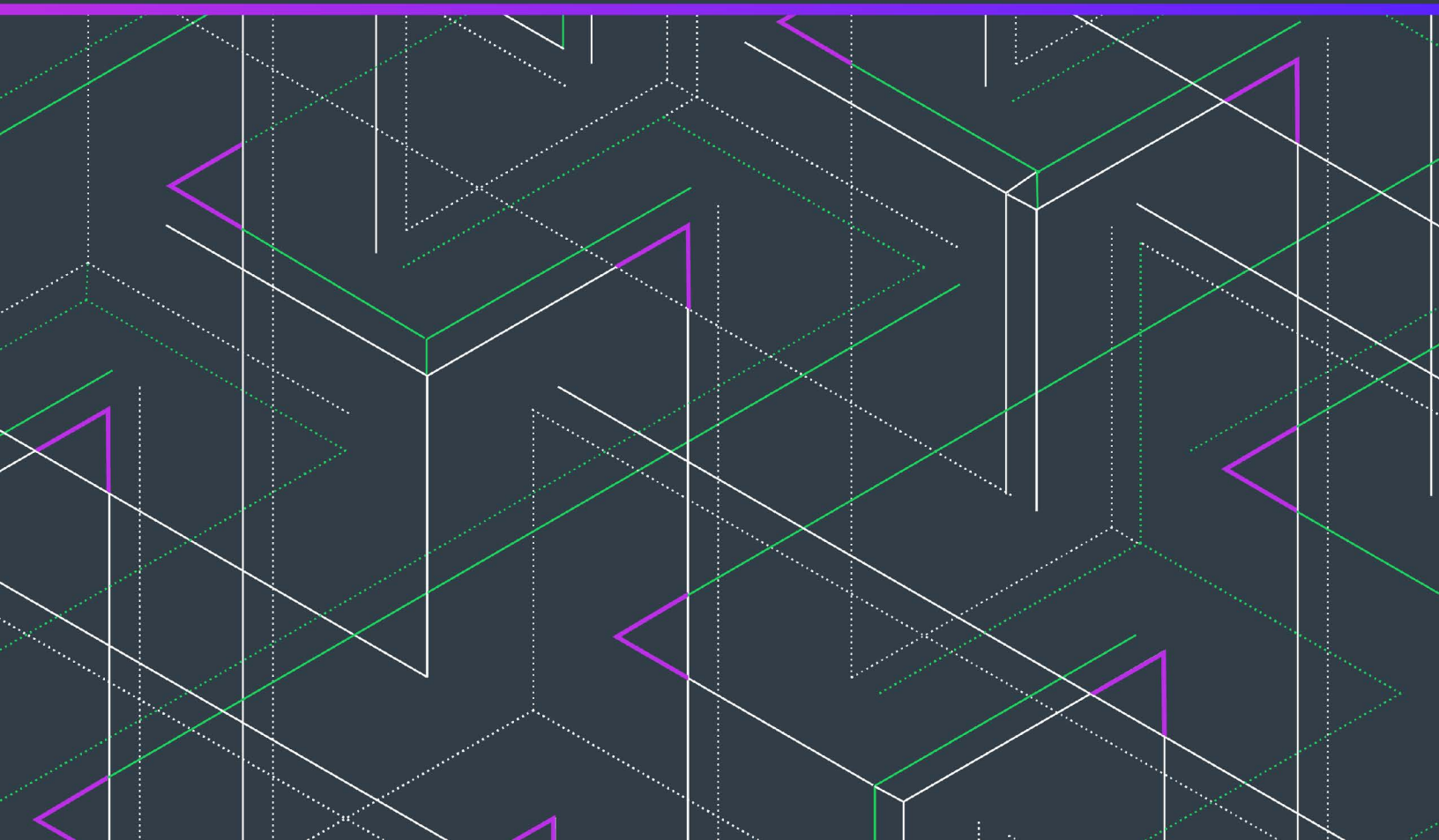
revenera

FlexNet Embedded 2026.01

License Server Migration Guide



Legal Information

Book Name:	FlexNet Embedded 2026.01 License Server Migration Guide
Part Number:	FNE-2026.01-LMG00
Product Release Date:	January 2026
Documentation Last Updated:	September 30, 2024

Copyright Notice

Copyright © 2026 Flexera Software

This publication contains proprietary and confidential information and creative works owned by Flexera Software and its licensors, if any. Any use, copying, publication, distribution, display, modification, or transmission of such publication in whole or in part in any form or by any means without the prior express written permission of Flexera Software is strictly prohibited. Except where expressly provided by Flexera Software in writing, possession of this publication shall not be construed to confer any license or rights under any Flexera Software intellectual property rights, whether by estoppel, implication, or otherwise.

All copies of the technology and related information, if allowed by Flexera Software, must display this notice of copyright and ownership in full.

FlexNet Embedded incorporates software developed by others and redistributed according to license agreements. Copyright notices and licenses for these external libraries are provided in a supplementary document that accompanies this one.

Intellectual Property

For a list of trademarks and patents that are owned by Flexera Software, see https://www.revenera.com/legal/intellectual-property.html. All other brand and product names mentioned in Flexera Software products, product documentation, and marketing materials are the trademarks and registered trademarks of their respective owners.

Restricted Rights Legend

The Software is commercial computer software. If the user or licensee of the Software is an agency, department, or other entity of the United States Government, the use, duplication, reproduction, release, modification, disclosure, or transfer of the Software, or any related documentation of any kind, including technical data and manuals, is restricted by a license agreement or by the terms of this Agreement in accordance with Federal Acquisition Regulation 12.212 for civilian purposes and Defense Federal Acquisition Regulation Supplement 227.7202 for military purposes. The Software was developed fully at private expense. All other use is prohibited.

Contents

License Server Migration Quick Start Reference 5

Basic Migration Steps 5

Migration Tasks for Less Frequently Used Features 6

1 Introduction 7

What's in this Guide 7

Product Support Resources 8

Contact Us 9

2 Feature Comparison 11

Background 11

Benefits of the New License Server 11

Feature Comparison 12

Capability Exchange Protocol 12

Feature Comparison Matrix 12

3 Migrating the License Server Configuration 15

SA and LLS Configuration Settings Matrix 15

Special Migration-Specific Setting for LLS 21

Learn More About LLS Configuration 21

4 Installation and Packaging Changes 23

Requirements for Java Runtime 23

Default Locations of LLS Components 24

Quick Reference: Default Locations of LLS Components 24

More About the Trusted-Storage Location 24

More About the Log-File Location 25

More About Anchor Locations 25

Customer Deliverables 25
Installation of License Server Service 25

5 Migrating Trusted Storage 27
Migrating Trusted Storage without License-Distribution Preservation 27
Advantages 28
Disadvantages 28
Migration Process 28
Migrating Trusted Storage with License-Distribution Preservation 30
Advantages 30
Disadvantages 30
Migration Process 31

6 Migrating Administration Tools 33
Migrating a Custom Administrative Tool 33
Moving from Device Manager on SA to License Server Manager on LLS 35

7 Other Migration Considerations 37
Migrating Hostid Customizations 37
Migrating Reservations 37

License Server Migration Quick Start Reference

This “quick start” reference provides an overview of the process used to migrate your customers from the previous C-based FlexNet Embedded server application (referred as the “SA” in this guide) to the new FlexNet Embedded local license server (referred as “LLS” in this guide).

This overview provides a general description about each step in the migration process. For more information about the step, click the associated link.

This “quick start” reference contains the following topics:

- Basic Migration Steps
- Migration Tasks for Less Frequently Used Features

Basic Migration Steps

Use these steps to migrate from the SA to the new LLS:

Table 1 ▪ Basic Migration Steps

Step	Description	Go here for details...
1	Compare the features between the two license servers. This comparison helps you plan your migration strategy, especially for those features that you currently deploy for the SA, but are not supported with the LLS.	Feature Comparison
2	Define your LLS policy settings based on your current SA settings.	Migrating the License Server Configuration
3	Prepare for LLS packaging and its installation at the customer enterprise. The LLS ‘s deliverables and its default locations for trusted storage, logs, and anchors are different from those of the SA.	Installation and Packaging Changes

Table 1 ▪ Basic Migration Steps (cont.)

Step	Description	Go here for details...
4	Determine which method to use to migrate trusted storage at your customer enterprises—migration with license-distribution preservation or without; and provide the enterprises with the appropriate migration instructions.	Migrating Trusted Storage
5	If you provided a custom administrator tool with the SA, you will need to rewrite the tool to enable it to work with the LLS's REST APIs.	Migrating Administration Tools

Migration Tasks for Less Frequently Used Features

The following provides a quick reference for migrating less frequently used features.:

Table 2 ▪ Additional Migration Tasks

Task	Description	Go here for more information...
Move from the Device Manager to the License Server Manager	Deliver the License Server Manager in the LLS package, and provide deployment and configuration instructions for the License Server Manager.	Moving from Device Manager on SA to License Server Manager on LLS in Chapter 6, *Migrating Administration Tools*
Migrate producer-defined hostids	Implement a custom shared library for the retrieval of the producer-defined hostid, and provide the necessary configuration on the LLS.	Migrating Hostid Customizations in Chapter 7, *Other Migration Considerations*
Migrate license reservations	Adjust current SA reservation specifications to accommodate the different format and behavior of reservations on the LLS.	Migrating Reservations in Chapter 7, *Other Migration Considerations*

1

Introduction

As of FlexNet Embedded 2015, a new Java-based local license server product replaced the previous C-based FlexNet Embedded server application as the standard license server offering. This guide highlights the differences between the two license server types and provides guidance in migrating from the SA to the LLS.

Use of “LLS” and “SA” in the Guide

This guide refers to the new Java-based FlexNet Embedded local license server as “LLS” and to the previous C-based FlexNet Embedded server application as “SA”. These abbreviated terms provide a clear, simple means of referencing (and distinguishing between) the two server types throughout the document.

Note ▪ *In FlexNet Embedded documentation other than this document, the new Java-based local license server is officially called the “FlexNet Embedded local license server”.*

What’s in this Guide

The FlexNet Embedded License Server Migration Guide includes the following chapters:

Table 1-1 ▪ FlexNet Embedded License Server Migration Guide Content

Topic	Content
Introduction	Provides an overview of the book and list of conventions used in the book’s contents.
Feature Comparison	Provides a brief background of the origins of the LLS and compares SA features with LLS features.
Migrating the License Server Configuration	Maps SA configuration settings to comparable LLS settings to help you configure the LLS.

Table 1-1 ▪ FlexNet Embedded License Server Migration Guide Content (cont.)

Topic	Content
Installation and Packaging Changes	Highlights differences in LLS packaging and installation requirements compared to those of the SA.
Migrating Administration Tools	Describes how to migrate your custom license-server administration tools to the LLS. It also provides a guide for migrating from the Device Manager in the SA to the License Server Manager in the LLS.
Migrating Trusted Storage	Describes the two methods for migrating SA trusted storage to the LLS: with or without license-distribution preservation.
Other Migration Considerations	Provides resources to help you migrate custom hostids and license reservations to the LLS.

Product Support Resources

The following resources are available to assist you:

- Revenera Product Documentation
- Revenera Community
- Revenera Learning Center
- Revenera Support

Revenera Product Documentation

You can find documentation for all Revenera products on the Revenera Product Documentation site:

https://docs.revenera.com

Revenera Community

On the Revenera Community site, you can quickly find answers to your questions by searching content from other customers, product experts, and thought leaders. You can also post questions on discussion forums for experts to answer. For each of Revenera's product solutions, you can access forums, blog posts, and knowledge base articles.

https://community.revenera.com

Revenera Learning Center

The Revenera Learning Center offers free, self-guided, online videos to help you quickly get the most out of your Revenera products. You can find a complete list of these training videos in the Learning Center.

https://learning.revenera.com

Revenera Support

For customers who have purchased a maintenance contract for their product(s), you can submit a support case or check the status of an existing case by first logging into the Revenera Community, clicking **Support** on the navigation menu to open the **Support Hub** page, and then clicking the **Open New Case** or **Case Portal** button.

Contact Us

Revenera is headquartered in Itasca, Illinois, and has offices worldwide. To contact us or to learn more about our products, visit our website at:

http://www.revenera.com

You can also follow us on social media:

- Twitter
- Facebook
- LinkedIn
- YouTube
- Instagram

2

Feature Comparison

This chapter provides a brief background of the origins of the LLS, describes its benefits, and compares its individual features with features in the SA.

Background

In 2013, Revenera introduced the Cloud Licensing Service (CLS) component as part of the FlexNet Usage Management platform. The CLS component is built as an Enterprise Class Java application service engineered for scalability, reliability, security, and performance. The new LLS is developed from the same code base as the CLS to leverage these same benefits and to offer a license server that runs basically the same whether at the customer site or in the Revenera Cloud.

Benefits of the New License Server

Moving from the previous C-based SA to the new Java-based LLS provides offers several benefits.

No Build Step Needed

The Java-based LLS takes advantage of built-in behavior—such as built-in REST functionality, JSON marshalling, and better database support—that the producer would have to build with the C-based SA. Plus, the removal of the producer's build step speeds up the delivery security and bug fixes for the server.

JSON Flexibility

Data exchange with LLS uses JSON, a format that is more flexible and simpler to use than XML. Additionally, REST functionality frequently inter-operates with "single-page applications" written in Javascript. JSON, a native format for Javascript, easily works with these applications.

Licensing Code Identical to CLS

The licensing code in the LLS is identical to that used in the CLS, ensuring the same licensing behavior in the two license servers.

Support for Metered Licensing

The LLS supports metered licensing—that is the Usage Capture and Management. The SA architecture did not lend itself to such support.

Feature Comparison

The following sections compare features in the new LLS with those in the SA, providing a high-level look at the differences and similarities between the two license servers. In addition to serving as a resource for learning about the new license server, the comparison provides as a basis for planning a migration strategy for features that you currently deploy in the SA but that are not supported in the LLS.

Capability Exchange Protocol

The capability-exchange protocol used by the new license server is the same as the protocol used by the SA. That is, any FlexNet Embedded client that successfully performed capability exchanges with the SA can continue to do so with the LLS; no modifications to the LLS or to the FlexNet Embedded client are needed.

Feature Comparison Matrix

The following chart compares feature support between the SA and LLS:

Table 2-1 ▪ Feature Comparison Matrix

Category	Feature	SA	LLS
Technology	Programming language	C++/C	Java EE C/C++
Server policy (producer)	Producer configuration	Compile-time configuration requires platform-specific toolchain setup: 1. Modify source code (`server_properties.c`) with configuration changes. 2. Compile and build the code.	Run-time configuration does not require any additional setup and is platform-independent: 1. Generate `producer-settings.xml`. 2. Include the file in the license server package.
Redistribution	Redistributing/ updating server components to enterprises	Rebuild and redistribute platform-specific binary.	Platform-independent package does not require compilation.

Table 2-1 ▪ Feature Comparison Matrix (cont.)

Category	Feature	SA	LLS
Server security	Coverage for trusted-storage backup/ restore exploits (anchoring)	Basic	Basic plus secure anchoring (see next entry)
	Secure anchoring	Not supported	Supported
	Incoming HTTPS (communication from FlexNet Embedded client to license server using SSL)	Supported	Supported
	Outgoing HTTPS (communication from license server to FlexNet Operations using SSL)	Supported	Supported
License models	Concurrent licensing	Supported	Supported
	Metered licensing	Not supported	Supported
Capability polling	Online and offline	Supported	Supported
Synchronization	Synchronize to FlexNet Operations (online and offline)	Supported (concurrent-license distribution only)	Supported (concurrent-license distribution and metered usage)
	Synchronize from FlexNet Operations (online)	Supported	Supported
High availability	Failover configuration	Supported	Supported (concurrent licenses only)
Reservations	Device-based and user-based reservations	Supported In the SA, user-based reservations are identified by tokens sent in the vendor dictionary in capability requests.	Supported (concurrent licenses only; reservations managed in “groups”) In the LLS, user-based reservations are identified by secondary hostids sent in capability requests.

Table 2-1 ▪ Feature Comparison Matrix (cont.)

Category	Feature	SA	LLS
Feature partitions	Allocating licenses to specific client devices or users	Not supported	Supported
Hostid support	Ethernet	Supported	Supported
	FlexID9	Supported	Supported
	FlexID10	Supported	Supported
	Producer-defined (custom)	Supported	Supported
	Extended hostid	Supported	Supported
Client communication modes	“Push” architecture (license server initiates license updates to clients)	Supported	Not supported
	“Pull” architecture (clients initiate license requests to license server)	Supported	Supported
Special licensing types	Info messages	Supported	Not supported
Administrative interfaces	Programmable APIs to access license server’s configuration and status	REST/XML endpoints	REST/JSON endpoints
	License administrator UI tool	Supported	Supported
	License administrator command-line tool	Not supported	Supported

3

Migrating the License Server Configuration

For SA configuration, you defined license server policies in the `server_properties.c` file, which was then built with the server application. The license administrator at the customer enterprise could update additional, editable properties directly in the `properties.xml` file, provided with the server, or through the server's `setproperties` REST API (or an administrative interface, such as the FlexNet License Server Manager, that called this API).

To configure the LLS, you generate a `producer-settings.xml` file containing your license server policies, including the enterprise-editable ones, and distribute this file with the license server. The license administrator at the customer enterprise can use the `configuration` REST API to override any of the editable settings. As with the SA, this API can be called through an LLS administrative interface. (For the LLS, this interface might be the FlexNet License Administrator command-line tool, the FlexNet License Server Manager, or a custom tool you develop.)

The LLS license administrator can also define and edit properties specific to the server's local environment. These properties are found in the `flexnetls.settings` file on Windows and in the `/etc/default/flexnetls-`*`producer_name`* file on Linux.

The following sections provide guidance in migrating your SA configuration settings to comparable LLS settings:

- SA and LLS Configuration Settings Matrix
- Special Migration-Specific Setting for LLS
- Learn More About LLS Configuration

SA and LLS Configuration Settings Matrix

The following table attempts to provide one-to-one correspondence between settings in the SA configuration files and those in the LLS configuration files, when such correspondence is applicable.

Once you determine how you want to configure the LLS to accommodate your previous SA configuration, generate the `producer-settings.xml` file to define the LLS configuration (described in the later section Learn More About LLS Configuration).

***Note** ▪ Some SA configuration settings are not supported by the LLS. If you require these settings, you must devise a solution for supporting them as part of your license server migration strategy.*

Unless noted otherwise, all settings listed in the table are defined in either `server_properties.c` for the SA or in `producer-settings.xml` for the LLS.

Table 3-1 ▪ Migration Matrix for License Server Configuration Settings

General Policy or Definition	SA Setting	LLS Setting	Comments
Producer name	`#define PUBLISHER_NAME`	N/A	**For LLS:** Retrieved from identity data during generation of `producer-settings.xml`
Licensing Policies			
Response lifetime	`FNE_SERVER_RESPONSE_LIFETIME`	`licensing.responseLifetime`	
Virtual host detection	`FNE_SERVER_VIRTUAL_HOST_DETECTION_ENABLED`	`licensing.disableVirtualMachineCheck`	**For LLS:** By default, hidden in `producer-settings.xml` unless overridden
Allow clients running on virtual machines	`FNE_SERVER_VIRTUAL_CLIENTS_ENABLED`	`licensing.allowVirtualClients`	
Allow server to run on virtual machine	`FNE_SERVER_VIRTUAL_HOST_ENABLED`	`licensing.allowVirtualServer`	
Expiration granularity	`FNE_SERVER_EXPIRATION_GRANULARITY_INTERVAL`	`licensing.defaultBorrowGranularity`	
Interval for validating enabled hostid	`FNE_SERVER_HOSTID_VALIDATION_INTERVAL`	`licensing.hostIdValidationInterval`	
Back-up server URI	`failover-backup-uri` (in `properties.xml`)	`licensing.backup.uri`	
Main server URI	`failover-main-uri` (in `properties.xml`)	`licensing.main.uri`	
License Server Settings			

Table 3-1 ▪ Migration Matrix for License Server Configuration Settings (cont.)

General Policy or Definition	SA Setting	LLS Setting	Comments
Enable producer-defined hostid	FNE_SERVER_PUBLISHER_HOSTID_STRING	server.publisherDefinedHostId.policy	**For LLS:** Producer-defined hostid retrieved from shared library that you create and distribute with server
Enable extended hostid support	FNE_SERVER_EXTENDED_HOSTID_ENABLED	server.extendedHostId.enabled	
Allow trusted-storage reset when unsynchronized data exists	FNE_SERVER_FORCE_TRUSTED_STORAGE_RESET_ALLOWED	server.forceTSResetAllowed	
Directory for cache file	FNE_SERVER_PROPERTY_CACHE_DIR	N/A	
Borrow interval for served licenses	FNE_SERVER_BORROW_INTERVAL	N/A	**For LLS:** Property defined in license model in FlexNet Operations *
Renew interval for served licenses	FNE_SERVER_RENEW_INTERVAL	N/A	**For LLS:** Property defined in license model in FlexNet Operations *
Back-up server's maintenance interval	FNE_SERVER_MAINTENANCE_INTERVAL	server.backupMaintenance.interval	
Enable support for information messages	FNE_SERVER_INFOMSG_USAGE_ENABLED	N/A	**For LLS:** Not supported
Trusted-storage location	FNE_SERVER_PERSISTENT_DATA_LOCATION	server.trustedStorageDir	**For LLS:** (Linux only) Editable through the DATA-DIR local setting in /etc/default/flexnetls-*producer_name*
Server host type	FNE_SERVER_HOST_TYPE	N/A	**For LLS:** Not supported

Table 3-1 ▪ Migration Matrix for License Server Configuration Settings (cont.)

General Policy or Definition	SA Setting	LLS Setting	Comments
Enable PUSH architecture	FNE_SERVER_PUSH_ ARCHITECTURE_ENABLED	N/A	**For LLS:** Not supported
Back Office URL			
Back-office URL	fno-servername, fno-port, and fno-request-uri (in properties.xml)	lfs.url	
Polling of Back Office for License Updates			
Enable polling of the back office for license updates	fno-poll-enabled (in properties.xml)	lfs.capability.enabled	
Interval between polling sessions with back office	fno-poll-interval (in properties.xml)	lfs.capability.repeats	
Number of retries allowed when polling fails	fno-poll-retry-count (in properties.xml)	lfs.capability.retryCount	
Interval between retries	fno-poll-retry-interval (in properties.xml)	lfs.capability.retry Repeats	
Synchronization to the Back Office			
Enable online synchronization to back office	FNE_SERVER_SYNC_TO_FNO_ ENABLED	lfs.syncTo.enabled	
Interval for synchronization to back office	FNE_SERVER_SYNC_TO_FNO_ INTERVAL	lfs.syncTo.repeats	
Number of retries when synchronization fails	FNE_SERVER_SYNC_TO_FNO_ RETRY_COUNT	lfs.syncTo.retryCount	
Interval between retries	FNE_SERVER_SYNC_TO_FNO_ RETRY_INTERVAL	lfs.syncTo.retryRepeats	
Include historical data	FNE_SERVER_SYNC_TO_FNO_ INCLUDES_HISTORICAL_DATA	lfs.syncTo.includeAll	

Table 3-1 ▪ Migration Matrix for License Server Configuration Settings (cont.)

General Policy or Definition	SA Setting	LLS Setting	Comments
Page size	FNE_SERVER_SYNC_TO_FNO_PAGE_SIZE	lfs.syncTo.pagesize	
Enable offline synchronization to back office	FNE_SERVER_SYNC_TO_FNO_OFFLINE_ENABLED	N/A	**For LLS:** Enabled by default
Synchronization from the Back Office			
Enable synchronization recovery from the back office	FNE_SERVER_SYNC_FROM_FNO_ENABLED	lfs.syncFrom.enabled	
Number of retries when synchronization fails	sync-from-fno-retry-count (in properties.xml)	lfs.syncTo.retryCount	**For LLS:** Uses value for corresponding syncTo setting
Interval between retries	sync-from-fno-retry-interval (in properties.xml)	lfs.syncTo.retryRepeats	**For LLS:** Uses value for corresponding syncTo setting
Page size	sync-from-fno-page-size (in properties.xml)	lfs.syncTo.pagesize	**For LLS:** Uses value for corresponding syncTo setting
Communications			
Use HTTPS for communication with back office	FNE_SERVER_FNO_CAPABILITY_HTTPS_ENABLED	N/A	**For LLS:** Enabled through flexnet.certs or through additional configuration
Use HTTPS for synchronization with back office	FNE_SERVER_FNO_SYNC_HTTPS_ENABLED	N/A	**For LLS:** Enabled through flexnet.certs or through additional configuration
Ignore HTTPS errors	FNE_SERVER_IGNORE_HTTPS_ERRORS	N/A	

Table 3-1 ▪ Migration Matrix for License Server Configuration Settings (cont.)

General Policy or Definition	SA Setting	LLS Setting	Comments
Root certificate path	FNE_SERVER_HTTPS_ROOT_CA_PATH	On Linux, HTTPS_SERVER_FILE in local settings file (flexnetls-*producer_name*) On Windows, HTTPS_SERVER_CONFIGURATION in local settings file (flexnetls.settings)	**For LLS:** See FlexNet Embedded License Server Producer Guide (specifically "Incoming HTTPS" in the "Advanced License Server Features" chapter)
License Server Failover			
Enable license-server failover	sync-to-fne-enabled (in properties.xml)	fne.syncTo.enabled	
Back-up server URI	sync-to-fne-backup-uri (in properties.xml on main server)	fne.syncTo.mainUri (in producer-settings.xml on back-up server)	Opposite methods for specifying failover server configuration: **SA:** Identify back-up server in properties.xml on main server **LLS:** Identify main server producer-settings.xml on back-up server
Interval for synchronization to main license server	sync-to-fne-interval (in properties.xml)	fne.syncTo.repeats	
Number of retries should synchronization fail	sync-to-fne-retry-count (in properties.xml)	fne.syncTo.retryCount	
Interval between retries	sync-to-fne-retry-interval (in properties.xml)	fne.syncTo.retryRepeats	
Page size	sync-to-fne-page-size (in properties.xml)	fne.syncTo.pagesize	
Logging			
Logging threshold	log-level (in properties.xml)	logging.threshold	

Table 3-1 ▪ Migration Matrix for License Server Configuration Settings (cont.)

General Policy or Definition	SA Setting	LLS Setting	Comments
Logging directory	`log-directory` (in `properties.xml`)	`logging.directory`	

* Revenera recommends that you switch to the license-model approach for defining this setting. However, if you would like to define the setting using a behavior similar to that used in the SA, consult Revenera.

Special Migration-Specific Setting for LLS

You can choose to preserve license-distribution data during migration from the SA to the LLS. (See Migrating Trusted Storage with License-Distribution Preservation in the "Migrating Trusted Storage" chapter for details.) If you make this choice, change the default `false` value for the `server.synCompatibility` setting to `true` in the LLS's `producer-settings.xml` file:

```
server.syncCompatibility=true
```

This setting reconciles a time-unit discrepancy (resulting from the migration) in the timestamps used for synchronization operations. The setting is required to help ensure the following:

- Synchronization from the back office works properly *during* migration to the LLS.
- Synchronization to and from the back office continues to work properly *after* migration.

For more information about updating the default settings in `producer-settings.xml`, see the "License Server Configuration Utility" section in the *Producer Tools* chapter in the FlexNet Embedded License Server Producer Guide.

Learn More About LLS Configuration

Not all SA settings are supported in the LLS, as pointed in the previous section. Likewise, the LLS has configuration settings not available in the SA.

Refer to the following parts of the FlexNet Embedded License Server Producer Guide for more information about LLS configuration:

- For a description of each setting in the `producer-settings.xml` file for the LLS, see the appendix *Reference: Policy Settings for the License Server*. (The descriptions also indicate which settings are editable by the customer enterprise.)
- For instructions on generating the `producer-settings.xml` file, see the "Generating the Producer Settings File" section in the *Getting Started* chapter and the "License Server Configuration Utility" section in the *Producer Tools* chapter.
- For information about using the `configuration` REST API in your custom administration tool to enable the enterprise license administrator to override any editable settings in the `producer-settings.xml` file, see the *License Server REST APIs* chapter.

Additionally, for information about the FlexNet License Administrator command-line tool or the FlexNet License Server Manager, which also allow the enterprise license administrator to override editable settings in `producer-settings.xml`, refer to the FlexNet Embedded License Server Administration Guide.

4

Installation and Packaging Changes

The installation requirements for the LLS differ from the those for the SA. The following sections highlight differences that you might want to consider as you prepare for the migration process:

- Requirements for Java Runtime
- Default Locations of LLS Components
- Customer Deliverables
- Installation of License Server Service

Requirements for Java Runtime

The LLS requires that JRE be installed on the machine on which the license server runs. Specifically, the server requires the following:

- Oracle Java SE 17 or OpenJDK 17.
- (Only for releases prior to 2017 R2) A 64-bit JRE for a 64-bit license server or a 32-bit JRE for a 32-bit license server. Not adhering to this requirement can cause the license server to fail to start.
- Windows only: 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. See the FlexNet Embedded License Server Release Notes for details.*

For a list of all system requirements, see the FlexNet Embedded License Server Release Notes.

Default Locations of LLS Components

The LLS uses default locations for trusted storage, logs, and anchors that are different from the those used by the SA. Refer to the following sections for more information about the default locations for LLS components:

- Quick Reference: Default Locations of LLS Components
- More About the Trusted-Storage Location
- More About the Log-File Location
- More About Anchor Locations

Quick Reference: Default Locations of LLS Components

The following table lists the default locations used by the LLS for trusted storage, logs, and anchors.

Table 4-1 ▪ Default Locations for LLS Trusted Storage and Logs

Server Component	Server Mode	Windows Location	Linux Location
Trusted storage	Service	`C:\Windows\ServiceProfiles\NetworkService\flexnetls\`*`producer_name`*	`/var/opt/flexnetls/`*`producer_name`*
	Console	`C:\Users\`*`user_name`*`\flexnetls\`*`producer_name`*	`/var/opt/flexnetls/`*`producer_name`*
Logs	Service	`C:\Windows\ServiceProfiles\NetworkService\flexnetls\`*`producer_name`*`\logs`	`/var/opt/flexnetls/`*`producer_name`*`/logs`
	Console	`C:\Users\`*`user_name`*`\flexnetls\`*`producer_name`*`\logs`	`/var/opt/flexnetls/`*`producer_name`*`/logs`

More About the Trusted-Storage Location

The default location of the LLS trusted storage files is the `flexnetls\`*`producer_name`* directory, located in the home directory (`${base.dir}`) of the license-server service or the user (see Quick Reference: Default Locations of LLS Components).

As producer, you can change the default location for trusted storage using the `server.trustedStorageDir` property in the `producer-settings.xml` file. Additionally, on Linux, customer enterprises can override the location set in the `producer-settings.xml` file by editing the `DATA-DIR` property in server's local settings file (`/etc/default/flexnetls-`*`producer_name`*). However, no comparable property exists (or is necessary) in `flexnetls.settings` to allow customer enterprises on Windows to override the trusted-storage location.

For more information, see the "Installing and Running the License Server" chapter in the FlexNet Embedded License Server Producer Guide.

More About the Log-File Location

The default location of the LLS log files is the `flexnetls\`*`producer_name`*`\logs` directory, located in the home directory (`${base.dir}`) of the license-server service or the user (when running in console mode). As producer, you can change the default log location using the `logging.directory` property in the `producer-settings.xml` file. In turn, customer enterprises can override this default setting to accommodate their local environments.

More About Anchor Locations

Like the SA, the LLS uses default anchoring techniques to provide a standard level of trusted-storage security. The files needed to enable this anchor-based security are stored in certain locations and cannot be changed.

In addition to standard anchoring security, the LLS offers advanced anchoring functionality called *secure anchoring*. This functionality uses additional techniques to store anchor information *throughout the system* on which your license server runs to provide a level of anchor security greater than the default anchoring techniques. For more information, see “Enabling Secure Anchoring” in the “Getting Started” chapter in the FlexNet Embedded License Server Producer Guide.

Customer Deliverables

The LLS components packaged and delivered to customers are different from those packaged and delivered with the SA. For an overview of LLS deliverables to a customer enterprise, see “Delivering the License Server” in the “Getting Started” chapter of the FlexNet Embedded License Server Producer Guide.

Installation of License Server Service

The process for installing the LLS service is different from the process used to install the SA service. Refer to the “Installing and Running the License Server” chapter in the FlexNet Embedded License Server Producer Guide for details.

5

Migrating Trusted Storage

Trusted storage on both the SA and the LLS includes the following basic information:

- License rights activated on the license sever
- License distribution state
- Enterprise-specific license server configuration

Depending on the license model that you deployed on the SA (specifically the borrow and renew interval for the desired features), you must consider whether or not preservation of the current license-distribution state is important for your customers when migrating to the LLS. Either migration method—with license-distribution preservation or without—has its benefits and disadvantages. This information, along with the migration procedure, is described in the following sections:

- Migrating Trusted Storage without License-Distribution Preservation
- Migrating Trusted Storage with License-Distribution Preservation

When you decide which migration method works best for a given enterprise, provide the appropriate migration instructions described here to the enterprise.

Migrating Trusted Storage without License-Distribution Preservation

Migration of trusted storage without preserving the current license-distribution state is simpler than preserving license-distribution during migration. Basically, when this migration method is used, the new LLS trusted storage is created and it licenses activated. However, no license-distribution or usage data from the SA trusted storage is restored to the LLS.

To send capability requests to the LLS, FlexNet Embedded clients can use the same URI (`/fne/bin/capability`) they used previously for the SA.

The following sections provide more detail about this type of migration:

- Advantages

- Disadvantages
- Migration Process

Advantages

The following are the major advantages of the migrating trusted storage without data preservation:

- The process is simpler as no synchronization from the back office (FlexNet Operations) to the LLS occurs.
- No online connection to the back office is required—license activation on the license server can be performed offline, and no synchronization from the back office is run.

Disadvantages

The major disadvantage of migrating trusted storage without data preservation is the possibility of temporary license overage. That is, more served clients might exist than is allowed for given licenses due to clients holding licenses during the migration period. Once migration is complete and records no longer exist in trusted storage for these served clients, the LLS can serve the licenses to new clients, possibly serving beyond the available license count. However, once the license-borrowing period expires for the existing served clients, the overage is corrected.

Because of the possibility of overages, migration without data preservation is suitable when the borrow period is short enough so that the cost of license leakage due to temporary overages is justified. However, this type of migration is unsuitable when the borrow period is long or licenses are permanent.

Migration Process

The process of migrating trusted storage from the SA to the LLS without preserving the current license-distribution state consists of these phases:

- Migration Procedure
- Post-Migration Validations

Migration Procedure

Have the license administrator at the customer enterprise perform these steps to migrate trusted storage.

Task ***To migrate to the LLS without license-distribution preservation***

1. Download and install the LLS as a service on the same machine where the SA is installed. For more information, see the “Installing and Running the License Server” chapter in the FlexNet Embedded License Server Producer Guide.
2. Shut down the SA.
3. Activate the licenses on the LLS using the online- or offline-activation process; otherwise, if you have enabled capability-polling on the LLS, the licenses should be automatically activated.

- For more information about setting up capability-polling, see the “Policies for Back Office Polling...” in the appendix “Reference: Policy Settings for the License Server” in the FlexNet Embedded License Server Producer Guide.
- For information about passing activation IDs and counts to the back office (FlexNet Operations) in online or offline mode, see “Activating Licenses on the Server” or “APIs to Perform Offline Binary Operations” in the “License Server Endpoints” chapter in the FlexNet Embedded License Server Producer Guide.
- Use the ready-to-use FlexNet License Administrator command-line tool to activate the licenses. See the FlexNet Embedded License Server Administration Guide for details.

The LLS should proceed with normal licensing operations—accepting capability requests from client devices, synchronizing to the back office (if enabled), polling the back office for license updates (if enabled), and other operations.

Note ▪ *The LLS accepts capability requests at the same URI (“/fne/bin/capability”) at which the SA accepted them; therefore, no change in the FlexNet Embedded client configuration is required.*

4. Use the validation procedure described in Post-Migration Validations to verify that the license server has been successfully migrated.
5. Once successful migration is verified, uninstall the SA.

Post-Migration Validations

Have the license administrator at the customer enterprise perform these tasks to verify that the trusted-storage migration completed successfully.

Task ***To validate that the trusted-storage migration to the LLS was successful***

- Using the FlexNet License Administrator tool, run the following:
  - `flexnetlsadmin -status` to perform a health check on the LLS.
  - `flexnetlsadmin -licenses` to verify that licenses have been activated appropriately on the LLS.
- Confirm that the LLS allows existing clients to renew their borrowed licenses.
- Confirm that new clients can obtain licenses from the LLS.
- Examine the LLS log to verify that online synchronization to the back office is working (if the producer has enabled this functionality).
- Use the FlexNet Operations End User Portal to verify that the LLS’s operations with the back office are functioning properly. That is, make sure that you can view the license server’s properties, its served clients, and other information.

Migrating Trusted Storage with License-Distribution Preservation

To preserve the license-distribution state when migrating to the LLS, the SA is synchronized to the back office prior (FlexNet Operations) to the migration. During migration the new LLS trusted storage is created and its licenses activated. Synchronization from the back office to the LLS is then automatically performed to restore the server's license distribution to its state just previous to the migration. Once recovery is complete, the LLS can begin accepting capability requests from client devices. (Client devices can use the same URI—`/fne/bin/capability`—used previously to send requests to the SA.)

Important ▪ *This type of migration works only with* FlexNet Operations *Cloud. It does not work with* FlexNet Operations *On-Premises.*

The following sections provide more detail about this type of migration:

- Advantages
- Disadvantages
- Migration Process

Advantages

The major advantage in migrating trusted storage with data preservation is that the license-distribution state is restored to what it was before the migration. Thus, the possibility for license overages is reduced since records of served clients have been retained.

This type of migration is preferable when the borrow period is long or licenses are permanent.

Disadvantages

The disadvantages of migrating trusted storage with data preservation include the following. (See Pre-Migration Tasks for details.)

- This type of migration requires an online connection between the back office and the LLS to allow synchronization from the back office.
- The migration works only with FlexNet Operations Cloud as the back office. It does not work with FlexNet Operations On-Premises.
- Modifications to the SA configuration are required to enable synchronization—either online or offline—to the back office if one or the other is not already enabled.
- Modifications to the LLS configuration are required both to enable a new timestamp-compatibility policy for synchronization and to enable synchronization from the back office if it is not already enabled.

Migration Process

The process of migrating trusted storage from the SA to the LLS with the preservation of the current license-distribution state consists of these phases:

- Pre-Migration Tasks
- Migration Procedure
- Post-Migration Validations

Pre-Migration Tasks

The license administrator at the customer enterprise and the producer must perform the following tasks prior to the migration of trusted storage to enable the LLS to recover the license-distribution data from the SA.

- The producer ensures that the SA configuration has synchronization to the back office enabled. One of these properties in `server_properties.c` must be set to `true`:
  - `FNE_SERVER_SYNC_TO_FNO_ENABLED` (for online synchronization)
  - `FNE_SERVER_SYNC_TO_FNO_OFFLINE_ENABLED` (for offline synchronization).
- The producer ensures that the `producer-settings.xml` file distributed with the LLS includes the following:
  - The `lfs.syncFrom.enabled` must be set to `true` to enable synchronization from the back office to recover license-distribution data.
  - The new `server.syncCompatibility` setting must be set to `true` to enable proper conversion of time units used for timestamps in synchronization from the back office *during* migration (and in all synchronization operations to and from the back office that occur thereafter).

  For more information, see Migrating the License Server Configuration.
- The license administrator makes sure that an online connection between the LLS and the back office is set up—at least for the duration of synchronization from the back office to the LLS.
- The license administrator ensures that the LLS will use the same hostid currently used by the SA in its operations with the back office.
- Just prior to the migration, the license administrator forces a synchronization from the SA to the back office using one of the following:
  - The `/fne/xml/start-to-sync-to-fno` XML endpoint (for online synchronization)
  - The offline synchronization procedure

Migration Procedure

Once the synchronization from the SA to the back office is complete, have the license administrator perform the same steps described in Migration Procedure for the “Migrating Trusted Storage without License-Distribution Preservation”.

Post-Migration Validations

To verify that the trusted-storage migration completed successfully, have the enterprise license administrator perform the same tasks listed in Post-Migration Validations for the “Migrating Trusted Storage without License-Distribution Preservation”.

6

Migrating Administration Tools

This chapter provides assistance in migrating any license-administrator tools that you have deployed with the SA and that use the license server's XML endpoints.

- Migrating a Custom Administrative Tool
- Moving from Device Manager on SA to License Server Manager on LLS

Migrating a Custom Administrative Tool

The SA uses REST APIs (endpoints) in XML format to access and post license server data; the LLS uses REST APIs in JSON format. You can rewrite your administrative tool using the new APIs for LLS. The following table provides a reference for mapping the functionality between the XML and JSON formats of the two license servers.

Table 6-1 ▪ API-to-API Translation Matrix

XML API for SA	JSON API for LLS	HTTP Method
/fne/xml/reservation	`/api/1.0/reservationgroups` `/api/1.0/reservationgroups/`*`gid`*`/reservations`	GET
/fne/xml/setreservation	`/api/1.0/reservationgroups`	POST
/fne/xml/features	`/api/1.0/features` `/api/1.0/clients`	GET
/fne/xml/features?detailed=true	`/api/1.0/features` `/api/1.0/clients`	GET
/fne/xml/features/*feature_name*	`/api/1.0/features` `/api/1.0/clients`	GET

Table 6-1 ▪ API-to-API Translation Matrix (cont.)

XML API for SA	JSON API for LLS	HTTP Method
/fne/xml/features/*feature_name*?detailed=true	`/api/1.0/features` `/api/1.0/clients`	GET
/fne/xml/overages	`/api/1.0/overages` `/api/1.0/clients`	GET
/fne/xml/devices	`/api/1.0/clients`	GET
/fne/xml/devices/*device_name*	`/api/1.0/clients`	GET
/fne/xml/devices/device_name (DELETE method)	`/api/1.0/clients`	DELETE

/fne/xml/setproperties	`/api/1.0/configuration`
	`/api/1.0/hostids`
	`/api/1.0/hostids/selected`
		POST

/fne/xml/properties	`/api/1.0/configuration` `/api/1.0/hostids`	GET
/fne/xml/errorlog	Corresponding endpoint not supported	N/A
/fne/xml/accesslog	Corresponding endpoint not supported	N/A
/fne/xml/diagnostics	`/api/1.0/version` Only the server version provided (no directly corresponding endpoint supported)	GET
/fne/bin/offline-capability-request	`/api/1.0/capability_request/offline`	GET
/fne/bin/offline-capability-response	`/api/1.0/capability_response`	POST
/fne/xml/activation	`/api/1.0/activation_request`	POST
/fne/xml/offline-activation	`/api/1.0/activation_request/offline`	POST
/fne/xml/start-sync-to-fno	`/api/1.0/sync_message`	GET
/fne/xml/timers	Corresponding endpoint not supported	N/A
/fne/bin/capability (used by the test tool `caprequestutil` and the FlexNet Embedded client toolkit `capabilityrequest` example)	`/fne/bin/capability`	POST
/fne/bin/offlinesync	`/api/1.0/sync_message/offline`	GET
/fne/bin/offlinesyncack	`/api/1.0/sync_ack/offline`	POST

Moving from Device Manager on SA to License Server Manager on LLS

The FlexNet License Server Manager is the UI administrator tool for the LLS, comparable to the Device Manager for the SA. Instructions for installing License Server Manager are found in the "Setting Up the FlexNet License Server Manager" chapter in the *FlexNet License Server Manager Guide*, which is available from the Revenera Product Documentation site (https://docs.revenera.com).

Differences Between the License Server Manager and the Device Manager

The following are some major differences between the License Server Manager and the Device Manager:

- The **Reservations** view in the License Server Manager is designed to manage license reservations using LLS reservation functionality. This functionality orders reservations in a group-hostid-entry hierarchy and uses JSON-formatted reservation definitions. No mechanism is available to migrate the reservation definitions from the SA to the LLS.

  See the "More About License Server Functionality" in the FlexNet Embedded License Server Administration Guide.

- The **Device Details** page in the License Server Manager does not contain a **Reserved Features** section, showing the reserved feature counts for a specific client device, as it did in the Device Manager. However, in the License Server Manager, you can view the total reserved count *for a specific feature* on the **Feature Details** page.

- Similar to the **Properties and Status** view in the Device Manager, the **Properties** view in the License Server Manager shows properties and license server policies specific to the LLS. Refer to the following resources for more information:

  - Migrating the License Server Configuration chapter for a mapping of policies between the LA and the SA
  - "Reference: Policy Settings for the License Server" appendix in the FlexNet Embedded License Server Producer Guide for a description of the LLS policies and their corresponding labels in the License Server Manager
  - The "Installing and Running the License Server" chapter in the FlexNet Embedded License Server Producer Guide for a description of LLS local-environment properties used to run the license server service

- Search and sort operations are not supported in the License Server Manager.

- Because the LLS does not support REST APIs that correspond to the SA's `/fne/xml/errorlog` and `/fne/xml/accesslog` REST APIs, no log data can be retrieved and displayed. Consequently, the License Server Manager has no **Server Logs** view.

7

Other Migration Considerations

This chapter provide information about migrating other SA components to the LLS:

- Migrating Hostid Customizations
- Migrating Reservations

Migrating Hostid Customizations

If you have deployed the producer-defined hostid feature in the SA, you need to do the following to migrate this feature:

- Create a custom shared library to conform to the new interfaces defined in the LLS for custom hostid retrieval.
- Configure the LLS to enable support for producer-defined hostids.

For details, refer to the “Advanced License Server Features” chapter in the FlexNet Embedded License Server Producer Guide.

Migrating Reservations

If your SA customer enterprises use the device-based or user-based reservation features—either through direct XML API access or through a custom administrator tool—you and the enterprise need to adjust the current SA reservation “system” to accommodate the different behavior and format of the reservations on the LLS. (No mechanism exists to migrate the reservations.)

Additionally, if you intend to use user-based reservations, you need to upgrade your client application to the FlexNet Embedded 2016 or later client SDK.

Refer to the following sources for more details:

- The “More About Basic License Server Functionality” chapter in the FlexNet Embedded License Server Producer Guide

- The *FlexNet License Server Manager Guide* that is available from the Revenera Product Documentation site (https://docs.revenera.com).
- The “Using the FlexNet License Server Administrator Command-Line Tool” chapter in the FlexNet Embedded License Server Administration Guide