Linux Disk Encryption With PKI Token

Setup Guide

Table of Contents

Prerequisites

Platform

The solution has been tested under the following platforms:

- Debian 9.1 (x86_64) running as a guest OS under VMWare ESXi

Software Packages

- OpenSC 0.16.0
  ```
  apt-get install opensc
  ```
- cryptsetup 1.7.3
  ```
  apt-get install cryptsetup
  ```

Hardware

The solution has been tested with the following hardware

- Microcosm PKI token (FT ePass2003)

Setup Guide

Single Encrypted Partition

This guide will demonstrate how to set up a single encrypted partition with the PKI token acting as a secure store for the encryption key.

For this guide we will assume the following:

- The partition to be encrypted is /dev/sda3
- The mapped name for this partition is ‘enc’
- The ID of the key on the PKI token is 010203
- You are using the root account

Steps

Initialize the PKI Token

Clean and initialize the token:

```
$> pkcs15-init -E

$> pkcs15-init --create-pkcs15 --profile pkcs15+onepin --label “something”
```

Generate the key-pair on the token:

```
$> pkcs15-init --generate-key rsa/2048 --id 010203 --key-usage sign,decrypt --auth-id 01 --label “disk-enc-key”
```

Setup the Encrypted Partition

Create the disk encryption key:

```
$> dd if=/dev/urandom of=luks-key bs=1 count=245
```

Encrypt the partition using the key:

```
$> cryptsetup luksFormat /dev/sda3 luks-key
```

Load the encrypted partition and format a filesystem on it:

```
$> cryptsetup --key-file=luks-key luksOpen /dev/sda3 enc

$> mkfs.ext4 /dev/mapper/enc
```

Mount and test the encrypted partition:

```
$> mkdir /mnt/enc

$> mount /dev/mapper/enc /mnt/enc
```

Check you can read/write files on /mnt/enc.

If all is OK, you must now umount and close the LUKS container before continuing:

```
umount /mnt/enc

cryptsetup luksClose enc
```

Secure the Encryption Key

Extract the public key from the token:

```
$> pkcs15-tool --read-public-key 010203 > 010203-pub.pem
```

Encrypt the disk encryption key using the token public key:

```
$> openssl rsautl -in luks-key -encrypt -pkcs -pubin -inkey 010203-pub.pem -out luks-key.enc
```

Securely delete the disk encryption key file:

```
$> shred -u luks-key
```

Mount & Test the Encrypted Partition

```
$> pkcs15-crypt --decipher --key 010203 --pkcs1 --raw --input luks-key.enc | cryptsetup --key-file=- luksOpen /dev/sda3 enc

$> mount /dev/mapper/enc /mnt/enc
```

Check you can read/write to /mnt/enc.

If all is OK, you must now umount and close the LUKS container before continuing:

```
$> umount /mnt/enc

$> cryptsetup luksClose enc
```

Mount the Encrypted Partition at Boot

Add the following line to the **`/etc/crypttab`** file:

```
enc       /dev/sda3       none       luks
```

Create a file in **`/etc/systemd/system`** called **`systemd-cryptsetup@enc.service`**. Note that the ‘enc’ in the name of that file is important because it relates to the ‘enc’ entry in **`/etc/crypttab`**.

Add the following to **`/etc/systemd/system/systemd-cryptsetup@enc.service`**

```
[Unit]

Description=EncDisk

DefaultDependencies=no

IgnoreOnIsolate=yes

Before=systemd-user-sessions.service

[Service]

Type=oneshot

ExecStart=/root/enc-disk-start

ExecStop=/bin/umount /mnt/enc && /sbin/cryptsetup luksClose %i

RemainAfterExit=yes
```

Next, create the `enc-disk-start` file in `/root`, and add the following text to it:

```
#!/bin/bash

/usr/bin/pkcs15-crypt --decipher --key 010203 --pkcs1 --raw --input /root/luks-key.enc -p $(/bin/systemd-ask-password "Enter Token PIN: ") | /sbin/cryptsetup --key-file=- luksOpen /dev/sda3 enc

/bin/mount /dev/mapper/enc /mnt/enc
```

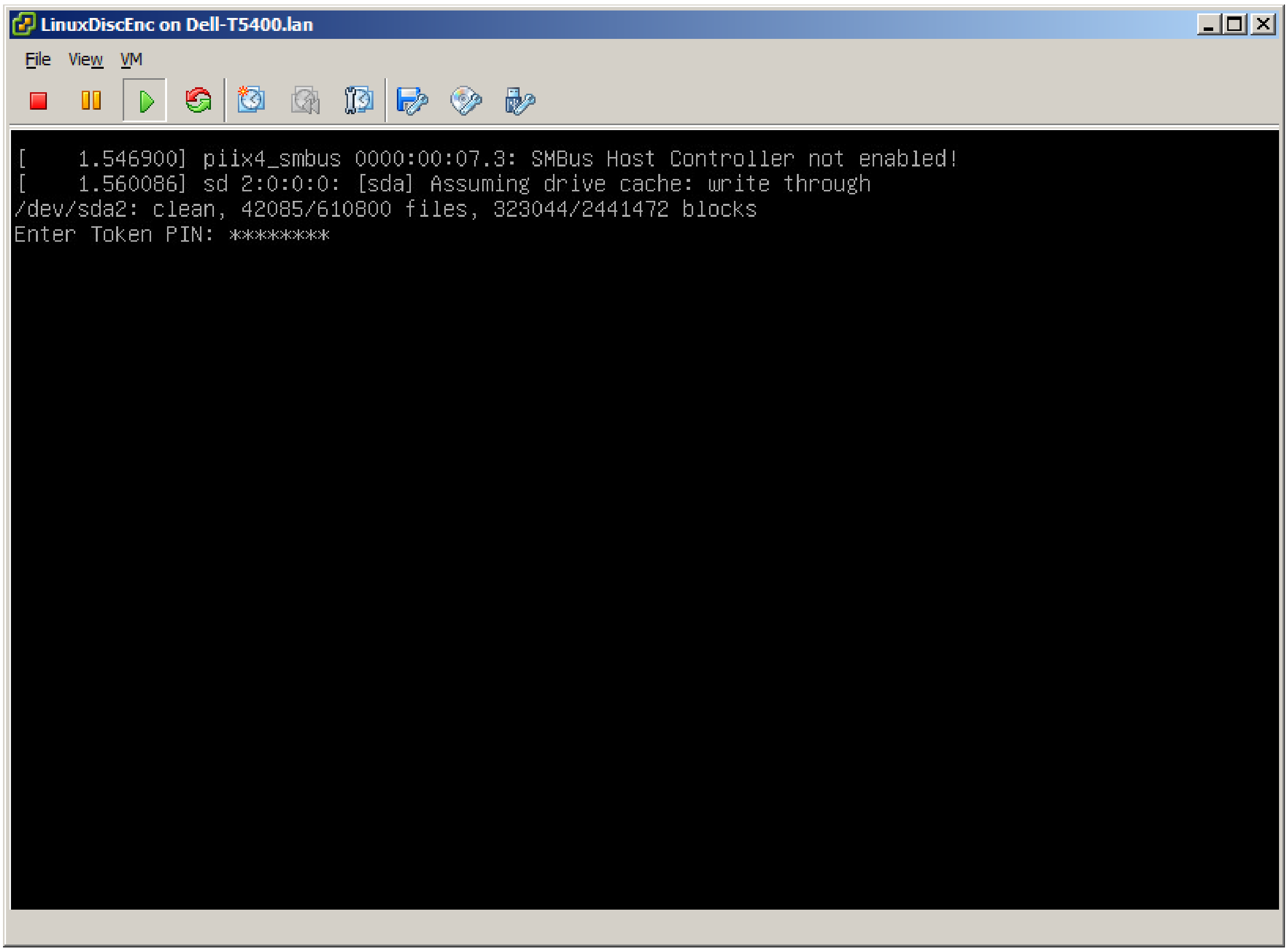
Now make the `/root/enc-disk-start` script executable:

```
$> chmod 700 /root/enc-disk-start
```

Now you can reboot your machine and test the solution:

```
$> reboot
```

You should see the `Enter Token PIN:` prompt at boot. Enter the token PIN then hit Enter.



```
[    1.546900] piix4_smbus 0000:00:07.3: SMBus Host Controller not enabled!
[    1.560086] sd 2:0:0:0: [sda] Assuming drive cache: write through
/dev/sda2: clean, 42085/610800 files, 323044/2441472 blocks
Enter Token PIN: ********
```

Log in to your system when prompted then check that **`/mnt/enc`** has your encrypted partition mounted on it.

```
$> ls -l /mnt/enc
```

That’s it. You now have an encrypted partition that is loaded at boot with the encryption key secured on a hardware token.

If you have any questions please contact us via one of the methods listed on the Support page.

Support

If you have any questions about the PKI product please contact Microcosm using one of following methods.

Technical Support & General Enquiries

Email: support@microcosm.com

Telephone: +44 (0) 117 983 0084

Sales/Ordering

Email: sales@microcosm.com

Telephone: +44 (0) 117 983 0084