

SAMSUNG ELECTRONICS

Knox E-FOTA On-Premises

Troubleshooting & Logging Guide

Version: 1.3

Last Updated: March 2026

Table of Contents

1.	Introduction	4
1.1.	Purpose of this document	4
1.2.	DFM Modules	4
2.	Troubleshooting	5
2.1.	Service-related	5
2.1.1.	Login	5
2.1.2.	License	5
2.1.3.	Device upload and enrollment (using bulk actions)	5
2.1.4.	Device Tag Management	6
2.1.5.	Device Data Extraction	6
2.2.	Service infrastructure-related	6
2.2.1.	Service Running Hardware (H/W) and Network	6
2.2.2.	Showing Error Messages	7
2.3.3.	Error When Starting Mysql	7
3.	Logging	8
3.1.	Auto Install Related Log Files	8
3.2.	The Server Log Files	8
3.2.1.	Fetch Logs for the Container	8
3.2.2.	DFM Database Container (MySQL)	9
3.2.3.	DFM Firmware Storage Container (MinIO)	10
3.2.4.	DFM Core Container	10
3.2.5.	DFM Admin Console Container	10
3.2.6.	HAProxy container	10
3.3.	How to extract log data for DFM Core and Admin Console Servers	10
3.3.1.	DFM Core Logs	10
3.3.2.	Admin Console Logs	10
3.4.	Device (Client) Logs	11
3.4.1.	Device Log	11
	APPENDICES	12
	Appendix A. Terms and Abbreviations	12

Table of Figures

【Figures】

Fig 1-1 Knox E-FOTA On-Premises Architecture	4
--	---

【Tables】

Table 3-1 Log files from the DFM Modules (Servers)	8
Table 3-2 The flow to get device logs	10

1. Introduction

1.1. Purpose of this document

This document is intended **for all personnel using the Knox E-FOTA On-Premises service, including Samsung personnel and external customers.**

The purpose of this document is to provide guidance on how to address issues more quickly and proactively for a smoother on-premises service experience.

This document contains the following contents:

- How to perform troubleshooting tasks
- How to get log data related to the issue

If you're not able to resolve the issue after following these steps, follow the issue raising process as provided by the Knox E-FOTA On-Premises sales team. Be sure to include a log file when raising an issue, as this file can help expedite the troubleshooting process for Samsung engineers.

1.2. DFM Modules

This section presents the components of the DFM Module to help you understand what it is and how it works. and describes the deliverables related to the Knox E-FOTA On-Premises service.

In short, the customer installs the server infrastructure in their environment. The DFM Module enables the E-FOTA On-Premises service and contains the following core modules:

- **DFM Admin Console Server:** The frontend module that provides the graphical user interface (UI) on the Google Chrome browser, accessible by the customer's IT admins.
 - **DFM Core Server:** The backend module to manage device (client app) actions, which is integrated into the device using RESTful APIs from the client.
 - **DFM Database:** The MySQL-based database containing all service-related data, including device models and their IDs, as well as policy dependencies in Knox E-FOTA On-Premises campaigns.
 - **DFM Firmware Storage Management:** Firmware file storage to download files from the client app.
- Proxy:** Used for redirection between the outer and DFM modules, and for AP gateway and SSL certification.

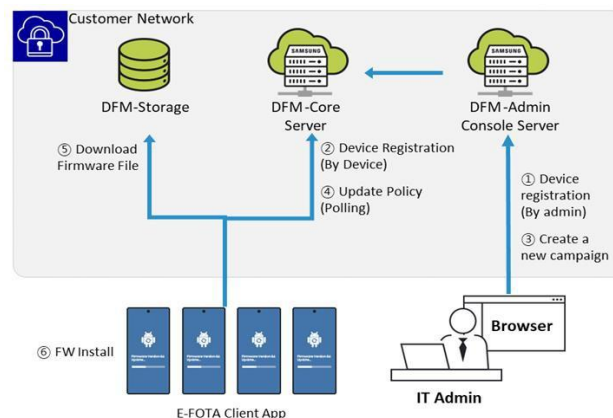


Fig 1-1 Knox E-FOTA On-Premises Architecture

2. Troubleshooting

Samsung provides troubleshooting assistance to support the continuous usability of the service. This chapter systematically guides you through the steps to address an issue, but doesn't cover all possible cases.

If you're not able to resolve the issue after following these steps, or if you suspect the issue is on the device side, then follow the issue raising process as provided by the Knox E-FOTA On-Premises sales team.

2.1. Service-related

If you encounter an issue while using Knox E-FOTA On-Premises, you can refer to the following information to try to resolve the issue before you raise a support ticket. When an issue occurs, an error message is shown in the top-right corner of the console.

2.1.1. Login

Item	Error Message	Description
Login	"User account is locked"	The account was locked because the password was entered incorrectly more than 5 times during sign-in. Contact your super admin to unlock the account.

2.1.2. License

Item	Error Message	Description
License	"The license is not valid"	No license was uploaded, or the license isn't valid.
Upload License	"Only *.lic files can be uploaded"	The license isn't in the correct file format. Try uploading a "*.lic" file provided by Samsung.

2.1.3. Device upload and enrollment (using bulk actions)

Item	Error Message	Description
Bulk Action in CSV	"Enter at least one device"	No device data could be imported from the CSV file.
Bulk Action in CSV	"There is a device that has more than 10 tags. Check csv file"	At least one row in the CSV file contains more than 10 tags. Make sure each row has a maximum of 10 tags.
Bulk Upload	"You have exceeded the maximum number"	The number of devices you're uploading is higher than your license allows. You can either contact your local reseller to purchase a license, or remove some devices from your CSV file.
Bulk Upload	"Enter at least one campaign."	The Assign Campaign option was selected during the CSV upload, but a campaign wasn't assigned. Either assign a campaign or deselect Assign Campaign .

Bulk Upload	"Can't upload more than 10000 devices. Check csv file"	Your CSV file contains more than 10,000 devices. Starting from Knox E-FOTA On-Premises 1.0.1.6, more than 10,000 devices can be uploaded in one CSV file.
-------------	--	--

2.1.4. Device Tag Management

Item	Error Message	Description
Devices > Actions > Manage device tags	"The maximum number of tags is ten "	More than 10 tags were entered for the device. Each device can have a maximum of 10 tags.
Devices > Actions > Manage device tags	"Duplicate tag"	The device already has this tag. Make sure each tag is only entered once for the device.

2.1.5. Device Data Extraction

Item	Error Message	Description
Devices > Actions > View Device Deletion Log	"Export failed"	The device data couldn't be downloaded as a CSV file. Try exporting the data again.
Device Management	"Export Failed"	The device data couldn't be downloaded as a CSV file. Try exporting the data again.

2.2. Service infrastructure-related

This section describes how to troubleshoot potential issues regarding to the infrastructure running the service on the DFM Modules (servers) or on the device (client app).

2.2.1. Service Running Hardware (H/W) and Network

The recommended hardware and network infrastructure for optimal operation of this service differs based on the product type (that is, PoC or Commercial Products—please refer to "[Installation Guide for Knox E-FOTA On-Premises](#)" for more information).

While using this service, if any network jamming occurs, please check the recommended values for hardware or network. Note that the recommended values represent the minimum specifications, so you may want to consider higher specifications as needed).

To avoid service jamming, the customer should provide the Samsung installer with the proper hardware and network environment during the initial installation.

2.2.2. Showing Error Messages

While using this service, you may unexpectedly see some error messages that can have an impact on service status. Refer to the following table for a description and resolution for each error message.

Item	Error Message	Description
Server	"Service Unable (503)"	There is an issue with the network capacity among the devices and DFM Modules (servers). Please raise a ticket with Samsung.
Device	"Firmware download failed (567)"	To ensure proper service performance, the device needs to have at least 2 GB of free storage to be used for firmware downloaded from the server. Free up some storage on the device, then try again.
Device	"Firmware install failed (773)"	To ensure proper service performance, the device needs to have at least 2 GB of free storage to be used for the firmware installation process. Free up some storage on the device, then try again.

2.3.3. Error When Starting Mysql

You may encounter unexpected error messages when starting mysql:

```
dfm start dfm-mysql
> Error: CI runtime error: crun: unknown version specified
> Error: exit status 126
```

To resolve, update system packages and podman with dnf, and reinstall crun:

```
sudo dnf update -y
sudo dnf update podman -y
sudo dnf list crun
sudo dnf install crun -y
```

Then restart MySQL with:

```
dfm start dfm-mysql
```

3. Logging

This section describes how to get log files for an issue you may encounter while using Knox E-FOTA On-Premises. These logs can be sent to Samsung for further analysis by Samsung engineers.

This on-premises service consists of servers (that is, DFM Modules), and devices (that is, a service app running on each device) used by the device users. The following sections explain how to get log files from both the server and device.

3.1. Auto Install Related Log Files

The DFM auto install tool logs the installation process.

The log file is created in the log folder where the automatic installation package is decompressed.

For example:

```
//decompress package
$ tar -zxvf OnPrem.tar.gz -C /tmp

//after run install script
$ ls -al /tmp/OnPrem/log
-rw-rw-r--. 1 nightwatch nightwatch 23631 Dec  2 05:45 install.22.12.02.05:41:48.log
```

3.2. The Server Log Files

The DFM Modules use the Java “log4j2” package to provide logging information. All log file names, logging filters, and file rollover policies are controlled by the log4j2 configuration file, such as:

`${efota.dir}/core/log4j2.xml`

`${efota.dir}/admin/log4j2.xml`

Each sub-module of the DFM Module creates its own log files. The following table lists these files.

Sub-module (Functionality)	File Names	Description
DFM Admin Console Server	/dfm/console/logs/logs.log	Admin Console Log
DFM Core Server	/dfm/core/logs/ACCESS/*.log	Access Log
	/dfm/core/logs/INFO/*.log	Others

Table 3-1 Log files from the DFM Modules (Servers)

A new log file for each sub-module is created every day, and saved for one month. However, files over 100 MB are split into two files. Files older than one month are automatically removed from the directory to optimize system space. To manage the logs, use the following commands.

3.2.1. Fetch Logs for the Container

To clearly describe which modules are being used, some basic terminology is defined here. The “Container” refers to the following:

Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and deploy it as one package. By doing so, thanks to the container, the developer can rest assured that the application will run on any other Linux machine regardless of any customized settings (to learn more, please refer to “[Installation Guide for Knox E-FOTA On-Premises](#)” provided as additional material).

This section describes how to fetch the logs from MySQL, MinIO, DFM Core, DFM Admin Console, HAProxy Servers (and refer to <https://docs.docker.com/engine/reference/commandline/logs/>). Next, the commands to extract the log file are provided.

3.2.2. DFM Database Container (MySQL)

Here is a command to retrieve logs and display them in the terminal at the time of execution:

```
$ docker logs --tail 30 -f dfm-mysql
```

3.2.3. DFM Firmware Storage Container (MinIO)

Here is a command to retrieve logs and display them in the terminal at the time of execution:

```
$ docker logs --tail 30 -f dfm-minio
```

3.2.4. DFM Core Container

Here is a command to retrieve logs and display them in the terminal at the time of execution:

```
$ docker logs --tail 30 -f dfm-core
```

3.2.5. DFM Admin Console Container

Here is a command to retrieve logs and display them in the terminal at the time of execution:

```
$ docker logs --tail 30 -f dfm-console
```

3.2.6. HAProxy container

Here is a command to retrieve logs and display them in the terminal at the time of execution:

```
$ docker logs --tail 30 -f dfm-proxy
```

3.3. How to extract log data for DFM Core and Admin Console Servers

This section describes how to extract log files for the DFM Core and Admin Console.

3.3.1. DFM Core Logs

For the core server, which is the dfm-core container, log data is generated in this directory:

```
/dfm/core/logs/INFO
```

3.3.2. Admin Console Logs

For the admin server, which is the dfm-console container, log data is generated in this directory:

```
/dfm/console/logs
```

3.4. Device (Client) Logs

Refer to the following steps to get log files from each device.

3.4.1. Device Log

This section shows how to get dumpstate logs from a device. Device log buffers clear after 15 minutes, make sure you reproduce the issue within this period of time.

On devices running Android 15 and higher, you may have to disable **Auto Blocker** before you can access SysDump. To do this, go to your device's settings, and disable **Security and privacy > Auto Blocker**.

For Samsung mobile devices, follow these steps to generate a log file:

- Launch the **Phone** app.
- Perform the following steps in sequence:

Flow	Description
#9900#	Enter the string to launch SysDump. (This is confidential for Samsung mobile devices)
SILENT LOG : OFF	Tap Silent log : Off , and set it to OFF .
Default	Tap Default .
{ DO TEST }	Do test to reproduce the issue.
*#9900#	Enter the string to launch debug mode again.
RUN DUMPSTATE/LOGCAT/MODEMLOG	Tap RUN DUMPSTATE/LOGCAT/MODEM LOG .
COPY TO SDCARD(INCLUDE CP RAMDUMP)	Tap COPY TO SDCARD(INCLUDE CP RAMDUMP)
'Log' folder in the device	① Connect your device to a PC through USB ② Open the "Log" folder on the device
Copy "Log" folder to the local PC	
Compress the "Log" folder	Compress the log folder if needed.
Attach the compressed "Log" folder as a file	The compressed log folder can then be shared through email or with Samsung.

Table 3-2 The flow to get device logs

APPENDICES

This section presents more detailed explanations for each term.

Appendix A. Terms and Abbreviations

App: Application in Client

CAT: Category Codes

DE: Docker Enterprise

DFM: Device Firmware Management

DISA: Defense Information Systems Agency

DNS: Domain Name Server

DoD: Department of Defense

DoDI: DoD Instruction

ECMP: Equal-Cost Multi-Path routing

E-FOTA: Enterprise – Firmware over the Air

HA: High Availability

H/W: Hardware

ID: Identification

KE: Knox E-FOTA (Brand)

OS: Operating System

PoC: Proof of Concept

SSL: Secure Sockets Layer

TLS: Transport Layer Security (TLS)

UI: User Interface

< **EOF** (End Of File) >