings are initially disabled; edit or enable them as needed once the service is installed. See Edit local-configuration.yaml Settings on page 15 for details.

- By default, enables the standard Syslog process for logging on the LLS. If desired, enable Systemd logging to be used in place of Syslog once the service is installed. For details, see Enable systemd Logging on page 15.

- Starts the license server service.

Install and Start the Service with the Default Configuration

The following command starts the LLS UI so that the user can then install and start the LLS using the default configuration. See Start the Local License Server on page 19 for info on registering and starting the service through the LLS UI started from this command. For more about the default values used to configure the service, see Configuration Values Editable from the Command Line on page 12.

```
sudo ./zebrals.sh
```

Install and Start the Service with a Modified Configuration

The following command installs and starts the license server service using a configuration customized through one or more specified command-line options. For a description of the configuration values that can be updated from the install-script command line, see Configuration Values Editable from the Command Line on page 12.

NOTE: It is recommended that you use the zebrals.sh to register the service. Any changes made to the default configuration should be brought to the attention of the Zebra Software Licensing Team prior to implementation.

When moving forward with an alternate configuration than that provided using the zebrals.sh, run the install script while specifying one or more command-line options to change specific configurations settings (for example, the user value, as shown here):

```
sudo ./install-systemd.sh --user zebrals01
```

Configuration Values Editable from the Command Line

Use the following options with the `./install-systemd.sh` command to edit current configuration settings for the license server service. (Running `./install-systemd.sh --help` also lists these command-line options.)

Table 2   Command Line Options

Setting	Description
--user user_name	The user name under which the service runs.
--group group_name	The group name under which the service runs.
--java_home path or --jre_home path	The path for JDK or JRE installation that the license server should use.
	By default, the license server uses the value of your **JAVA_HOME** or **JRE_HOME** system environment variable, whichever is defined on your device, to determine the Java installation location. However, if you want the license server to use different Java installation on your system, provide the explicit path for the installation as either the java_home or jre_home value. (This override pertains to the license server only; your device in general continues to use the Java installation defined by the system environment variable.)
The following two settings, -"port" and "logging-threshold", do not exist in the `.conf` file. However, if you update either setting using the command-line option, the new value is captured in the `local-configuration.yaml` file during its creation and is subsequently used by the license server service. If neither setting is updated, the service will use the default value (or the value specified in `producer-settings.xml`) for the setting.	
--port port	The listening port used by the license server service (default is 7070).
	If the machine on which the license server is running uses multiple network interfaces, you can use the --port option to specify the interface that you want the license server to use. Simply include the IP address for the interface in square brackets, as shown in this example:
	`--port [127.0.0.1].144`3
--logging-threshold level	The lowest level of log-message granularity to record-FATAL, ERROR, WARN, or INFO, or POLICY. For example, if **FATAL** is set, only messages about fatal events are recorded. However, if **WARN** is set, fatal-event, error, and warning messages are recorded. (Default is INFO.)
	The POLICY threshold records information about the checkout process when feature selectors are used to filter features.

Identifying the Host ID in Linux

Host ID is the identifier for the LLS. To make any configuration edits to set the Active Host ID of the LLS, it is required that the LLS has already been registered one time.

To find the correct HostID to set the LLS, go to **terminal** and type `nmcli dev show or ifconfig -a`.

Identify the Network Connection which shows the Wireless connection with which the devices will be connected.

In the following example, SGuest is the wireless connection with which the devices will be connected. Pick the Hardware Address or GENERAL.HWADDR (without colons) and that will be the Host ID.

Host ID in this example is 648099F2644E.

Figure 1 Host ID for Linux


```
File  Edit  View  Search  Terminal  Help

hales@Subu17:~$ nmcli dev show
GENERAL.DEVICE:                     wlo1
GENERAL.TYPE:                       wifi
GENERAL.HWADDR:                     64:80:99:F2:64:4E
GENERAL.MTU:                        1500
GENERAL.STATE:                      100 (connected)
GENERAL.CONNECTION:                 SGuest
GENERAL.CON-PATH:                   /org/freedesktop/NetworkManager/ActiveConnection/1
IP4.ADDRESS[1]:                     10.80.200.145/22
IP4.GATEWAY:                        10.80.200.1
IP4.ROUTE[1]:                       dst = 0.0.0.0/0, nh = 10.80.200.1, mt = 600
IP4.ROUTE[2]:                       dst = 10.80.200.0/22, nh = 0.0.0.0, mt = 600
IP4.ROUTE[3]:                       dst = 169.254.0.0/16, nh = 0.0.0.0, mt = 1000
IP4.DNS[1]:                         8.8.8.8
IP4.DNS[2]:                         4.2.2.2
IP4.DOMAIN[1]:                      sguest.lan
IP6.ADDRESS[1]:                     fe80::deb9:a335:4cc3:eed8/64
IP6.GATEWAY:                        --
IP6.ROUTE[1]:                       dst = ff00::/8, nh = ::, mt = 256, table=255
IP6.ROUTE[2]:                       dst = fe80::/64, nh = ::, mt = 256
IP6.ROUTE[3]:                       dst = fe80::/64, nh = ::, mt = 600
```

Navigate to the **local-configuration.yaml** file in the **/opt/zebra** directory.

In this file, the Active Host ID entry can be set via the **active-hostid** field.

Editing the file can be done through any text editor like gedit or vim, but also requires that the user perform the edits with root access or as the sudo user. (Remove the # at the front of the line to activate this setting and ensure there is a space between the colon separating your value.)

13

Figure 2   Active Host ID in local-configuration.yaml File

Following this update to the `local-configuration.yaml` file, run the below command in terminal to stop and restart the service to apply the configuration changes or enter "**4**" in the `zebrals.sh` UI:

- `sudo systemctl restart localserver-zebra`

Editing the Local Settings Post - Installation in Linux

The following sections describe how to edit current settings in the `zebrals.conf` and `local-configuration.yaml` files after the license server service is installed:

- [Edit Settings in zebrals.conf](#)
- [Edit the DEFINES Setting on page 15](#)
- [Enable systemd Logging on page 15](#)
- [Edit local-configuration.yaml Settings on page 15](#).

Edit Settings in zebrals.conf

Use the following methods to update configuration settings in the `/etc/systemd/system/localserver-zebra.service.d/zebrals.conf` unit file:

1. Edit the .conf file manually.

    Use the following procedure to manually edit the contents of the zebrals.conf unit file:

    a. Stop the license server service using "**3**" in the zebrals.sh terminal or:

14

```
sudo systemctl stop localserver-zebra
```

b. As a root or sudo user, edit the **/etc/systemd/system/localserver-zebra.service.d/zebrals.conf** file in an editor (vim or nano).

2. Once the edits are complete, run the following command to re-read the systemd unit files to capture the configuration changes:

```
sudo systemctl daemon-reload
```

3. Start the license server service using "**1**" in the zebrals.sh terminal or:

```
sudo systemctl start localserver-zebra
```

Edit the DEFINES Setting

The **DEFINES** setting in the **zebrals.conf** unit file is the same as **EXTRA_SYSPROPERITES** in the **zebrals.settings** file used in the legacy (SysV init) and Windows license-server service installations. By design, this setting has no corresponding command-line option to update its value; therefore, you need to edit the setting manually in the configuration file. For instructions on manually editing this file, see Edit Settings in zebrals.conf on page 14.

This setting is used to define one or more system properties (each in -Dkey=value format) that the license server needs to pass to the Java Runtime system to support certain network functionality, such as an HTTP proxy.

For example, if you plan to have the license server communicate with the back office through an HTTP proxy, use this setting to identify the proxy parameters needed to configure the server. The following shows example proxy parameters listed as -D system properties for this setting:

```
DEFINES="-Dhttp.proxyHost=10.90.3.133 -Dhttp.proxyPort=3128 -Dhttp.proxyUser=user1a
-Dhttp.proxypassword=user1apwd35"
```

By default, this setting has no properties defined.

Enable systemd Logging

By default, the standard Syslog system for logging is used when the license server is installed as a systemd service. However, you can switch to the systemd integrated logging system, called systemd journal, once the service is installed.

To enable systemd journal, manually edit the **zebrals.conf** unit file (using the process described in Edit Settings in zebrals.conf on page 14) to update the **OPTIONS=** line to include **--systemd-logging**:

OPTIONS="--systemd-logging"

If you want to switch back to Syslog, simply remove "--systemd-logging" from **OPTIONS=**, as shown:

OPTIONS=

⚠️ **CAUTION** The license server should not use a logging level above INFO when systemd journal is enabled. A level above INFO can have a negative impact on license server performance.

Edit local-configuration.yaml Settings

Use the following procedure to update the local-configuration.yaml file generated in the /opt/zebra directory following initial registration of the service. This file contains optional settings for the license server service that are, by default, disabled.

1. From root or as a sudo user, edit the **local-configuration.yaml** file in an editor. Uncomment (or comment out) lines and edit setting values as needed. See Table 3 on page 16 below for setting descriptions.

2. Run the following command or enter "**4**" in the zebrals.sh terminal to stop and restart the service to apply the configuration changes:

```
sudo systemctl restart localserver-zebra
```

15

Table 3 Local Configuration File Settings

Setting	Description
port: port	The override for the listening port used by the license server, as described in Configuration Values Editable from the Command Line on page 12. You can edit this override value as needed. If no override is specified (that is, the setting is commented out), the service uses the default port 7070 (or the value defined in **producer-settings.xml**).
active-hostid: value/type	The hostid to use for the license server. The syntax is value/type (for example, 7200014f5df0/Ethernet). If no value is specified for this setting, the license server uses, by default, the first available Ethernet address on the machine. If using a dongle ID or a hostid other than the first available Ethernet address, specify it here.
extended-suffix: suffix	The suffix used for the Extended Host ID feature. Contact the Zebra Software Licensing team for details.
server-alias: alias	A user-defined name (sometimes called host name) for the license server. This name is added to server's capability requests to the back office, where it is then saved and used to identify the license server in the FlexNet Operations Producer and End User portals. One important use for this setting is that the alias can be included in the initial capability request sent at server registration, providing a helpful name by which users can identify the new server in the portals. If no alias is sent at registration, the server is identified by its hostid.
threshold-logging: level	The override for the lowest level of log-message granularity to record FATAL, ERROR, WARN, INFO, or POLICY, as described in Configuration Values Editable from the Command Line on page 12. You can edit this override value as needed. If no override is specified (that is, the setting is commented out), the service uses the default level INFO (or the value defined in **producer-settings.xml**).
rate-limit: value	The maximum number of capability requests per second that the license server will accept. By default, the local license server has no limit for the number of capability requests it will accept at a given time. However, if a license server experiences a high load of capability requests, a degradation in response time can result. This option is intended for use only in situations where this type of exceptional load is expected, as such a limit helps to enable consistent response times for accepted requests. A number of factors affect the rate at which the server is capable of handling requests, including the specification of the server hardware and the complexity of the requests. Therefore, the rate-limit value should be determined only through load-testing on a mirror of the production environment using a representative client load. If this setting is currently not included in the **.yaml** file, you can manually add it to the file to define the necessary rate limit.

Registering the Local License Server

As a first step, you need to register your Local License Server with Zebra Technologies. Create the server on the End User Licensing Portal with an ID that matches the **active-hostid** which you have set up in the **local-configuration.yaml** file.

To do so, first navigate to Zebra's End User Licensing Portal by entering the URL https://zebra-licensing.flexnetoperations.com/. From this page, hover over the **Devices** tab and select **Create Device**.

Figure 3 Devices Tab

From the create device page, first check the box next to **Runs license server?** which will update the fields for the device creation.

Enter an appropriate unique name for the server in the name field, select Local for a Local Licensing Server, select ID Type as Ethernet, and enter the ID as the HostID mentioned earlier from your `local-configuration.yaml` file (example ID in the image below is **value = 5CC5D44CE86B**). Do not add the "/Ethernet" to the ID field; value is just the Ethernet ID of the server without hyphens. Verify all of the information you have entered and ensure that the HostID matches the **active-hostid** value used earlier and then click **Save** to complete the device creation process.

Figure 4 New Devices Screen

Preparing to Use the License Server Manager

Integrate the License Server with Tomcat

The Zebra License Server includes the License Server Manager Administration tool to help you maintain the server and manage license distribution in your enterprise.

If you intend to use the License Server Manager, you must have an Apache Tomcat server installed. Before launching the License Server Manager for the first time, you are required to perform the following to integrate the license server with Tomcat.

1. Install the Tomcat server if one does not already exist on your system. Tomcat installers are available from the Apache Tomcat website at tomcat.apache.org/.

2. Locate the `flsm.war` file in your license server installation (in the `ui` folder).

3. Copy the `flsm.war` file to the `webapps` directory in the Tomcat installation folder.

4. If necessary, change the port on which the License Server Manager tool listens for browser requests. (By default, the License Server Manager listens on port 8080.) Use these steps:

    a. Locate the `server.xml` file in the `conf` directory of the Tomcat installation.

    b. Within the `server.xml` file, locate the appropriate connector element and modify its port attribute value to the desired port number.

5. Start the Tomcat server, and then launch the License Server Manager. See Start the License Server Manager on page 21.

Updating the LLS Web UI

In the case that you are upgrading your Local License Server to a new version, it is highly suggested that the `flsm.war` file is updated as well. After upgrading your Local License Server with the corresponding installer file, the new version of the `flsm.war` file can be found in the `ui` folder of your Local License Server installation directory.

To update the Local License Server Web UI to the new version, follow these steps:

1. Stop the Apache Tomcat service.

2. Delete the `flsm.war` file + `flsm directory` in the `webapps` folder of Tomcat.

3. Copy and paste the `flsm.war` file from the `ui` folder to the `webapps` folder of Tomcat.

4. Start the Apache Tomcat service.

Start the Local License Server

After you have completed creating and setting up your Local License Server device in Zebra's End User Licensing Portal, perform the following steps to start your local license server:

1. Open a terminal and navigate to the `server` folder in `ZebraLLS` and execute the command `sudo ./zebrals.sh` to open the following interface (see Figure 5) with different options:

Figure 5 Zebra LLS Program

```
--------------------------------------------------------
        Zebra Technologies - Local License Server (LLS) [Ver 2.0]
--------------------------------------------------------
Choose from the below options:

  Z - Register License Server as a Linux Service and Start
  U - Unregister the LLS as a Linux Service

  1 - Start the Local License Server
  2 - Check the Status of the Local License Server
  3 - Stop the Local License Server
  4 - Restart the Local License Server

  S - User Security Commands (Adding and Updating User Access)

  R - Return License to LLS License Pool (Needs Internet Connectivity)

  E - Export Served Device List

  O - Offline Sync Utilities (for LLS Without Connectivity)

  X - Exit the Interface (LLS will still be running in the background)

Input: Z


        The Local License Server can be set in 4 modes:
            1 - LLS with Internet Connectivity
            2 - LLS with Internet Connectivity + User Security Enabled
            3 - LLS without Internet Connectivity
            4 - LLS without Internet Connectivity + User Security Enabled

            B - Back to the Main Menu

Input:
```

2. Enter `Z`. This opens a sub-menu for selection of the type of service to be registered.

3. Users have four options to select when registering their service: **Options 1 and 2** offer online LLS options while **Options 3 and 4** offer offline LLS options (See Synchronization Policies Between Local License Server and Cloud on page 34 for more information about the difference between the online and offline configurations. See Enabling & Managing User Security on page 31 for information on functionalities of User Security.)

4. Confirm that the service is running by performing one of the following actions:

   • Enter `2` from the terminal main menu to check the status.

   • In the Linux, you may use `systemctl status localserver-zebra -l`.

5. You can stop or start the server using the appropriate options provided in the interface.

6. The trusted storage is created in the path `/var/opt/zebra`. This folder and its contents are machine generated. Do not tamper with them unless direct to.

7. To view the license server log, navigate to the server's logging directory (by default, `/var/opt/zebra/logs`), and review the contents of the appropriate .log file.

> **NOTE:**   Since the server is registered as a Linux Service, it will continue to run in the background even if you exit the Interface. It will continue to run unless until you explicitly stop it using the options provided or shut down your computer. When you restart your computer, the service will start running automatically and you need not have to start the server again.

Manage the Service from the Terminal

Once the license server service is installed, you can manage the service using either the zebrals.sh UI or with the systemd command `systemctl`. The following describes basic management functions for the services and their corresponding inputs in the zebrals.sh UI:

- Obtain the service status (Enter '**2**' in the zebrals.sh UI)
  ```
  sudo systemctl -l status localserver-zebra
  ```

  The -l switch disables truncation of lines in the output.

  The following example shows the status output. The current service status is at the bottom of this excerpt:

  ```
  localserver-zebra.service - Local License Server for Zebra.
  Loaded: loaded (/etc/systemd/system/localserver-zebra.service; enabled; vendor preset: disabled)
  Drop-In: /etc/systemd/system/localserver.service.d zebrals.conf Active: active (running) since
  Tue 2016-11-22 14:28:25 GMT; 18h ago
  ...
  ```

- Stop the service (Enter '**3**' in the zebrals.sh UI)
  ```
  sudo systemctl stop localserver-zebra
  ```

- Start the service (Enter '**1**' in the zebrals.sh UI)
  ```
  sudo systemctl start localserver-zebra
  ```

- Restart the service (Enter '**4**' in the zebrals.sh UI)
  ```
  sudo systemctl restart localserver-zebra
  ```

- Reread all systemd unit files
  ```
  sudo systemctl daemon-reload
  ```

Start the License Server Manager

The Zebra License Server Manager is a browser-based interface used to monitor and configure the LLS. To set up the License Server Manager using the Tomcat server:

1. Make sure you have installed Apache Tomcat and copied the `flsm.war` file from the server `ui` directory to the `webapps` directory of Tomcat.

2. Launch the Tomcat server using the instructions provided with the Tomcat product. It might be necessary to set the **JAVA_HOME** or **JRE_HOME** environment variable before starting the Tomcat server.

3. If the license server is not already running, start it. (The License Server Manager requires that the license server be running.)

4. Point a web browser to http://licenseServerHostName:8080/flsm/.

   Where: *licenseServerHostName* is the server's IP address

       8080 is the default port for Apache TomCat.

   In our example, the IP address corresponding to the Active Host ID is 10.80.200.145. So, the URL is http://10.80.200.145:8080/flsm/.

   With DNS enabled it will be something similar to http://llicenseserver.zebra.com:8080/flsm/

Figure 6 Local License Server IP

```
File  Edit  View  Search  Terminal  Help

hales@Subu17:~$ nmcli dev show
GENERAL.DEVICE:                         wlo1
GENERAL.TYPE:                           wifi
GENERAL.HWADDR:                         64:80:99:F2:64:4E
GENERAL.MTU:                            1500
GENERAL.STATE:                          100 (connected)
GENERAL.CONNECTION:                     SGuest
GENERAL.CON-PATH:                       /org/freedesktop/NetworkManager/ActiveConnection/1
IP4.ADDRESS[1]:                         10.80.200.145/22
IP4.GATEWAY:                            10.80.200.1
IP4.ROUTE[1]:                           dst = 0.0.0.0/0, nh = 10.80.200.1, mt = 600
IP4.ROUTE[2]:                           dst = 10.80.200.0/22, nh = 0.0.0.0, mt = 600
IP4.ROUTE[3]:                           dst = 169.254.0.0/16, nh = 0.0.0.0, mt = 1000
IP4.DNS[1]:                             8.8.8.8
IP4.DNS[2]:                             4.2.2.2
IP4.DOMAIN[1]:                          sguest.lan
IP6.ADDRESS[1]:                         fe80::deb9:a335:4cc3:eed8/64
IP6.GATEWAY:                            --
IP6.ROUTE[1]:                           dst = ff00::/8, nh = ::, mt = 256, table=255
IP6.ROUTE[2]:                           dst = fe80::/64, nh = ::, mt = 256
IP6.ROUTE[3]:                           dst = fe80::/64, nh = ::, mt = 600
```

> **NOTE** If licenses are activated on a device from a LLS (with or without DNS) and the IP or DNS name of the network changes, then devices are not be able to communicate with LLS for any new activation or returns of existing license. To fix this issue, update the LLS settings file with the new Ethernet Address associated to the new IP / DNS name change.

By default, the License Server Manager opens to the **Devices** view. However, if administrative security is enabled on the license server, the **Login** view opens instead so that you can enter your authorization credentials before proceeding with the next steps.

Figure 7   Devices View
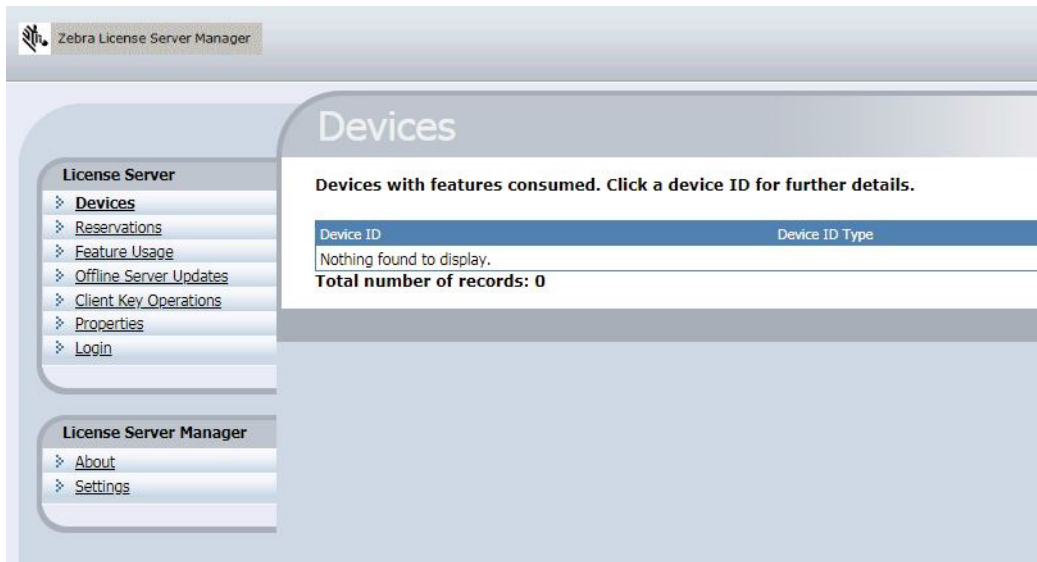
5. Click **Properties** on the left side of the interface.

6. On the Settings view, ensure that the following setting is defined correctly:

    - **Server host ID** - It should have the Host ID which was entered in the `active-hostid` field in the `local-configuration.yaml` file to register the LLS. If not, choose the correct Host ID and save the changes.

To stop the License Server Manager, close the web page and shut down the Tomcat server.

Acquire Licenses on the License Server

Licenses can be acquired to the LLS thorough the following options:

- Zebra Licensing - End User Portal
- Offline Server Updates view in the License Manager.

Zebra Licensing Portal

The licenses can be assigned to a Local License server from the End User Zebra Licensing Portal. From the End User Portal, open your Local License Server device details by navigating to the **Devices** tab and searching for the name of your Local License Server and clicking that name.

Figure 8   View Server Page
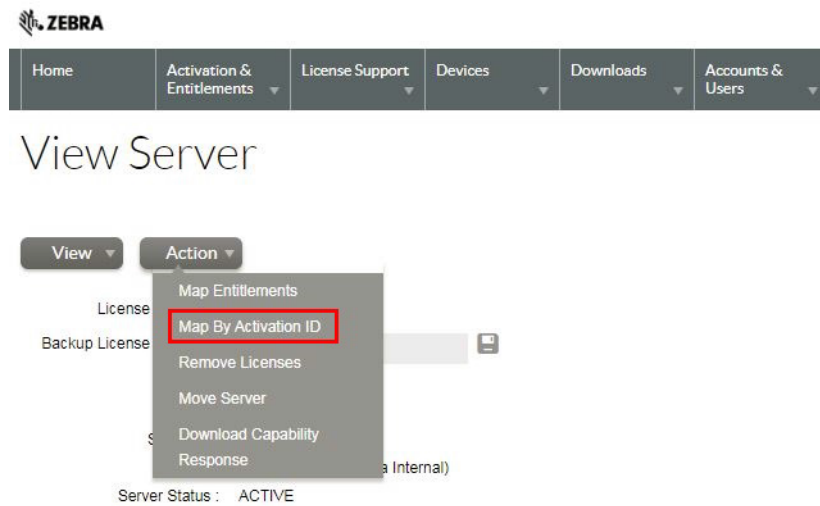
1.   In the **View Server** page, select **Actions > Map by Activation ID**. The **Map by Activation IDs** page appears.

Figure 9   Map by Activation IDs Page

Enter Activation IDs here.
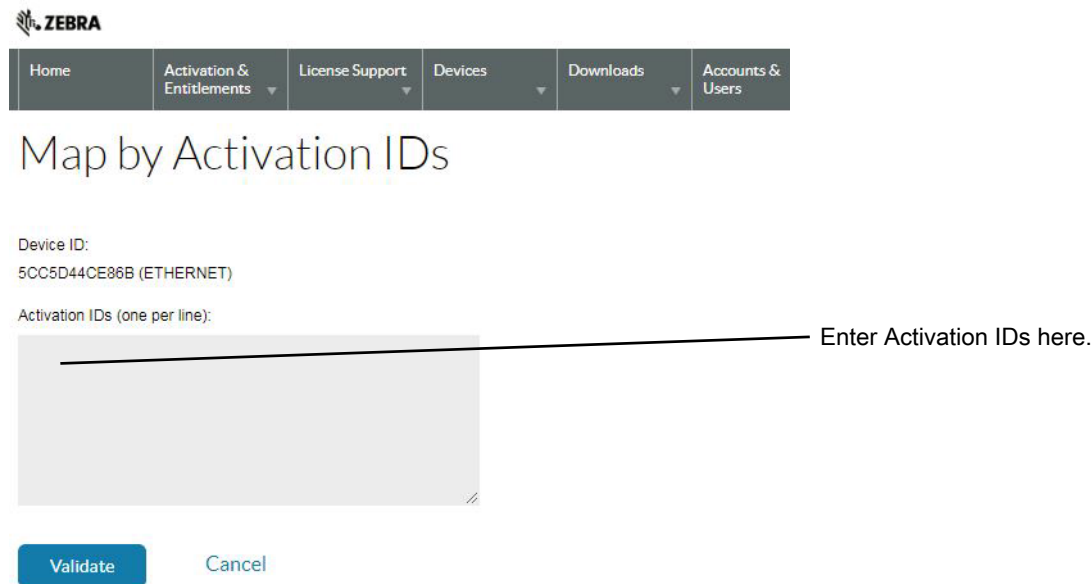
2.   In the **Activation IDs** field, enter in the desired Activation ID from the license you want to be available on the Local License Server. If you want to enter more than one Activation ID, then make sure all entries are separated by the enter key and only one per line.

3.   Click **Validate** to proceed to the next page.

Figure 10   Map by Activation IDs Page

4. After the Activation ID validation, in the **Qty to add** field enter the desired quantity of licenses you want to be available for that activation ID (must be equal to or less than the remaining quantity).

5. Click **Save** to complete the assigning process.

Once assigned to your Local License Server, you are able to view the assigned license features in the License Server Web Manager after a successful synchronization. The synchronization is scheduled to happen every two weeks for an LLS registered as an Online service. If you want to perform a manual synchronization, stop and start or restart the server from the Command Prompt UI. New license information can also be manually synced to the LLS via a binary file downloaded by the **Download Capability Response** option in the LLS device instance page in the Licensing Portal and then manually uploading the file in the Offline Server Updates View.

If you need to activate additional licenses from the same activation ID, repeat the above process and provide the additional quantity in the **Qty to add** field (see Figure 10).

Learn more about the synchronization policies in Synchronization Policies Between Local License Server and Cloud on page 34.

Offline Server Updates View

Use the **Offline Server Updates** view to update the served license pool using offline operations instead of direct communication to a back-office server. In the upper half of the view (see Figure 11 on page 25), you can do either of the following:

- Save a capability request to a binary file by clicking **Download** in the **Save capability/confirmation request for offline processing by back-office server** section. This request is used to poll for any license updates for the license server.

- Enter an activation ID (in the **Activation ID** field) and count (in the **Count** field) in the **Generate offline request using Activation ID and Count** section. When you click **Download**, the activation request is saved as a binary file. This request is used to obtain license rights for the license server through a specific activation ID that the Zebra Technologies has provided you.

Figure 11 Offline Server Updates View

Once the request is downloaded as a binary request file, navigate to Zebra's End User Licensing Portal. Hover over the **Devices** tab and then select **Offline Device Management**. In the **Offline Device Management** page (see Figure 12 on page 25), make sure **Upload Type** is selected as **Generate license** then click **Choose File** and select the recently generated binary request file. Once the file is selected, click **Upload** to generate the binary response file. To download the binary response file, click link in the alert created near the top of the page from uploading the request binary file.

Figure 12 Offline Device Management Page

After generating and downloading the binary response file from Zebra's End User Licensing Portal, browse for the file by clicking **Choose File** in the lower half of the view (in the Local Licensing Server Web Manager) and then click **Upload** (see Figure 13 on page 26). The license server then processes the response, which updates the contents of the server's trusted storage.

Figure 13   Offline Server Updates Screen

To acquire additional licenses or return licenses then the same steps need to be followed and the count needs to be changed accordingly.

For example:

- There are 20 licenses which are acquired already and there is a need to add additional 15 qty, the new count needs to be 35 (20+15).

- There are 20 licenses which are acquired already and there is a need to remove 8 licenses, the new count needs to be 12 (20-8).

NOTE: Unlike the Option from the **End Customer Portal**, the quantity is always cumulative from this interface. The Offline Server Updates method replaces the existing quantity with the new quantity. **Qty to add** or remove cannot be specified explicitly and it needs to be calculated.

Additional views available in the License Server Manger:

- **Devices** View

  The **Devices** view displays the client devices recognized by the license server as having been served licenses from the server's shared pool or from licenses reserved for the client. Each client is identified by its device ID, the ID type, and the device type (physical, virtual, or unknown). When you click the device ID for a specific client, the **Device Details** view opens, listing the features currently served to the client.

Figure 14   Devices View

- **Feature Usage** View

  The **Feature Usage** view displays details regarding all the features installed on the license server.

Figure 15   Feature Usage View

When you click a feature name, the **Feature Details** page displays containing detailed information about the feature. This includes the total count allocated to the server, the count currently being used, and the number of reserved licenses within the used count.

Figure 16   Feature Details

- **Properties** View

  The **Properties** view displays the current policy settings used by the Embedded local license server. Settings can be overridden in this view, with the exception of settings that could present a security vulnerability if editable by anyone but the producer-such as response lifetime, failover maintenance interval, or enablement of synchronization to the back office.

  Any changes made in this view take effect when you click **Save** at the bottom of the page.

Figure 17 Properties View

The properties included on this page are as follows:

- **Server host ID**: The license server's hostid value used to fulfill capability requests against a back-office server. If the server has multiple hostid values, the list contains the available hardware Ethernet addresses and dongle IDs. If virtual hosts are supported, the VM UUID is also listed. Select the value registered with your backoffice system.

- **General Properties**: The license server version, device UUID, and status.

- **Secure REST API Settings**: Settings that control administrative security on the license server.

- **Server sync settings**: Properties for synchronizing to the back office.

Figure 18   Properties View

License Generation		
Response Lifetime	1d	The lifetime of a served-license response on the client. This value can be specified with an optional unit-suffix letter-s, m, h, d, or w-indicating seconds, minutes, hours, days, or weeks. If no suffix is used, the server assumes the value is in seconds. If value is 0 (zero), the response has an unlimited lifetime.
Default Borrow Interval	1w	The borrow interval for served licenses, in seconds. This value can be specified with an optional unit-suffix letter-s, m, h, d, or w-indicating seconds, minutes, hours, days or weeks.
Default Renew Interval	15	The renew interval, as a percentage of the borrow interval, for features that do not specify an override. If set to zero, the renew interval is at client discretion.
Allow Virtual Clients	true	The property that determines whether virtual clients are allowed to obtain licenses.
Allow Virtual Server	true	The property that determines whether the license server is allowed to run on a virtual host.
Default Borrow Granularity	SECOND	The borrow granularity to use for clients that do not specify one. For clients before version 4.0, the granularity will be day, regardless of this setting.
Host ID Validation Interval	2m	The frequency with which the license server validates that its host ID has not changed. If this is set to zero, validation is disabled.
Backup URI		The URI of the backup server in a failover configuration.
Main URI		The URI of the main server in a failover configuration.
Disable Virtual Machine Check	false	The property that determines whether the server should check if it is running on a virtual host.
Client Expiry Timer Interval	2s	The interval between client expiry sessions
Settings for server to server sync between FNE servers		
Synchronization to fne enabled	false	The property that determines whether to enable server to server synchronization.
Main FNE Server URI		The URI of the main server in a failover configuration.
Synchronization to FNE interval	5m	The amount of time between initiating synchronization sessions with the main server. The value can be specified with an optional unit-suffix letter-s, m, h, d, or w-indicating seconds, minutes, hours, days, or weeks. If no suffix is used, the server assumes the value is in seconds.
Synchronization to fne pagesize	100	The maximum number of client records to include in a synchronization message to the backup server.
Synchronization to fne retry count	1	When a synchronization from the main server fails, the number of times to retry synchronization.
Synchronization to fne retry repeat interval	1m	The amount of time between synchronization attempts when, synchronization from the main server fails. The value can be specified with an optional unit-suffix letter-s, m, h, d, or w-indicating seconds, minutes, hours, days, or weeks. If no suffix is used, the server assumes the value is in seconds.
Capability polling settings		
Capability Polling Enabled	true	The property that determines whether capability-request polling is enabled. If polling is enabled, a capability request is sent to the back office periodically to update the license server's license rights.This property is used for the online deployment model of the license server.
Capability Polling Interval	1d	The amount of time between capability-request polls. The value can be specified with an optional unit-suffix letter-s, m, h, d, or w-indicating seconds, minutes, hours, days, or weeks. If no suffix is used, the server assumes the value is in seconds.
Capability Polling Retry Count	3	The number of capability-polling attempts allowed, if polling fails.
Capability Polling Retry Interval	30s	The amount of time between capability-polling attempts, if the polling fails. The value can be specified with an optional unit-suffix letter-s, m, h, d, or w-indicating seconds, minutes, hours, days, or weeks. If no suffix is used, the server assumes the value is in seconds.

- **License generation**: Policies that help manage served features.

- **Settings for server to server sync between FNE servers**: Policies for failover synchronization.

- **Capability polling settings**: Settings that control whether and how often the license server contacts the back office for license updates and how often the server should retry communications after a failed attempt.

Figure 19 Properties View

Locally deployed License Servers settings		
Server Host UUID		Host UUID that uniquely identifies this server instance when communicating with the back office.
Server Instance ID		Instance ID that uniquely identifies this server instance in the REST API.
Tenant ID		The tenant for which this server will serve licenses.
Enterprise ID		The enterprise to which this server belongs.
Site ID		The location of this server.
Trusted storage directory.	${base.dir}	The default directory for the trusted storage. The default value is the flexnetls folder in the user's home directory.
Log4J Configuration File		External override file for Log4J configuration. If no value is specified a default bundled configuration is used.
Access log pattern	access_yyyy_mm_dd.request.log	The name format for request log.
Publisher defined hostid policy	DISABLED	The property that determines whether to enable support for a custom hostid for the license server. If so, use the value strict.
Extended HostId enabled	true	The property that enables support for extended hostids for the license server.
TS Force reset	false	The property that determines whether trusted storage can be reset when unsynchronized data still exists on the license server.
Backup maintenance interval	3d	The maximum amount of time that the back-up server can serve licenses in a failover situation. This value can be specified with an optional unit-suffix letter-s, m, h, d, or w-indicating seconds, minutes, hours, days, or weeks. If no suffix is used, the server assumes the value is in seconds.
Sync Compatibility	false	Enable sync compatibility when migrating from FlexNet Embedded Server App to FlexNet Embedded License Server.
Logging Properties		
Logging Directory	C:\Windows\ServiceProfiles\NetworkService\flexnetls\zebra\logs	The directory to which the license server writes the log.
Logging Threshold	INFO	The lowest level of log-message granularity to recordfatal, error, warn, or info. For example, if fatal is set, only messages about fatal events are recorded. However, if warn is set, fatal-event, error, and warning messages are recorded.
Graylog Host		The host name of a Graylog server, if any, to which logging messages will be sent.
Graylog Threshold	WARN	The lowest level of log-message granularity to record fatal, error, warn, or info. For example, if fatal is set, only messages about fatal events are recorded. However, if warn is set, fatal-event, error, and warning messages are recorded.

- **Locally deployed license server settings**: Settings specific to your license server and environment.
- **Logging properties**: Log locations and the lowest level of granularity for log messages captured in the log.

License Server Manager Information and Settings

Below the other commands in the License Server Manager Menu is the **License Server Manager** group. The group contains the **About** and **Settings** views. The **About** view (not shown) displays build, system, and browser information. In the **Settings** view, you specify the port number and host name (network name or IP address) for the license server, as well as the number of records to show on a page. These configuration settings appear in a separate list because they can be modified even if the license server is inaccessible, unlike the license server policy settings and configuration properties.

The default port for the license server is 7070.

Figure 20 Settings View

Settings

Property	Value	Description
Page size	50	Number of records to display per page.
FlexNet license server host name	localhost	Host name where UI will look for FlexNet license server. Default is localhost.
FlexNet license server listen port	7070	Port on which UI will look for FlexNet license server. Default is 7070.
Connect using HTTPS	○ Yes ◉ No	Indicates whether to use HTTPS protocol while communicating with the server or not. Default is No.

Save

Options include:

- **Page size** - The number of records to display per page.

- **Zebra license server host name** - The hostname of the machine the license server is running. See Preparing to Use the License Server Manager on page 18 for more information about this file.
- **Zebra license server listen port** - The port number for the license server. (The default port for the server is 7070.)
- Connect using HTTPS - Indicates whether to use secured protocol while communicating with the server.

Device License Acquisition through the Local License Server

To acquire licenses that are hosted in a Local License Server on a device, the device must be directed to the correct URL.

There are two main components to be considered for the URL of device license acquisition:

- Local License Server base URL

Base URL for the Local License Server is always the IP address or DNS name associated to the Ethernet address assigned as the Host ID for the Local License Server.

For this example, the Local License Server is configured with the Host ID 648099F2644E. To enter into the Local License Server web manager, it was concluded that the Ethernet address 648099F2644E was associated with the IP address 10.80.200.145 which can be used to build the URL to access the web manager.

This means that the base URL for the Local License Server is http://10.80.200.145/ - in the case of DNS configurations, the base URL is the DNS name associated to the Ethernet address, which is http://licenseServerHostName/.

- Local License Server listen port

As identified in the previous section, the default listen port for the Local License Server is set to 7070. If any network configurations required changes to the listen port of the Local License Server, then please adjust this value accordingly.

Using both the base URL and the listen port of the Local License Server, the device license acquisition URL can be built by adding the two along with one additional requirement: /request

For this example, devices need to point to the following URL to acquire licenses from the Local License Server:

IP Address config: http://10.80.200.145:7070/request

DNS Name config: http://licenseServerHostName:7070/request

NOTE Any change to the IP address or DNS name following the configuration of the Local License Server requires an update to the device license acquisition URL as the base URL of the Local License Server would have changed.

Enabling & Managing User Security

Linux LLS version 2.0 supports User Security functionality. With User Security enabled on an LLS, queries and operations on the server are limited to a defined group of users (which can be managed through the LLS command prompt UI.) Unauthorized users are not able to perform the following actions:

- Access the LLS Web UI (includes all actions performed directly through the LLS Web UI)
- Return licenses on served devices through the command prompt UI
- Perform offline sync activities through the command prompt UI
- Manage list of authenticated users

The following are not impacted by the enabling of User Security:

- Starting/Stopping/Registering of the LLS Service on the host machine through the command prompt UI
- Activating a license on a device
- Access & Permissions to the Zebra Software Licensing Portal.

Figure 21   User Security Management Interface

```
Input: S


    Below are the commands associated to User Security (Requires Admin Password):

            1 - Create an Authorized User
            2 - Edit an Authorized User
            3 - Delete an Authorized User

            L - List All Authorized Users

            B - Back to the Main Menu

       ** Note: Enable User Security in the Registration Menu 'Z' **

Input:
```

User Security Roles & Credentials

As part of the LLS, a default administrator username and password are generated to begin using the User Security functionality:

> The default administrator credentials are in the **_UserSecurity.txt_** file in the server directory.

NOTE It is suggested that the password for the default administrator of the LLS be changed upon first activating User Security using the commands in the command prompt UI. New administrators with different usernames can also be created which will be described in the next section.

Roles for User Security

When administrative security is enabled, the default administrator account (described earlier) is assigned as ROLE_ADMIN, ROLE_RESERVATIONS, ROLE_DROPCLIENT, and ROLE_READ, granting that account full rights to administer the license server. Any user account created can be assigned one or more roles of these roles, including ROLE_ADMIN to create another administrator.

Keep in mind that any role can perform capability exchanges and synchronization operations as this functionality is exempt from security measures. These functionalities simply requires the input of any user's credentials currently listed as active from the terminal UI (using the command 'L' in the User Security section of the terminal interface).

Table 4   User Roles & Permissions Tables

Role	Permission
ROLE_READ	Permissions to perform "read" operations, such as query for features, licenses, or devices from the Web UI. When no other role is assigned to a user account, ROLE_READ is assigned by default as the only role.
ROLE_RESERVATIONS	Permissions to add and delete reservations (Note: license reservations are currently not supported so this role can be ignored).
ROLE_DROPCLIENT	Permissions to delete client records on the license server (for example, this role is required to perform the license return functionality in the LLS command prompt UI).
ROLE_ADMIN	Administrator privileges to update license server policies (LLS-only), create and manage other enterprise user accounts, and perform other administrative tasks like view the Properties panel in the LLS Web UI.
ROLE_PRODUCER	Currently not in use and can be ignored.

Credential Requirements

The requirements for username and password needed for each user account created for User Security are listed below:.

User Name

The user name (up to 64 characters) is case-sensitive and numbers can be used. Do not use special characters.

User Password

The password for a user account is case-sensitive and must meet the following criteria:

- At least 8 characters (with a max of 64 characters)
- At least one digit
- At least one upper-case character
- At least one special character (for example: ^ * $ - + ? _ & = ! % { }/ # and more)
- No whitespace.

Managing User Accounts

Once user security is enabled through the LLS command prompt UI (using either options '2' or '4' in the LLS Registration menu), the users are able to create, update and delete different user accounts based off the roles described earlier. All these commands are executed through the LLS command prompt UI.
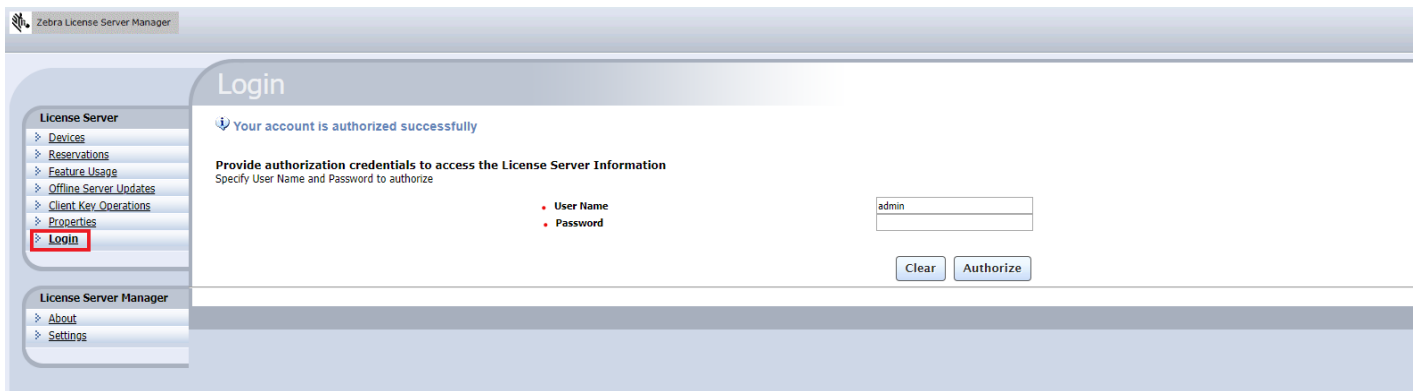
Note the following when using the User Security functionality:

- All user management functionalities require credentials from a user with ROLE_ADMIN.
- Creating or updating a user's roles requires that the exact role name be given (role is case-sensitive).
- Updating a user's role replaces the current roles with ones listed - this functionality does not aggregate roles and requires that the full list of roles be given when running the command.

- To update a user's password or roles (and not both), simply leave the input field blank for the parameter you do not wish to update and hit enter. DO NOT enter a space or any other character or that will result in a failure. For example, to update a user's password, enter the new password when prompted but hit enter when prompted to list the new roles.

- Deleting a user requires that you give the full user name in the exact same case it was created.

- Deleting a user deletes that user forever, but user name can be used again.

- Do not have more than 20 users at a time. The listing functionality of the command prompt UI will no longer work if more than 20 users are created. Delete user accounts that are no longer used to prevent this issue.

User accounts with ROLE_READ and/or ROLE_ADMIN can access the LLS Web UI through the **Login** page (see Figure 22).

Figure 22 Login Page and Authentication of LLS Web UI

Disabling User Security

To disable User Security, navigate to the main menu of the LLS command prompt UI and re-register their LLS service using the 'Z' command and selecting their desired connectivity level.

WARNING! Once User Security is turned on and a user logs in through the LLS Web UI, the Apache Tomcat web server will face issues if User Security is later turned off. To fix these issues, delete the flsm directory and flsm.war file in your Tomcat webapps directory and then recopy the flsm.war file from the UI directory of the LLS to the webapps folder.

Synchronization Policies Between Local License Server and Cloud

Synchronization between the local license server and the cloud server is enabled by default.

The default sync frequencies are as follows:

- Sync frequency between the local license server and cloud is once in two weeks.

- If the initial sync fails, two retry attempts are made after 24 hours after each attempt.

NOTE: Manual Sync - Stopping and restarting the server triggers a synchronization between the local license server and the cloud.

In terms of internet connectivity, the Local license server can be setup in the following two ways:

- Server is setup in a host machine which has network connectivity and can talk to the cloud.
- Server is setup in a standalone machine which does not have connectivity to the cloud on a continuous basis.

Server with Network Connectivity

When the server has the network connectivity, the default sync policies will come into play and the synchronization will happen automatically in the scheduled frequencies.

If there is a need for unscheduled synchronization, use the option of manual sync by stopping and restarting the server.

Server without Network Connectivity

When the server is out of network, the sync attempts will fail and the synchronization will not happen. In that case, offline synchronization tools needs to be used.

The command prompt user interface (`zebrals.sh`) needs to be installed on a machine which has the network connectivity.

Using the command prompt user interface (See Offline Synchronization Options on page 35 for details):

1. Generate the sync file from the machine where LLS is hosted.
2. Process the sync file in the new machine which has the network connectivity and get the acknowledgment file.
3. Process the acknowledgment file back in the machine where LLS is hosted.

> **NOTE:** Alternatively, the standalone machine can be connected to the network and the manual sync can be performed by restarting the server.

Disadvantages of Not Having Regular Synchronization

The served devices are showing up in the cloud which will result in issues during the repairs & services.

Offline Synchronization Options

Offline synchronization can be performed by using the options **O** (followed by options **5** through **7**) provided in the Command Prompt UI. When the LLS is out of network connectivity, you need another machine which has network connectivity to the cloud to complete the offline synchronization.

> **NOTE** Options 5 and 7 require admin credentials if User Security is enabled.

Figure 23   Offline Sync Utilities View

```
--------------------------------------------------------------
          Zebra Technologies - Local License Server (LLS) [Ver 2.0]
--------------------------------------------------------------
Choose from the below options:

  Z - Register License Server as a Linux Service and Start
  U - Unregister the LLS as a Linux Service

  1 - Start the Local License Server
  2 - Check the Status of the Local License Server
  3 - Stop the Local License Server
  4 - Restart the Local License Server

  S - User Security Commands (Adding and Updating User Access)

  R - Return License to LLS License Pool (Needs Internet Connectivity)

  E - Export Served Device List

  O - Offline Sync Utilities (for LLS Without Connectivity)

  X - Exit the Interface (LLS will still be running in the background)

Input: O


--------------------------------------------------------------
                    Offline Sync Utilities
--------------------------------------------------------------

      5 - Generate the sync file from LLS for the Cloud Server
      6 - Process the sync file from LLS in the Cloud Server
      7 - Process the sync ack file from the Could Server in the LLS

      B - Back to Main Menu
Input: █
```

Step 1: Download the Sync Files

NOTE: This step needs to be executed in the Machine where Local License Server is installed. Then the generated sync file needs to be transferred to the Machine with Internet Connectivity.

Use the option **O** followed by entering **5** and provide the IP address of the LLS machine and the path where you want the sync file to be generated. Once the download completes, a message stating the number of transaction records downloaded is displayed: OfflineSync utility started.

For example, an LLS with three devices have recently activated licenses from the LLS pool since the last successful sync. When the user enters **5** and then enters all the needed information for **Generating the sync file from LLS for the Cloud Server**, the user see the following output:

Sync completed for three device records.

If there are no new transaction records to download, the message displays as follows:

OfflineSync utility started.

No new data is available.

Step 2: Synchronize to the Cloud

NOTE: This step needs to be executed in the Machine which has Internet connectivity to the Cloud. The online machine need not have a registered the LLS service but requires all files from the **ZebraLLS** directory. Make sure the sync file generated in the machine hosting the LLS is given and give that as an input in this step.

Use the option **O** followed by option **6** and provide the path where the sync files from LLS are placed. A sync acknowledgment message is returned:

> **Successfully sent sync data and received a sync acknowledgment.**

Upon successful sync, a sync ack file is generated in the server folder with the name sync_ack.bin. This ack file needs to be transferred back to the LLS Machine.

Step 3: Update the Synchronization Time

NOTE: This step needs to be executed in the Machine where Local License Server is installed. Get the sync acknowledgment file generated and place it in the server folder.

The synchronization acknowledgment needs to be processed on the license server to update the last time of synchronization so that it knows that the data are synchronized to the back office.

Use the Option **O** followed by entering **7** and provide the IP address of the LLS Machine and the path where the sync file from Step 1 is located:

The server responds with the following message:

> **OfflineSync utility started.**
>
> **Purging file 20140613T105312.fnesync**

Return Devices Served by LLS

When a device being served by an LLS becomes inoperable and requires repair or a replacement, the user may choose to use the option **R** in the Linux LLS Terminal UI to obsolete the served device and return its licenses to the LLS License Pool.

NOTE: This action is only to be executed after receiving confirmation from Zebra's Return Merchandise Authorization team saying the non-functional device is served by an LLS and requires customer intervention to remove the licenses. The return of a served device cannot be undone so only proceed when absolutely necessary.

To return a served device, run the `zebrals.sh` and enter the option **R**.

This prompts users to enter the IP address of their Local License Server then the device serial number (or Device ID per the End User Portal) of the device they wish to return.

If User Security is enabled, users are prompted to enter admin credentials. Otherwise, no prompt are given to users.

Users are asked for confirmation and may enter `yes` to continue or `no` to cancel the operation.

The following messages are seen in sequential order if the execution is successful:

Server URL = http://<LLS IP>:7070

Device To Return = <Device ID>

Return Device in Progress… Please Wait…

Device Return Successful

Figure 24   Return Served Device View

```
--------------------------------------------------------------------
          Zebra Technologies - Local License Server (LLS) [Ver 2.0]
--------------------------------------------------------------------
Choose from the below options:

  Z - Register License Server as a Linux Service and Start
  U - Unregister the LLS as a Linux Service

  1 - Start the Local License Server
  2 - Check the Status of the Local License Server
  3 - Stop the Local License Server
  4 - Restart the Local License Server

  S - User Security Commands (Adding and Updating User Access)

  R - Return License to LLS License Pool (Needs Internet Connectivity)

  E - Export Served Device List

  O - Offline Sync Utilities (for LLS Without Connectivity)

  X - Exit the Interface (LLS will still be running in the background)

Input: R
This option allows to return/delete non functional device to the Local License Server.
This option should be used with caution as operation CANNOT be reversed.

Enter the IP Address and port number of the LLS machine [Ex: 10.12.198.01:7070]: localhost:7070
Enter the device serial number that you wish to remove: TC51_17068522513633
Enter the username for an authorized user: admin
Enter the password for the authorized user:
Do you really want to return this device? Operation cannot be reversed. Type 'yes' or 'no': yes
Return License v1.4
Authenticating user credentials - Please Wait...
Authorization of admin user was successful.
Server URL = http://localhost:7070
License to return from the Device = TC51_17068522513633
License Return in Progress... Please Wait...
License Return Successful
Press the [Enter] key to continue
```

Served Device Export Utility

In some instances, users are managing a large number of devices through the Local License Server. As part of enabling an easier method of served device management under these scenarios, Zebra provides a new export functionality.

Now available through the 'E' command in the LLS terminal UI, users can export an excel spreadsheet with a full list of their served devices info to the *deviceExport* directory found within the server folder — removing any need to manual query for devices in the Web UI.

Figure 25 Served Device Export Command

```
------------------------------------------------------------------
          Zebra Technologies - Local License Server (LLS) [Ver 2.0]
------------------------------------------------------------------
Choose from the below options:

  Z - Register License Server as a Linux Service and Start
  U - Unregister the LLS as a Linux Service

  1 - Start the Local License Server
  2 - Check the Status of the Local License Server
  3 - Stop the Local License Server
  4 - Restart the Local License Server

  S - User Security Commands (Adding and Updating User Access)

  R - Return License to LLS License Pool (Needs Internet Connectivity)

  E - Export Served Device List

  O - Offline Sync Utilities (for LLS Without Connectivity)

  X - Exit the Interface (LLS will still be running in the background)

Input: E
NOTE: This option exports teh list of served devices currently active in the LLS to an excel sheet for easier license management
          --- All exports will be sent to the deviceExport directory in the server folder ---
Enter IP address of the LLS with the port number [Ex: http://10.12.198.01:7070]: http://localhost:7070
Enter the username for an authorized user: admin
Enter the password for the authorized user:
Served Device Export - v1.0
Authenticating user credentials - Please Wait...
Authorization of admin user was successful.
Server URL = http://localhost:7070
Exporting device details... Please Wait...
Total Served Device Count = 225
Export complete. Served Device Extract available in /server/deviceExport directory
Press the [Enter] key to continue
```

The following details are included in the details of the export:

- Host ID of the Served Device
- Product Name of licenses acquired on the Served Device
- Activation ID(s) of licenses acquired on the Served Device
- Count for an Activation ID that is acquired on a Served Device
- Feature Details of the product activated on the Served Device.

NOTE: The export utility always saves the excel sheet in the deviceExport directory. The deviceExport can be found in the server folder in the Zebra Local License Server installation directory (this folder is only packaged as part of the LLS for v3.0 of the Windows LLS). Tampering with the deviceExport may lead to errors during exporting.

Uninstalling the Local License Server

To uninstall the license server service on Linux, follow the steps below:

1. Run `zebrals.sh` with root access or as the sudo user and enter **U** into the LLS UI window.

2. Alternately, run the following commands in this order:

   a. **sudo systemctl stop localserver** (to shut down the service)

   b. **sudo systemctl disable localserver-zebra** (to make the service ineligible to start on system reboot)

   c. **sudo rm /etc/systemd/system/localserver-zebra.service**

   d. **sudo rm -r /etc/systemd/system/localserver-zebra.service.d**

3. Optionally, remove these files (listed here with their default locations):

   • The trusted storage files in `/var/opt/zebra` (the .ks, .db, and .0 files)

   • The log files in `/var/opt/zebra/logs`.

Trusted storage and log locations are defined by the license server policies server.**trustedStorageDir** and **logging.directory**, respectively, the defaults for which are based on ${bases.dir}. Depending on the values set for these policies on your server, your trusted storage and log files might be in locations different from those mentioned in this step.

Recovery Options during System Failure

The server failure can be categorized into two cases, which are as follows:

   • Soft Failure - No Change in the Ethernet Mac Address

   • Hard Failure - Change in the Ethernet Mac Address.

Soft Failure

The machine on which the Local License Server is hosted has some issues which does not affect the Ethernet MAC address. The MAC address remains the same after the machine is recovers from the failure.

The Local License server is registered as a Linux Service and it will run once the machine is restarted. There is a chance that the registration is deleted or the service is stopped. So, the Host Machine recovers from failure and restarts, perform the following steps.

1. Check whether the LLS is up and running by checking in the License Server Manager UI or in the Zebra Command Prompt UI by using the **Option 2 - "Check the Status of the Local License Server"**.

2. If the server is running, no other actions required and the LLS will start reflecting the earlier status of the Licenses & Devices after the initial sync with the Zebra Cloud server. If not, continue with Step 3.

3. If the Zebra Command Prompt UI says the Service is stopped, try starting the server by using the **Option 1 - "Start the Local License Server"**.

4. If the Zebra Command Prompt UI says the Service is not installed, try registering the server again by using the **Option Z - "Register License Server as Linux Service and Start it"**.

5. Once the service starts running, the LLS will resume the earlier status before the system failure.

6. If there are any issues, please reach out the Zebra Help Desk team.

Hard Failure

The machine on which the LLS is hosted has fatal issues which affects the Ethernet MAC address, causing it to be different once the machine recovers from the failure.

In this case, a new server needs to be created using the new Ethernet MAC Address as the Host ID. The Devices which already activated licenses from the old server will continue to work without any Issues. For any new Devices, the new server needs to be used to activate License.

If there are any unused licenses which are available in the old server, reach out to the Zebra Help desk to migrate the licenses to the new server. Then the new devices can use the new server to activate those licenses.

NOTE: The served end-points however will continue to work based on existing entitlement on the end-point. Any check-in, refresh operation, etc. requires these end-points to be reconfigured to point to the new Server's updated URL.

NOTE: For details around the previously served devices or unused licenses, please reach out to the Zebra help desk to migrate the licenses to the new server. It is highly recommended that the LLS data be periodically synced with Zebra for such disaster recovery and backup purposes.

Frequently Asked Questions

Q: How do I get access to the Zebra's End User Licensing Portal?

A: Follow the Instructions in the system generated mail upon successful creation of the entitlements.

Q: What is the URL for the Zebra's End User Licensing Portal?

A: https://zebra-licensing.flexnetoperations.com/

Q: I'm not in a able to create the Local License Server in the portal because I receive "Unexpected Error" message.

A: Only users with Customer Administrator role are allowed to create a Local License Server. Contact the Zebra Help Desk to make sure you have the Customer Administrator role assigned to your user account.

Q: When I try to run the `zebrals.sh` file, it does not run the shell script.

A: Make sure the shell script is configured with execute permissions for the sudo user of the host machine.

Q: The licenses I assigned in the licensing portal are not reflecting in the local server.

A: Synchronization should happen between the cloud and the local server to reflect the changes. Refer to the sync schedule. You can manually initiate a sync by stopping the server and starting it again in the command prompt user interface.

Q: How do I upgrade to the latest version of the Local License Server?

A: There will be upgrade instructions provided for each newer version. Follow the instructions carefully to upgrade into a newer version.

Q: My server stops intermittently after I start it.

A: Make sure the host machine has enough available RAM. Insufficient RAM availability could cause the server to stop.

Q: License Server Manager URL shows 'This site can't be reached' message.

A: Make sure the Apache service is running and you have used the correct IP address in URL.

Q: What happens in the LLS when a license expires?

A: The features associated to the expired license will no longer appear in the Feature Usage page of the web manager.

Q: Why does my device say license activation failure for a particular AID but the license is still successfully acquired?

A: If two AIDs are present on a LLS for the same product, then a device pointing to the LLS will look for features associated to an AID. So when two AIDs are present, the device simply queries for any AID with the needed feature and uses that which can lead to an error message in the device since there are chances that the inputted AID may not be used to acquire the feature.