SW Encryption Layer

Certifiable Encryption

Document no.: GEC-US-0082
Author: Galleon Embedded Computing
Revision: 1.0.7
Date: July 14, 2022

Preface

Conventions

IEC prefixes are used to separate between decimal and binary multiples.

- Decimal prefixes k (kilo), M (mega), G (giga), and T (tera) denote 10^3, 10^6, 10^9, and 10^{12} respectively.
- Binary prefixes Ki (kibi), Mi (mebi), Gi (gibi), and Ti (tebi) denote 2^{10}, 2^{20}, 2^{30}, and 2^{40} respectively.

Dates printed in numerical format are in the order day month year. E.g. 24.01.10 (or 24.01.2010) denotes January 24th, 2010.

Copyright

Limited Liability

Galleon Embedded Computing assumes no liability arising from any use of information provided within this document.

Revision History

Rev.	*Date*	*Changes*
1.0.0	06-May-21	First draft
1.0.1	13-Apr-22	Added missing sections.
1.0.2	27-Apr-22	Added section on Protocols
1.0.3	28-Apr-22	Updated section on Protocols
1.0.4	12-May-22	Added document number
1.0.5	18-May-22	Minor changes
1.0.6	21-Jun-22	Improved clarity of evaluated configuration
1.0.7	14-Jul-22	Added passphrase strength guidance

Note: Revision numbers are on the form X.Y.Z, where X indicates the major revision number, Y indicates minor revision number, and Z indicates revision number for editorial changes that do not alter the technical content of this document.

Document Definitions

List of keyword definitions for document automation. Variables set at time of document creation.

Keyword:

Subject	Certifiable Encryption
Title:	SW Encryption Layer
Product:	Encryption Module
Version:	1.0.7
Author:	aknowles@galleonec.com
Company:	Galleon Embedded Computing

Contents

1 Overview

This document describes the software encryption layer of the certifiable encryption system. The SW layer is based on Linux Unified Key Setup (LUKS) and dm-crypt, standard Linux packages that enable secure encryption of block devices.

Galleon provides some simple commands that are thin wrappers around LUKS. These wrappers make LUKS easier to work within the context of Galleon's Removable Data Modules (RDMs) but do not change the encryption mechanisms of LUKS.

1.1 Protocols

In addition to the local interface, the software environment supports the following network protocols: CIFS, NFS, FTP, TFTP, iSCSI, SSH, TLS, and SNMP. These and other network protocols may be enabled or disabled by an administrator by using standard system administrator commands available for the RedHat 8.4. In the default configuration SSH and SNMP are enabled for remote administration and monitoring, SSH and TLS are enabled for authentication, and CIFS and NFS are enabled after successful authentication for remote file access.

As root, run:

```
systemctl list-unit-files              # To see all available services
systemctl is-enabled <service>         # To check a single service
systemctl disable <service>            # To disable a service
```

Where <service> corresponds to the following network protocols:

`smb`	for CIFS
`nfs-server`	for NFS
`gec-key-server`	for TLS authentication
`ssh`	for SSH (administration and authentication)
`tftp`	for TFTP
`iscsid`	for iSCSI
`snmpd`	for SNMP

Administrators must only access the unit and perform management activities locally and not remotely. For local management, the administrator may use either a directly attached Keyboard, Video, and Mouse (KVM) or a cross-over cable (providing a direct, local connection between the TOE and the administrator's PC) to access the Red Hat operating system.

In the evaluated configuration, remote administration and authentication protocols SSH, TLS and SNMP are disabled.

The claimed evaluated configuration, 'locally managed with remote management disabled', complies with NIAP TD0606.

NIAP's technical decision is in part based upon the fact that the FDEEE and FDEAA Protection Profiles did not consider, nor did they include networking protocols as part of the security functional requirements, and as a result, did not include any requirements for addressing these protocols.

Therefore, as per the protection profiles, these protocols have not been examined as part of the required assurance activities and consequently the evaluation can make no claims about the networking protocols on the XSRS/G1S.

The customer must consider the impact of using both remote administration and authentication methods over the network in light of their planned deployment scenario, and factor this into their risk management decision. Galleon strongly recommends that network access be limited to a segregated private network at all times.

2 For System Administrators

In the evaluated configuration, the system administrator logs in to the XSRS/G1S using the local terminal as root.

2.1 Installing System Updates

Galleon will notify customers directly as software update packages applicable to the XSRS/G1S become available.

To install a Galleon provided software update package:

1. Connect to the system on GbE Port 0 with the default IP address 192.168.100.101.
2. Copy the update file from outside the system:
   `scp gec-xsr-sw-update-1.0.tar galleon@192.168.100.101:/tmp`
3. Login to the system as root.
4. Install the update:
   `gec-install-signed-update /tmp/gec-xsr-sw-update-1.0.tar`

The update will be automatically validated by the pre-installed `gec-install-signed-update` utility and will not continue if the embedded digital signature in the update cannot be verified.

Note: if disabled, it may be necessary to re-enable the SSH protocol briefly in order to transfer the update file to the system, or use an alternate method to copy the file to the system.

2.2 Checking the Current Version

To check which version of the Galleon SW is currently installed, run:

```
cat /etc/galleon/sw-version
```

2.3 Preparing an RDM

When formatting a new RDM, or formatting an RDM, run the rdm-initialize script.

```
rdm-initialize
```

This will create a RAID array from the RDM, and then prompt the administrator to enter a passphrase for encrypting the RAID partition. It will then format the encrypted partition with a file system.

The passphrase must be strong and have high entropy in order to protect the encrypted data. The passphrase should be at least 16 characters in length and should be randomly generated from a high entropy source from a set of possible characters including upper case letters, lower case letters, digits, and special characters.

Note that only the first 511 bytes of the passphrase will be used to create the encrypted partition. Longer passphrases will be silently truncated to 511 bytes. However, the administrator should limit the passphrase to 256 bytes in length, as the operator user will be limited to entering passphrases up to 256 bytes in length.

Note that this will be necessary both when the HW encryption keys are changed and when a cryptographic erase of the SW encryption layer is performed. In the former case, the RAID array

will need to be recreated, while in the latter case the RAID array will still exist. The rdm-initialize script will automatically handle these details.

For the SW encryption layer, the important command in this script is:

```
cryptsetup --batch-mode --use-random --hash=sha384 --cipher=aes-xts-plain64
--key-size=512 --pbkdf=pbkdf2 --type=luks2 --label=ERDM --keyslot-
cipher=aes-cbc-essiv:sha256 --keyslot-key-size=256 luksFormat $DEVICE
```

Where usually $DEVICE = /dev/md127, the RDM RAID array, but in the case of a single disk RDM will refer to directly to the disk device file (e.g. /dev/sdb).

2.4 Authentication: Unlocking an RDM

To unlock an existing encrypted RDM and mount the encrypted file system, run:

```
rdm-mount
```

This will prompt the user to enter the passphrase that was used to encrypt the RDM. If entered correctly, the RDM will be unlocked and mounted.

Note that only the first 511 bytes of the passphrase will be used to unlock the encrypted partition. Longer passphrases will be silently truncated to 511 bytes.

If the passphrase is entered incorrectly 3 times, further authentication attempts will not be permitted for the next 60 seconds.

The important command in this script is:

```
cryptsetup open $DEVICE rdm
```

Where usually $DEVICE = /dev/md127, the RDM RAID array, but in the case of a single disk RDM will refer to directly to the disk device file (e.g. /dev/sdb).

2.5 De-authentication: Locking an RDM

To lock an encrypted RDM and prepare the RDM for removal from the system, run:

```
rdm-unmount
```

This will stop services that use the RDM, attempt to unmount the filesystem, lock the encrypted partition, and stop the RAID array. Once this is done, it will no longer be possible to access data on the RDM without once again entering the passphrase to unlock the encrypted partition.

The important command in this script is:

```
cryptsetup close rdm
```

2.6 Cryptographic Erase

To perform a cryptographic erase of an encrypted RDM, run:

```
rdm-luks-erase
```

This will use the standard LUKS tool cryptsetup to erase the encrypted LUKS keys on the disk. Once erased, it will be impossible to decrypt the disk, even with the passphrase. All data on the RDM will be lost. The RDM will also be unmounted and locked.

The important command in this script is:

```
cryptsetup -y erase $DEVICE
```

Where usually $DEVICE = /dev/md127, the RDM RAID array, but in the case of a single disk RDM will refer to directly to the disk device file (e.g. /dev/sdb).

2.7 System Shutdown

To safely prepare the system for shutdown, run:

```
poweroff
```

Once completed, the system will typically transition to the compliant power-saving state “mechanical off” within 10 seconds. Then the power supply can safely be removed from the system.

If the power supply is removed unexpectedly, the system immediately transitions to the compliant power-saving state, mechanical off.

3 For Operators

3.1 SSH Authentication

The operator connects to the XSR via SSH using the operator account. Upon successful SSH authentication, the operator will be presented immediately with a request for a passphrase. Upon entering the passphrase successfully, the RDM will be unlocked and mounted, and the operator SSH session will be disconnected.

The passphrase may be up to 256 bytes long. Attempts to enter longer passphrases will result in an error and no authentication will be attempted.

If the passphrase is entered incorrectly 3 times, further authentication attempts will not be permitted for the next 60 seconds.

In the evaluated configuration, the SSH protocol is disabled.

3.2 Terminal Authentication

The operator can directly log in to the operator account with a connected keyboard and monitor. Once successfully logged in, the operator will be presented immediately with a request for a passphrase. Upon entering the passphrase successfully, the RDM will be unlocked and mounted, and the operator login session will be terminated.

The passphrase may be up to 256 bytes long. Attempts to enter longer passphrases will result in an error and no authentication will be attempted.

If the passphrase is entered incorrectly 3 times, further authentication attempts will not be permitted for the next 60 seconds.

3.3 TLS Authentication

The operator uses a TLS client to connect to a specified port (by default 9000) and perform mutual authentication with preconfigured TLS certificates. The TLS client then follows a simple protocol to transfer a username and passphrase (along with some protocol version and length bytes). This is the same protocol used for the HW encryption layer. The username field must be left blank (a zero-length byte string). The passphrase is used to unlock the LUKS encryption on the RDM.

The passphrase may be up to 256 bytes long. Attempts to enter longer passphrases will result in an error and no authentication will be attempted.

If the passphrase is entered incorrectly 3 times, further authentication attempts will not be permitted for the next 60 seconds.

In the evaluated configuration, the TLS protocol is disabled.

3.4 De-authenticating: Locking an RDM

Append the rdm-unmount command when connecting over SSH as the operator user in order to lock an encrypted RDM and prepare the RDM for removal from the system:

```
ssh operator@192.168.100.101 rdm-unmount
```

This will stop services that use the RDM, attempt to unmount the filesystem, lock the encrypted partition, and stop the RAID array. Once this is done, it will no longer be possible to access data on the RDM without once again entering the passphrase to unlock the encrypted partition.

In the evaluated configuration, the SSH protocol is disabled and only the administrator or a delegated user can lock the RDM.

3.5 Cryptographic Erase

Append the rdm-luks-erase command when connecting over SSH as the operator user in order to perform a cryptographic erase of an encrypted RDM:

```
ssh operator@192.168.100.101 rdm-luks-erase
```

This will use the standard LUKS tool cryptsetup to erase the encrypted LUKS keys on the disk. Once erased, it will be impossible to decrypt the disk, even with the passphrase. All data on the RDM will be lost. The RDM will also be unmounted and locked.

In the evaluated configuration, the SSH protocol is disabled and only the administrator or a delegated user can perform a cryptographic erase of the RDM.

3.6 System Shutdown

Append the poweroff command when connecting over SSH as the operator user in order to safely prepare the system for shutdown:

```
ssh operator@192.168.100.101 poweroff
```

Once completed, the system will typically transition to the compliant power-saving state “mechanical off” within 10 seconds. Then the power supply can safely be removed from the system.

In the evaluated configuration, the SSH protocol is disabled and only the administrator or a delegated user can prepare the system for shutdown.