Implementing Pervasive Encryption in SUSE Linux Enterprise Server

Focus on Pervasive Encryption for data-at-rest

Don Vosburg – Systems Engineer

SUSE

Dvosburg@suse.com

Agenda

- **Protected volume support in SUSE Linux Enterprise Server**
- **Getting started with pervasive encryption for data volumes**
- **Introducing the lab**
- **Working with data-at-rest encryption**
  - Encrypting new ECKD partitions in an LVM
  - Re-encipher secure keys
- **What is next?**

Protected volume encryption support in SLES

What is required:

- **Linux kernel modules**
  - paes_s390x
  - pkey
  - dm-crypt
- **cryptsetup utility >= 2.0.3**
- **zkey and zkey-cryptsetup in s390-tools**

SLES 12 SP4

- paes_390x/pkey/dm-crypt ✓
- cryptsetup = 1.6.4 ⊘
- zkey/zkey-cryptsetup ✓ / ⊘
  - s390-tools = 2.1.0-13.6

SLES 15

- paes_390x/pkey/dm-crypt ✓
- cryptsetup = 2.0.1 ⊘
- zkey/zkey-cryptsetup ✓ / ⊘
  - s390-tools = 2.1.0-12.8

SLES 15 SP1

- paes_390x/pkey/dm-crypt ✓
- cryptsetup = 2.0.5 ✓
- zkey/zkey-cryptsetup ✓
  - s390-tools = 2.1.0-19.27

Summing up the SLES support picture

For SLES12 SP4 and SLES15:

- Volume encryption can be done with a clear key in LUKS1 or plain mode
- Takes advantage of the CPACF hardware acceleration and Crypto Express card

For SLES15 SP1:

- Volume encryption can be done with a secure/protected key combination in LUKS2 or plain mode
- Takes advantage of the CPACF hardware acceleration and Crypto Express card

Getting started with pervasive encryption for data volumes

Resources

IBM documentation

- Pervasive Encryption for Data Volumes (Updated June 2019)
  - Documents using LUKS2 or plain modes
  - Use SLES15 SP1
- Getting start with pervasive encryption (September 2017)
  - Documents using plain mode
  - Use with SLES12 SP4, SLES 15 and SLES15 SP1

Community documentation

- LUKS (Linux Unified Key Setup)
  - Frequently Asked Questions

Recommendation

Use LUKS versus plain

- Benefits of using LUKS (from cryptsetup FAQ)
  - protect the user from a lot of common mistakes
  - multiple user keys with one master key
  - anti-forensic features
  - metadata block at start of device

Plain dm-crypt is for experts!

Useful for swap or scratch space

Introducing the lab

Testing pervasive encryption for data volumes

IBM z13

- Located in SUSE headquarters – Nürnberg, Germany

SLES15 SP1 installed in an LPAR

- CPACF enabled
- CryptoExpress
  - One CEX5C domain assigned - not best practice for production deployments!

What if we do not have access to a TKE (Trusted Key Entry) workstation?

CCA is a binary only package provided by IBM

- Main landing page for CCA package, documentation and much more
  - wget https://public.dhe.ibm.com/security/cryptocards/pciecc3/CCA/csulcca-6.0.13-08.s390x.rpm
- Installation and configuration documentation for CCA
  - Starts at Chapter 29 page 1107

IMPORTANT:

- **USE A TKE WORKSTATION FOR PRODUCTION KEY TASKS!!**
- **CCA SHOULD ONLY BE USED FOR TESTING**

Interacting with the CryptoExpress card using the CCA

Load the AES master key

- panel.exe –h for help
- panel.exe –l for interactive menus to load master key parts
  - I preferred using CLI options
- First, middle and last key parts
  - Must enter 64 character hex string for each part
  - Suggest concatenated two uuids from uuidgen removing the dash characters “-”
- Problems encountered
  - Use journalctl to look at the panel.exe messages
  - Add root to the following groups: cca_setmk, cca_cmkp, cca_lfmkp, cca_clrmk, cca_admin

Set the AES master key so it can be used

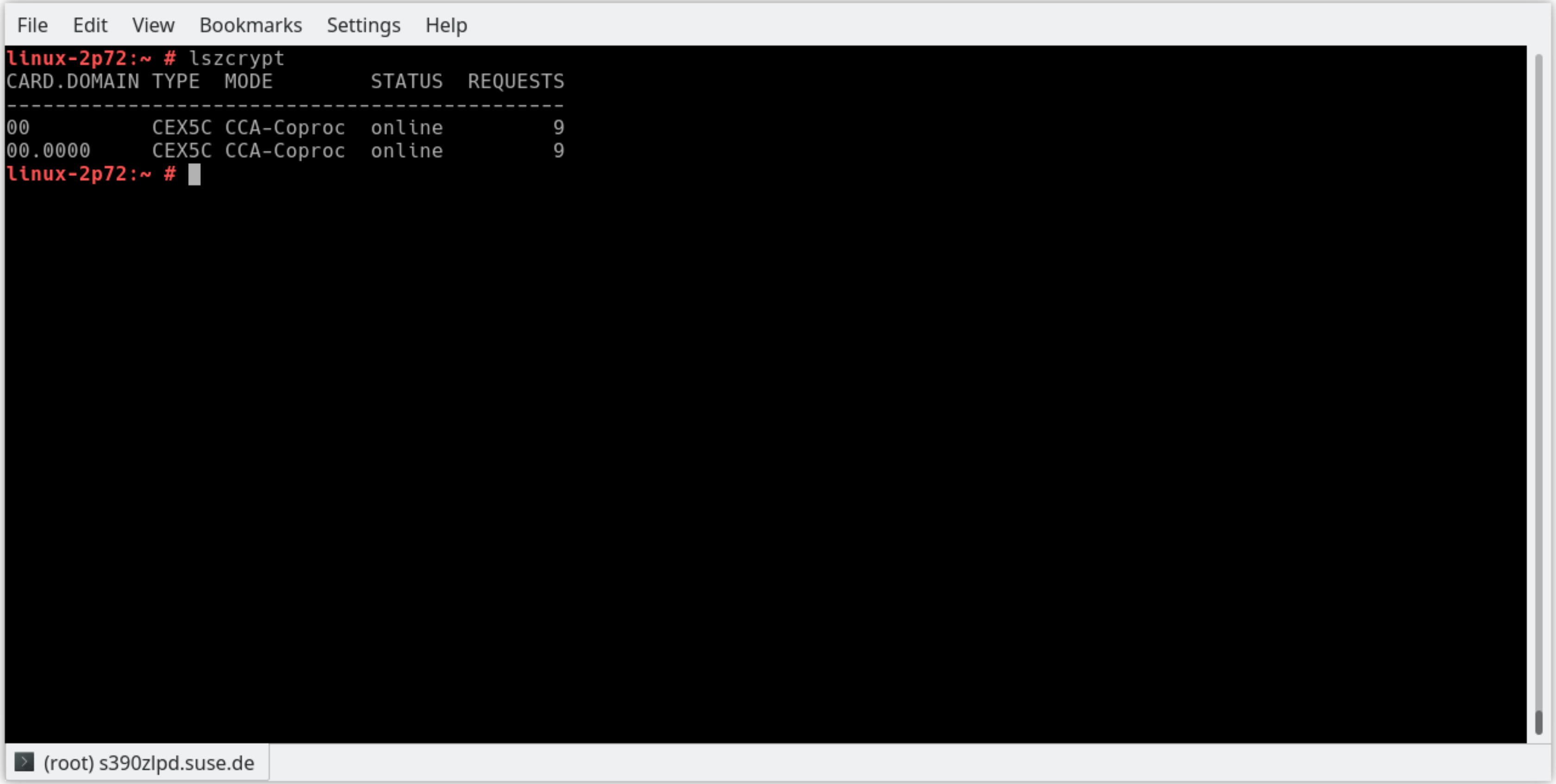
Rotating in a new master key

- Interact with NEW, CURRENT, OLD keys on CryptoExpress card

Working with data-at-rest encryption

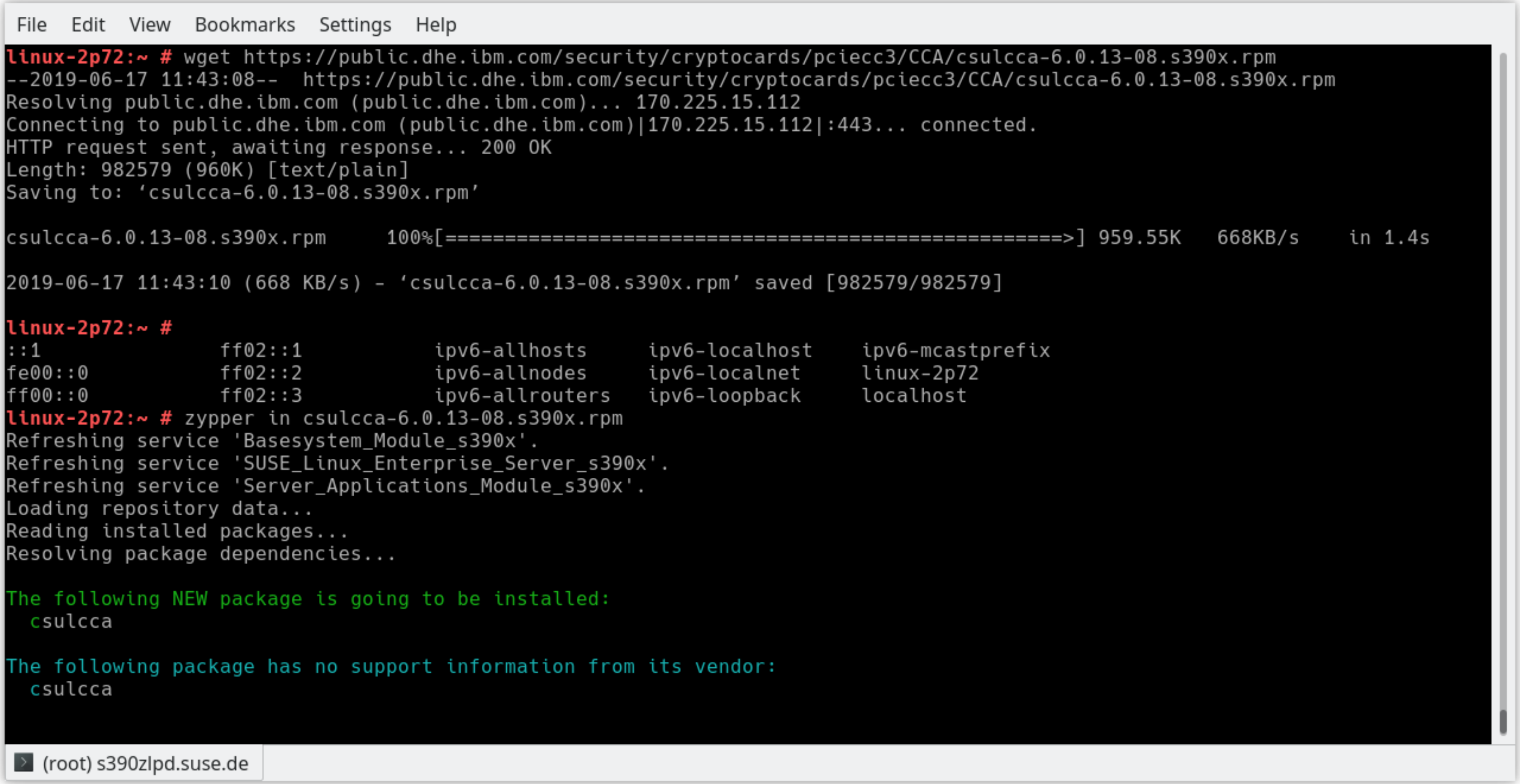
Encrypting new ECKD partitions in an LVM

Verify a CCA CryptoExpress domain exists



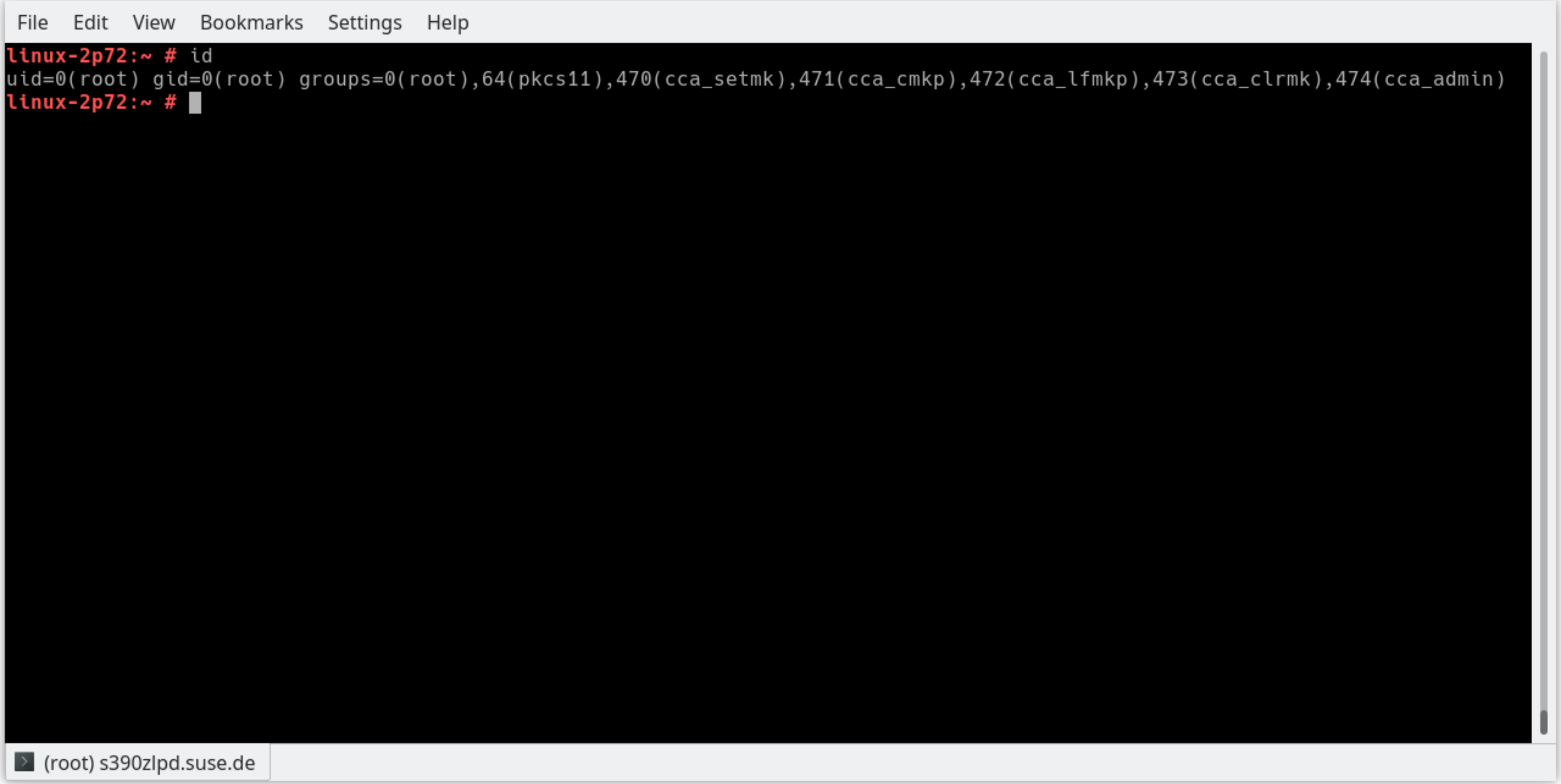
```
linux-2p72:~ # lszcrypt
CARD.DOMAIN TYPE  MODE        STATUS  REQUESTS
----------------------------------------------
00          CEX5C CCA-Coproc  online         9
00.0000     CEX5C CCA-Coproc  online         9
linux-2p72:~ #
```

Installing the IBM CCA binary only package



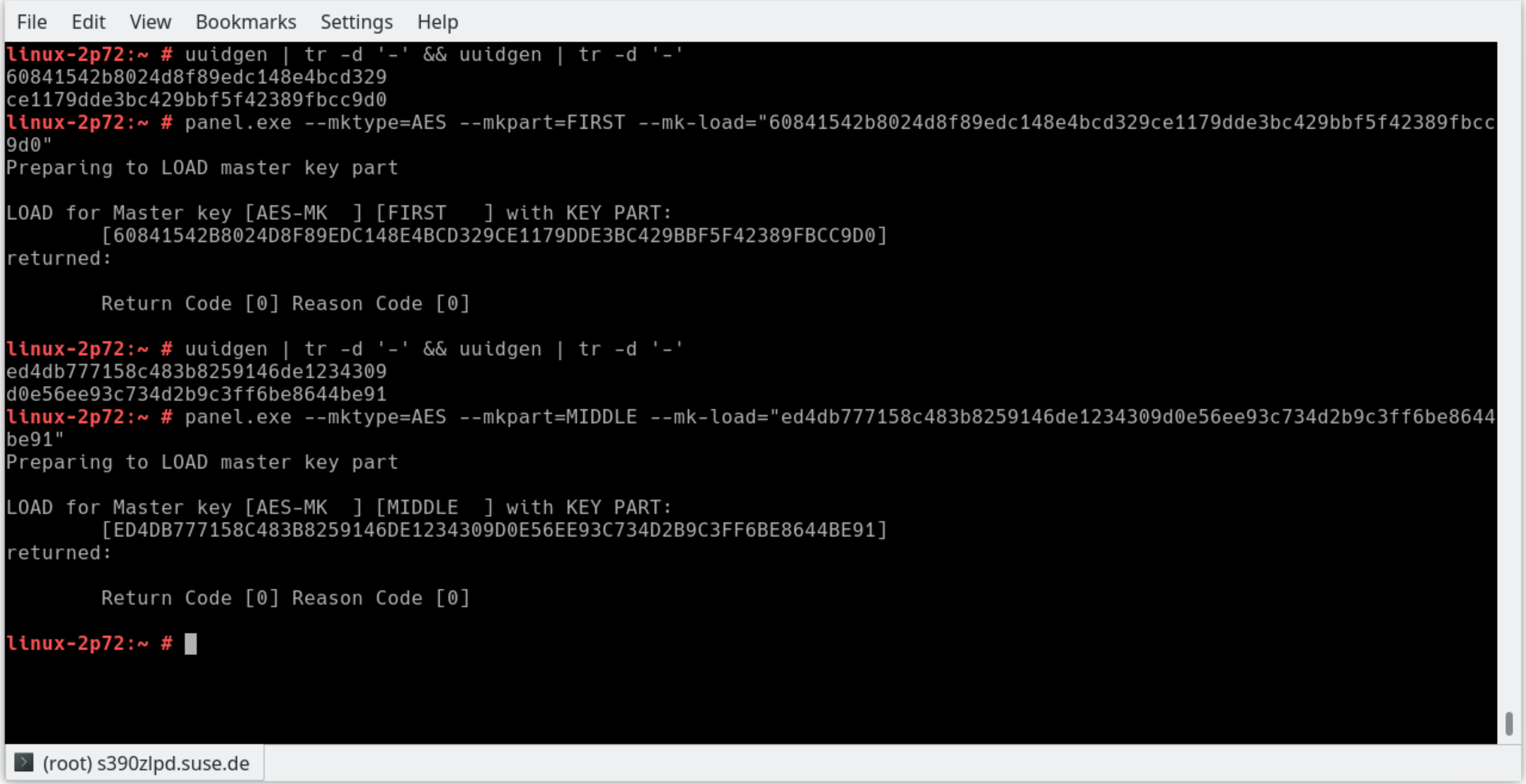
```
linux-2p72:~ # wget https://public.dhe.ibm.com/security/cryptocards/pciecc3/CCA/csulcca-6.0.13-08.s390x.rpm
--2019-06-17 11:43:08--  https://public.dhe.ibm.com/security/cryptocards/pciecc3/CCA/csulcca-6.0.13-08.s390x.rpm
Resolving public.dhe.ibm.com (public.dhe.ibm.com)... 170.225.15.112
Connecting to public.dhe.ibm.com (public.dhe.ibm.com)|170.225.15.112|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 982579 (960K) [text/plain]
Saving to: ‘csulcca-6.0.13-08.s390x.rpm’

csulcca-6.0.13-08.s390x.rpm     100%[=====================================================>] 959.55K   668KB/s    in 1.4s

2019-06-17 11:43:10 (668 KB/s) - ‘csulcca-6.0.13-08.s390x.rpm’ saved [982579/982579]

linux-2p72:~ #
::1                 ff02::1             ipv6-allhosts     ipv6-localhost     ipv6-mcastprefix
fe00::0             ff02::2             ipv6-allnodes     ipv6-localnet      linux-2p72
ff00::0             ff02::3             ipv6-allrouters   ipv6-loopback      localhost
linux-2p72:~ # zypper in csulcca-6.0.13-08.s390x.rpm
Refreshing service 'Basesystem_Module_s390x'.
Refreshing service 'SUSE_Linux_Enterprise_Server_s390x'.
Refreshing service 'Server_Applications_Module_s390x'.
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following NEW package is going to be installed:
  csulcca

The following package has no support information from its vendor:
  csulcca
```

Make root member of the CCA groups



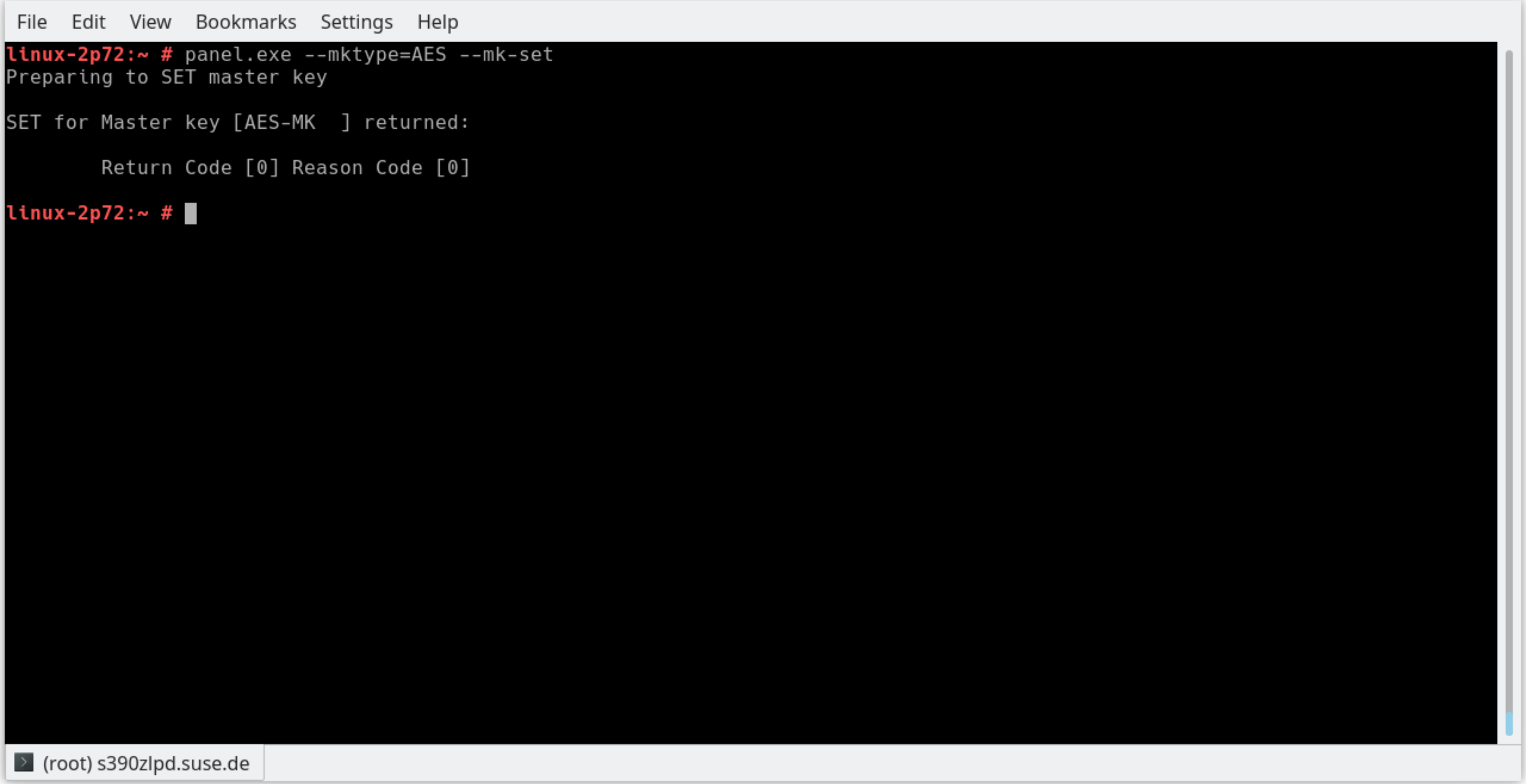
```
linux-2p72:~ # id
uid=0(root) gid=0(root) groups=0(root),64(pkcs11),470(cca_setmk),471(cca_cmkp),472(cca_lfmkp),473(cca_clrmk),474(cca_admin)
linux-2p72:~ # 
```

Load NEW master key parts



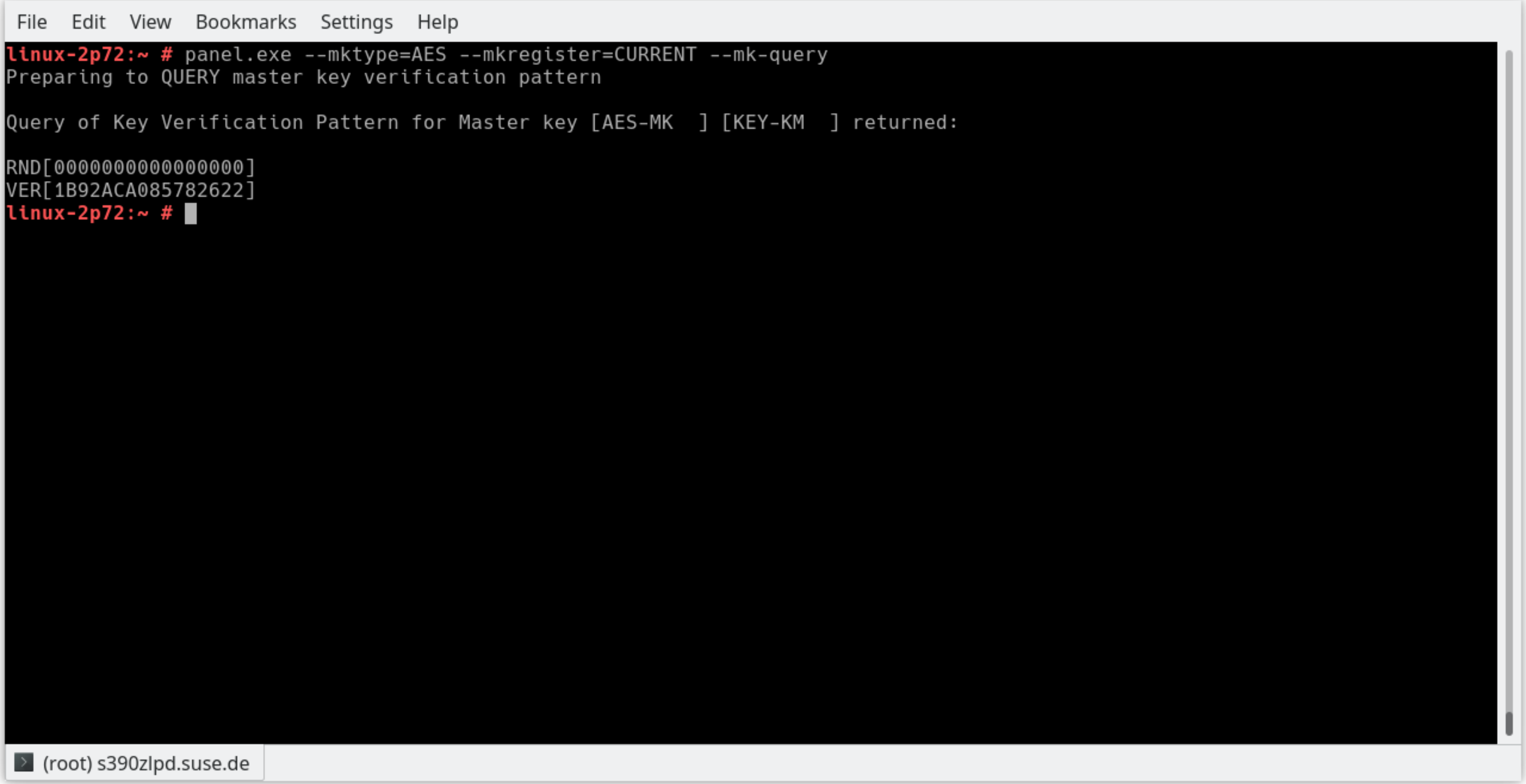
```
linux-2p72:~ # uuidgen | tr -d '-' && uuidgen | tr -d '-'
60841542b8024d8f89edc148e4bcd329
ce1179dde3bc429bbf5f42389fbcc9d0
linux-2p72:~ # panel.exe --mktype=AES --mkpart=FIRST --mk-load="60841542b8024d8f89edc148e4bcd329ce1179dde3bc429bbf5f42389fbcc
9d0"
Preparing to LOAD master key part

LOAD for Master key [AES-MK   ] [FIRST    ] with KEY PART:
        [60841542B8024D8F89EDC148E4BCD329CE1179DDE3BC429BBF5F42389FBCC9D0]
returned:

        Return Code [0] Reason Code [0]

linux-2p72:~ # uuidgen | tr -d '-' && uuidgen | tr -d '-'
ed4db777158c483b8259146de1234309
d0e56ee93c734d2b9c3ff6be8644be91
linux-2p72:~ # panel.exe --mktype=AES --mkpart=MIDDLE --mk-load="ed4db777158c483b8259146de1234309d0e56ee93c734d2b9c3ff6be8644
be91"
Preparing to LOAD master key part

LOAD for Master key [AES-MK   ] [MIDDLE   ] with KEY PART:
        [ED4DB777158C483B8259146DE1234309D0E56EE93C734D2B9C3FF6BE8644BE91]
returned:

        Return Code [0] Reason Code [0]

linux-2p72:~ # 
```

Set NEW master key as CURRENT



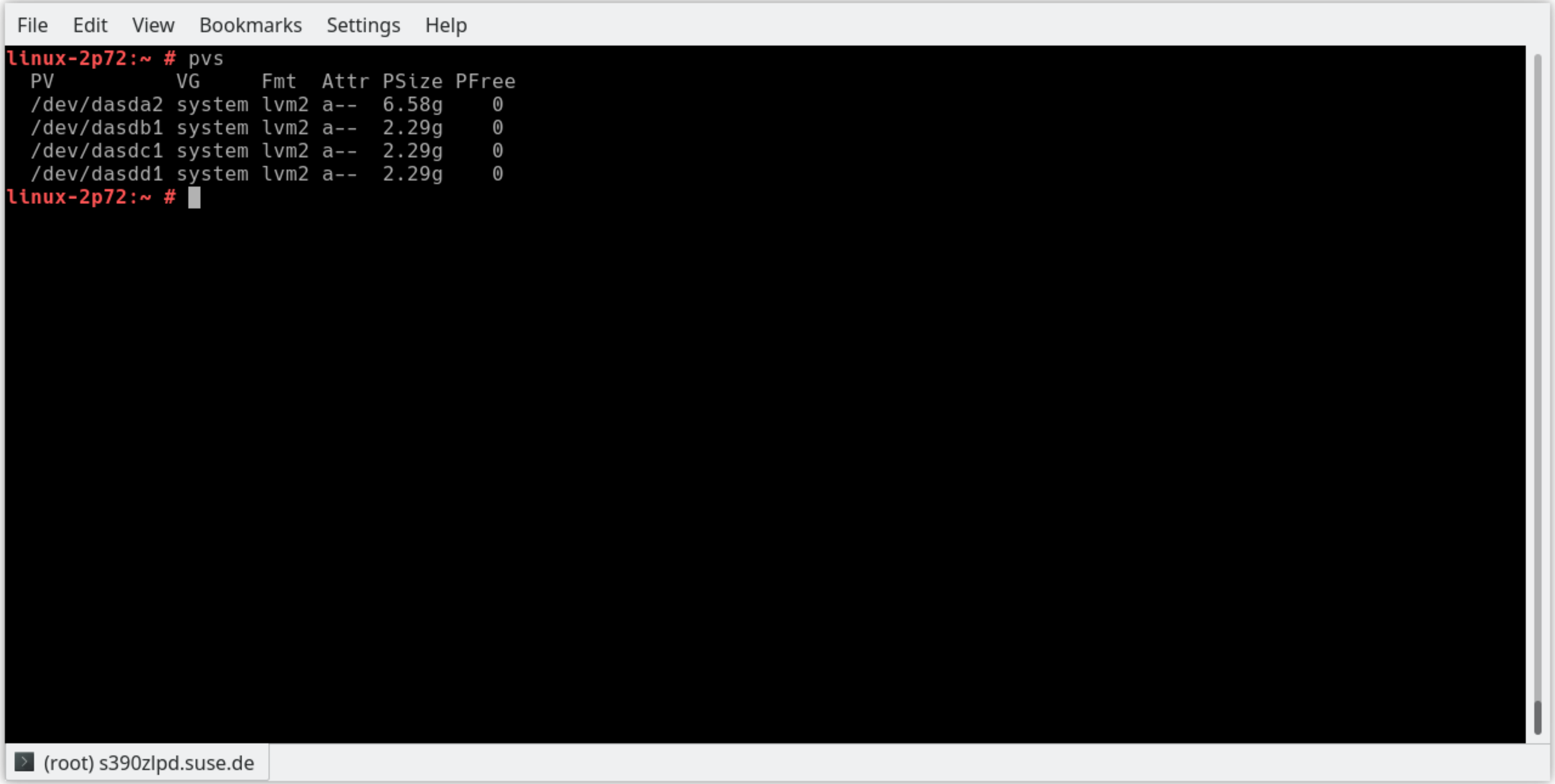
```
linux-2p72:~ # panel.exe --mktype=AES --mk-set
Preparing to SET master key

SET for Master key [AES-MK  ] returned:

        Return Code [0] Reason Code [0]

linux-2p72:~ # 
```

Verify CURRENT master key is valid



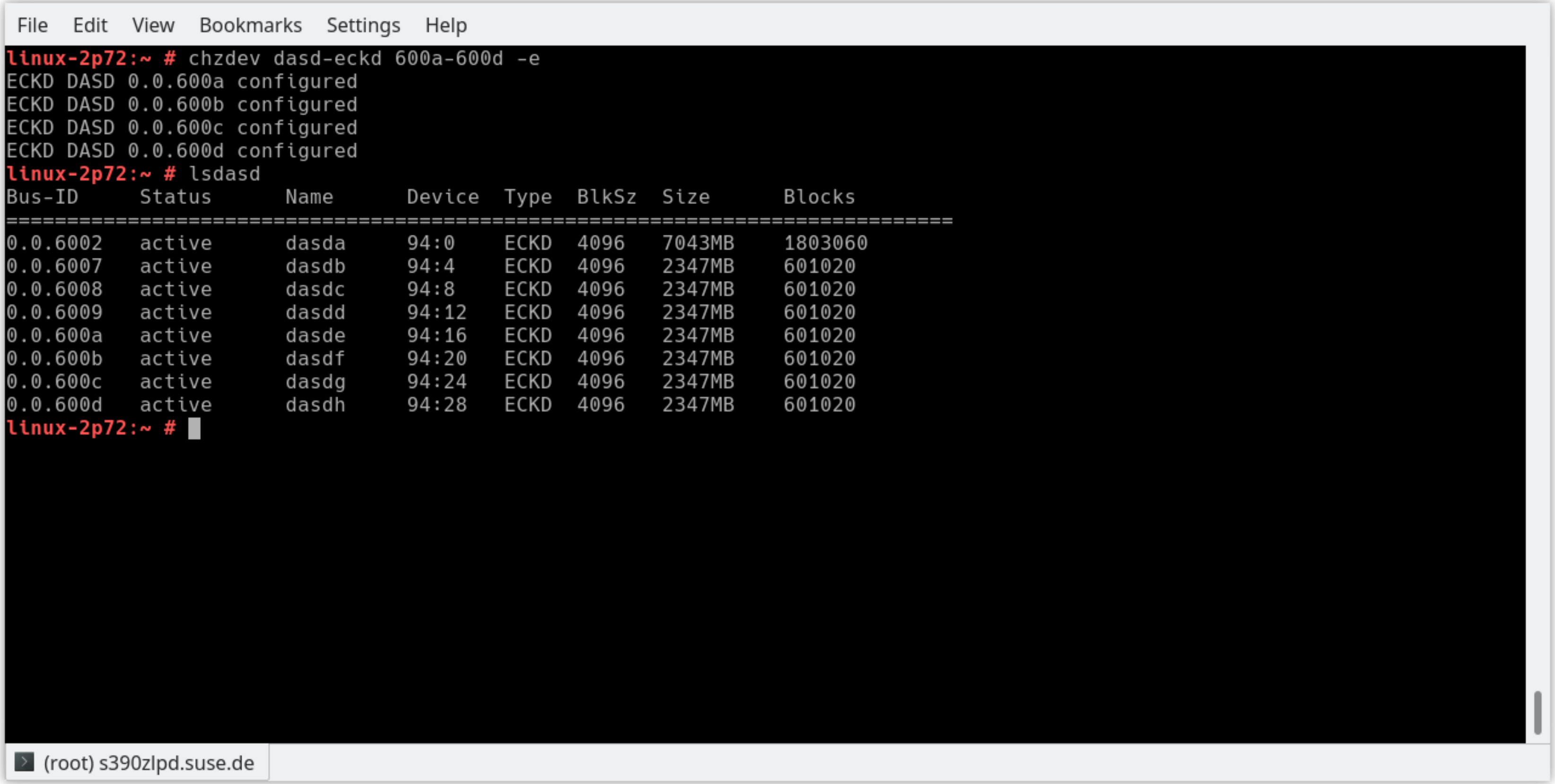
```
linux-2p72:~ # panel.exe --mktype=AES --mkregister=CURRENT --mk-query
Preparing to QUERY master key verification pattern

Query of Key Verification Pattern for Master key [AES-MK  ] [KEY-KM  ] returned:

RND[0000000000000000]
VER[1B92ACA085782622]
linux-2p72:~ #
```

The operating system volume group



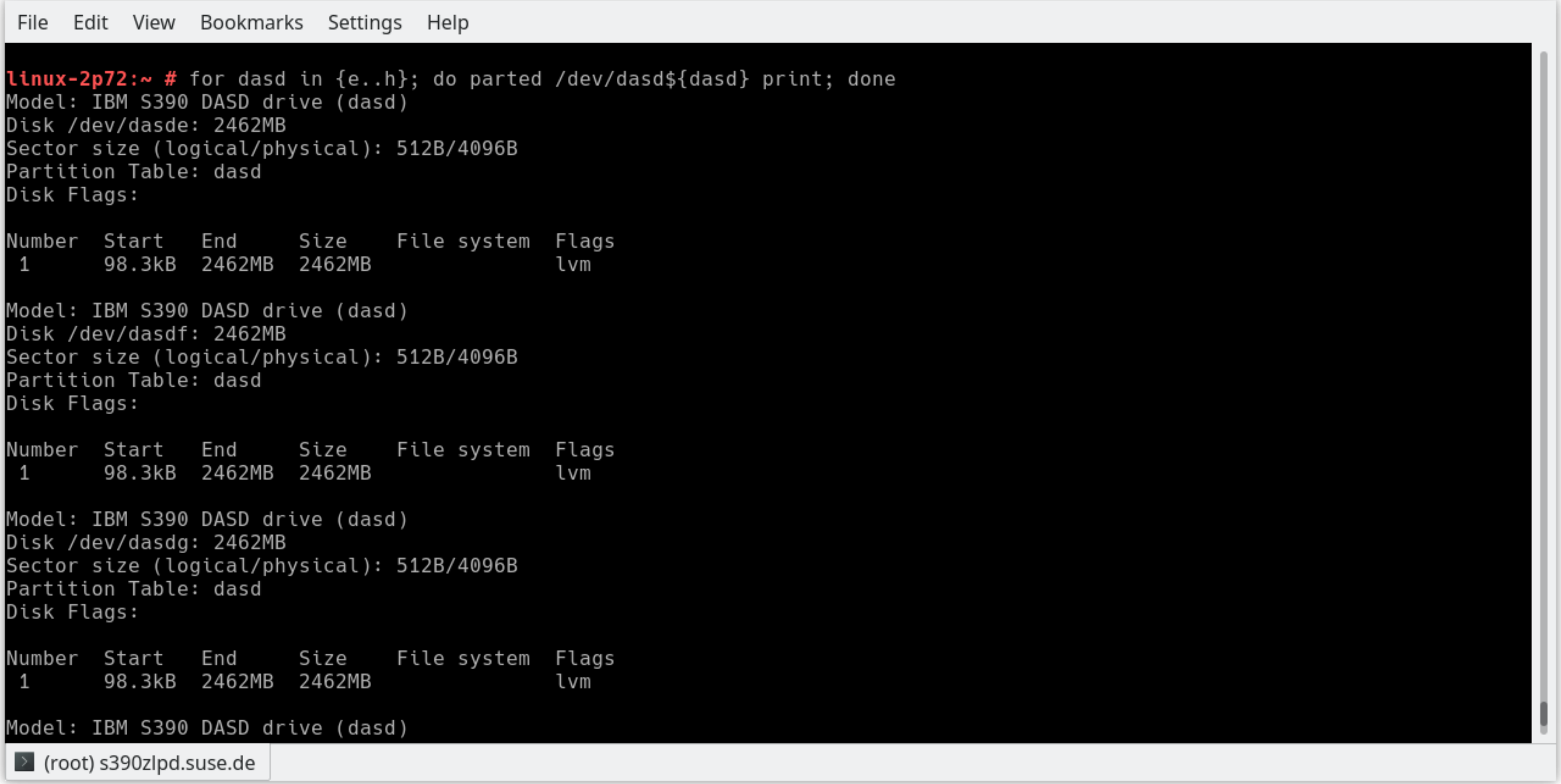
```
linux-2p72:~ # pvs
  PV          VG     Fmt  Attr PSize PFree
  /dev/dasda2 system lvm2 a--  6.58g    0
  /dev/dasdb1 system lvm2 a--  2.29g    0
  /dev/dasdc1 system lvm2 a--  2.29g    0
  /dev/dasdd1 system lvm2 a--  2.29g    0
linux-2p72:~ #
```

Enabling the DASDs for pervasive encryption



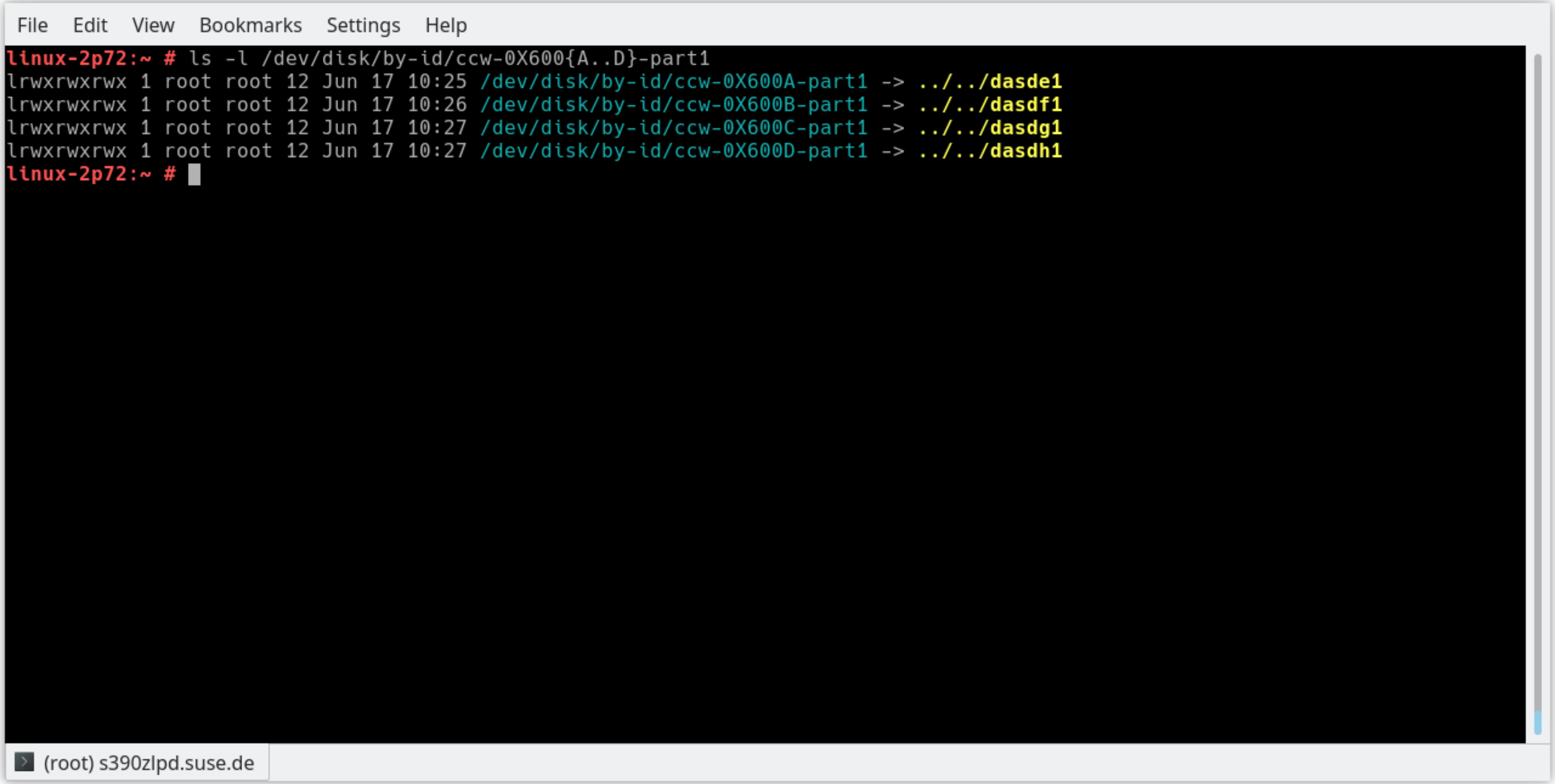
```
linux-2p72:~ # chzdev dasd-eckd 600a-600d -e
ECKD DASD 0.0.600a configured
ECKD DASD 0.0.600b configured
ECKD DASD 0.0.600c configured
ECKD DASD 0.0.600d configured
linux-2p72:~ # lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.6002   active      dasda     94:0    ECKD  4096   7043MB    1803060
0.0.6007   active      dasdb     94:4    ECKD  4096   2347MB    601020
0.0.6008   active      dasdc     94:8    ECKD  4096   2347MB    601020
0.0.6009   active      dasdd     94:12   ECKD  4096   2347MB    601020
0.0.600a   active      dasde     94:16   ECKD  4096   2347MB    601020
0.0.600b   active      dasdf     94:20   ECKD  4096   2347MB    601020
0.0.600c   active      dasdg     94:24   ECKD  4096   2347MB    601020
0.0.600d   active      dasdh     94:28   ECKD  4096   2347MB    601020
linux-2p72:~ # 
```

Format and partition the DASDs



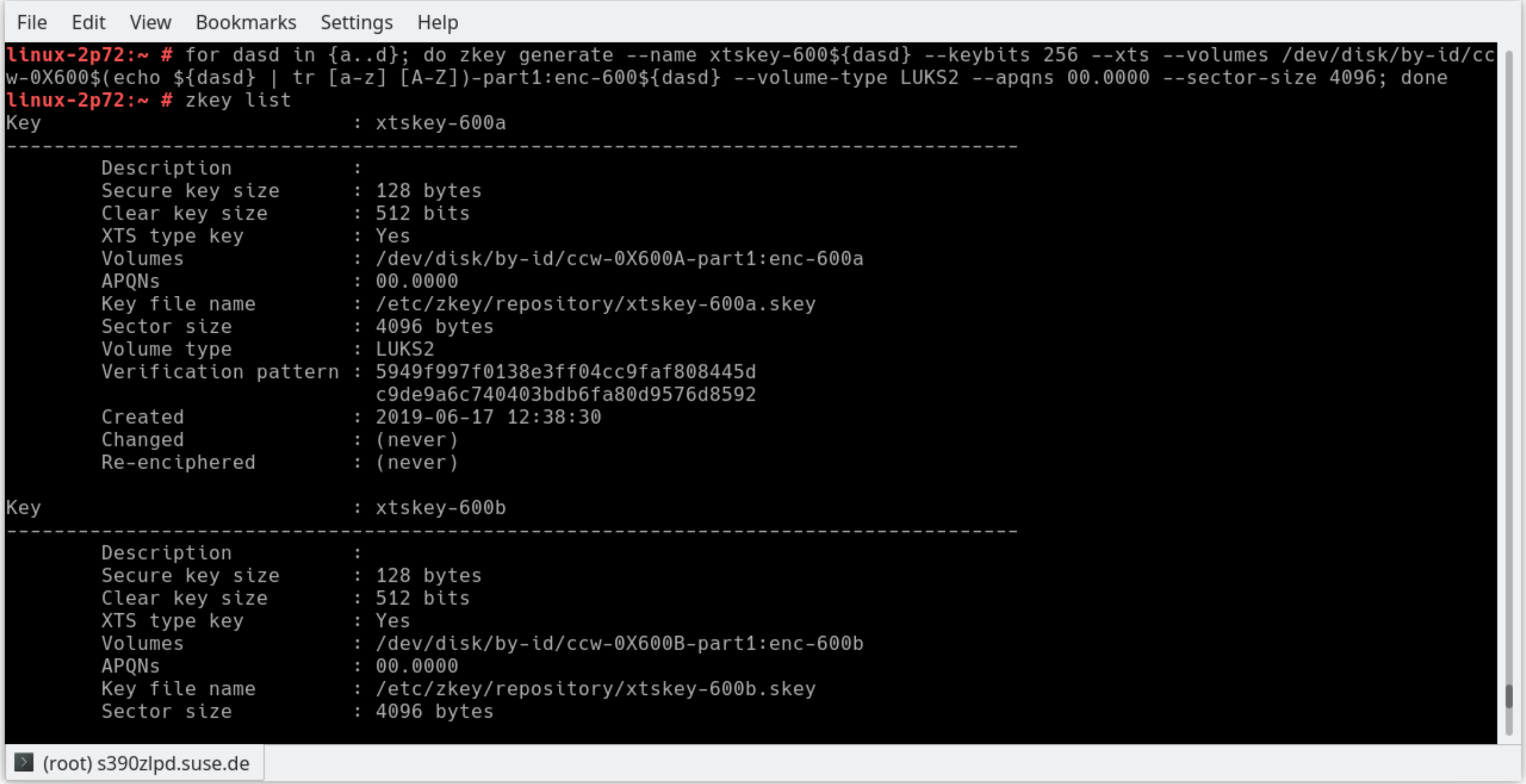
```
linux-2p72:~ # for dasd in {e..h}; do parted /dev/dasd${dasd} print; done
Model: IBM S390 DASD drive (dasd)
Disk /dev/dasde: 2462MB
Sector size (logical/physical): 512B/4096B
Partition Table: dasd
Disk Flags:

Number  Start   End     Size    File system  Flags
 1      98.3kB  2462MB  2462MB               lvm

Model: IBM S390 DASD drive (dasd)
Disk /dev/dasdf: 2462MB
Sector size (logical/physical): 512B/4096B
Partition Table: dasd
Disk Flags:

Number  Start   End     Size    File system  Flags
 1      98.3kB  2462MB  2462MB               lvm

Model: IBM S390 DASD drive (dasd)
Disk /dev/dasdg: 2462MB
Sector size (logical/physical): 512B/4096B
Partition Table: dasd
Disk Flags:

Number  Start   End     Size    File system  Flags
 1      98.3kB  2462MB  2462MB               lvm

Model: IBM S390 DASD drive (dasd)
```

Using /dev/disk/by-id to refer to the partitions



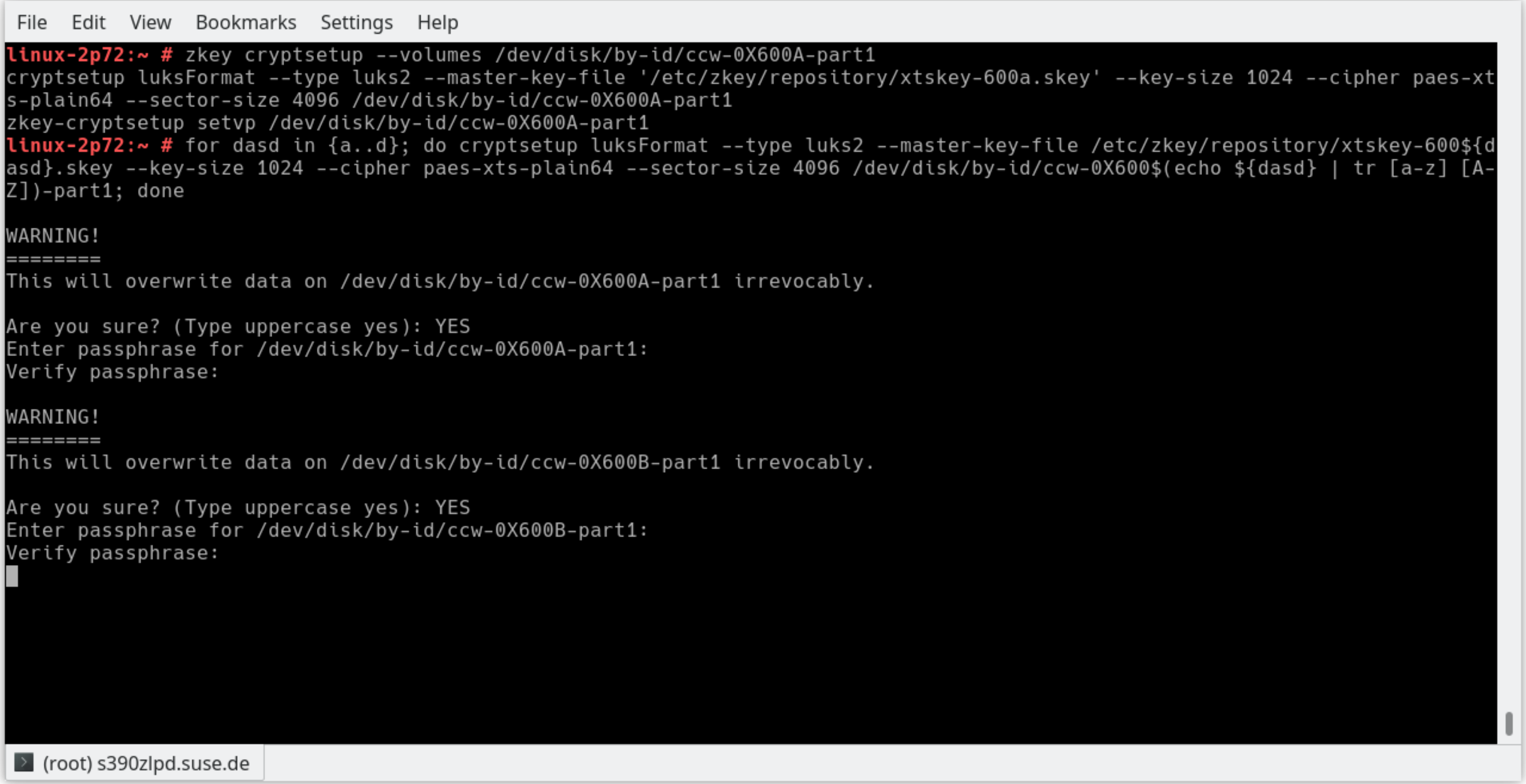
```
linux-2p72:~ # ls -l /dev/disk/by-id/ccw-0X600{A..D}-part1
lrwxrwxrwx 1 root root 12 Jun 17 10:25 /dev/disk/by-id/ccw-0X600A-part1 -> ../../dasde1
lrwxrwxrwx 1 root root 12 Jun 17 10:26 /dev/disk/by-id/ccw-0X600B-part1 -> ../../dasdf1
lrwxrwxrwx 1 root root 12 Jun 17 10:27 /dev/disk/by-id/ccw-0X600C-part1 -> ../../dasdg1
lrwxrwxrwx 1 root root 12 Jun 17 10:27 /dev/disk/by-id/ccw-0X600D-part1 -> ../../dasdh1
linux-2p72:~ # 
```

Generate a secure key for each partition



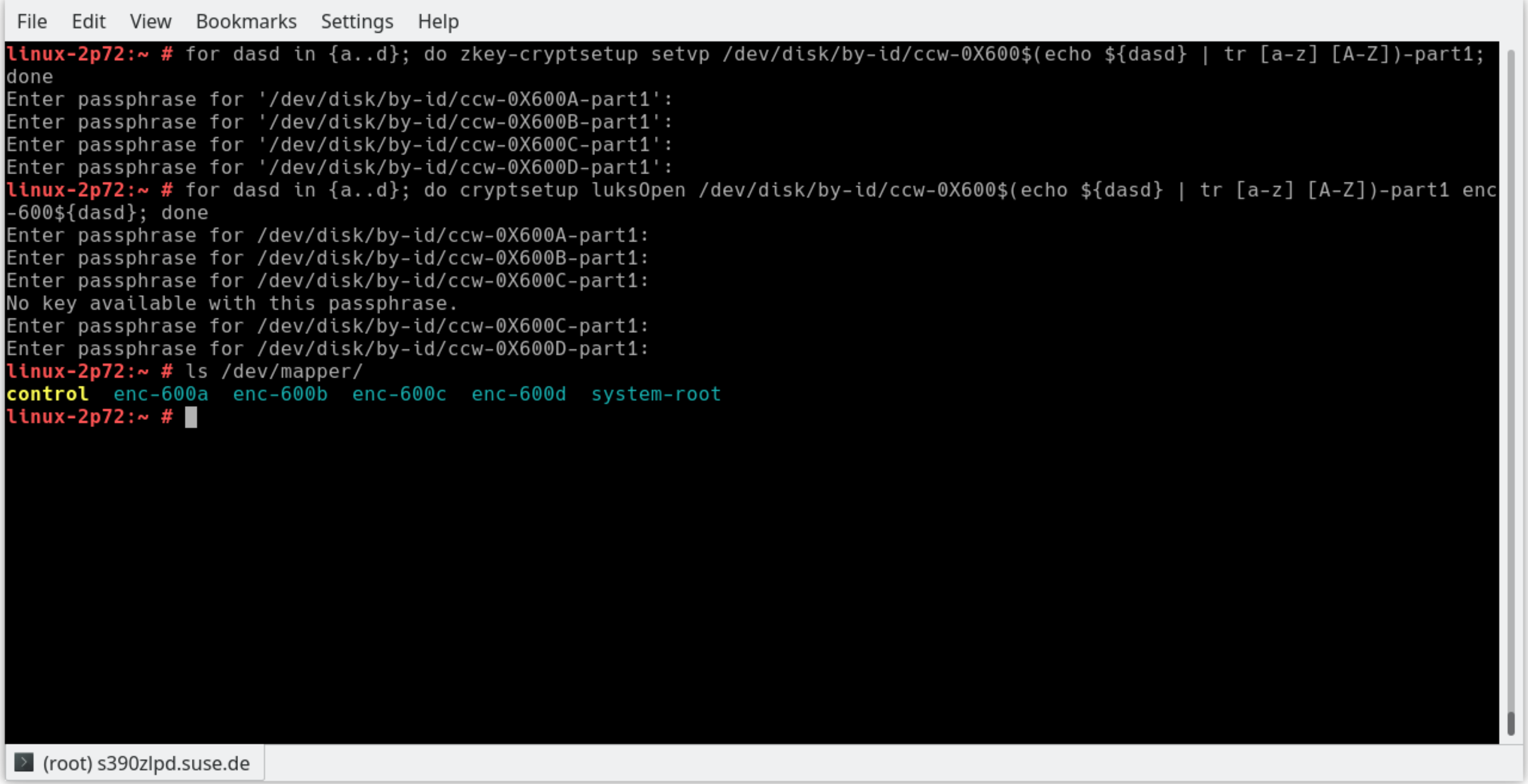
```
linux-2p72:~ # for dasd in {a..d}; do zkey generate --name xtskey-600${dasd} --keybits 256 --xts --volumes /dev/disk/by-id/cc
w-0X600$(echo ${dasd} | tr [a-z] [A-Z])-part1:enc-600${dasd} --volume-type LUKS2 --apqns 00.0000 --sector-size 4096; done
linux-2p72:~ # zkey list
Key                           : xtskey-600a
-------------------------------------------------------------------------------------
        Description           :
        Secure key size       : 128 bytes
        Clear key size        : 512 bits
        XTS type key          : Yes
        Volumes               : /dev/disk/by-id/ccw-0X600A-part1:enc-600a
        APQNs                 : 00.0000
        Key file name         : /etc/zkey/repository/xtskey-600a.skey
        Sector size           : 4096 bytes
        Volume type           : LUKS2
        Verification pattern : 5949f997f0138e3ff04cc9faf808445d
                                c9de9a6c740403bdb6fa80d9576d8592
        Created               : 2019-06-17 12:38:30
        Changed               : (never)
        Re-enciphered         : (never)

Key                           : xtskey-600b
-------------------------------------------------------------------------------------
        Description           :
        Secure key size       : 128 bytes
        Clear key size        : 512 bits
        XTS type key          : Yes
        Volumes               : /dev/disk/by-id/ccw-0X600B-part1:enc-600b
        APQNs                 : 00.0000
        Key file name         : /etc/zkey/repository/xtskey-600b.skey
        Sector size           : 4096 bytes
```

Use zkey to generate cryptsetup commands that will be run for each partition



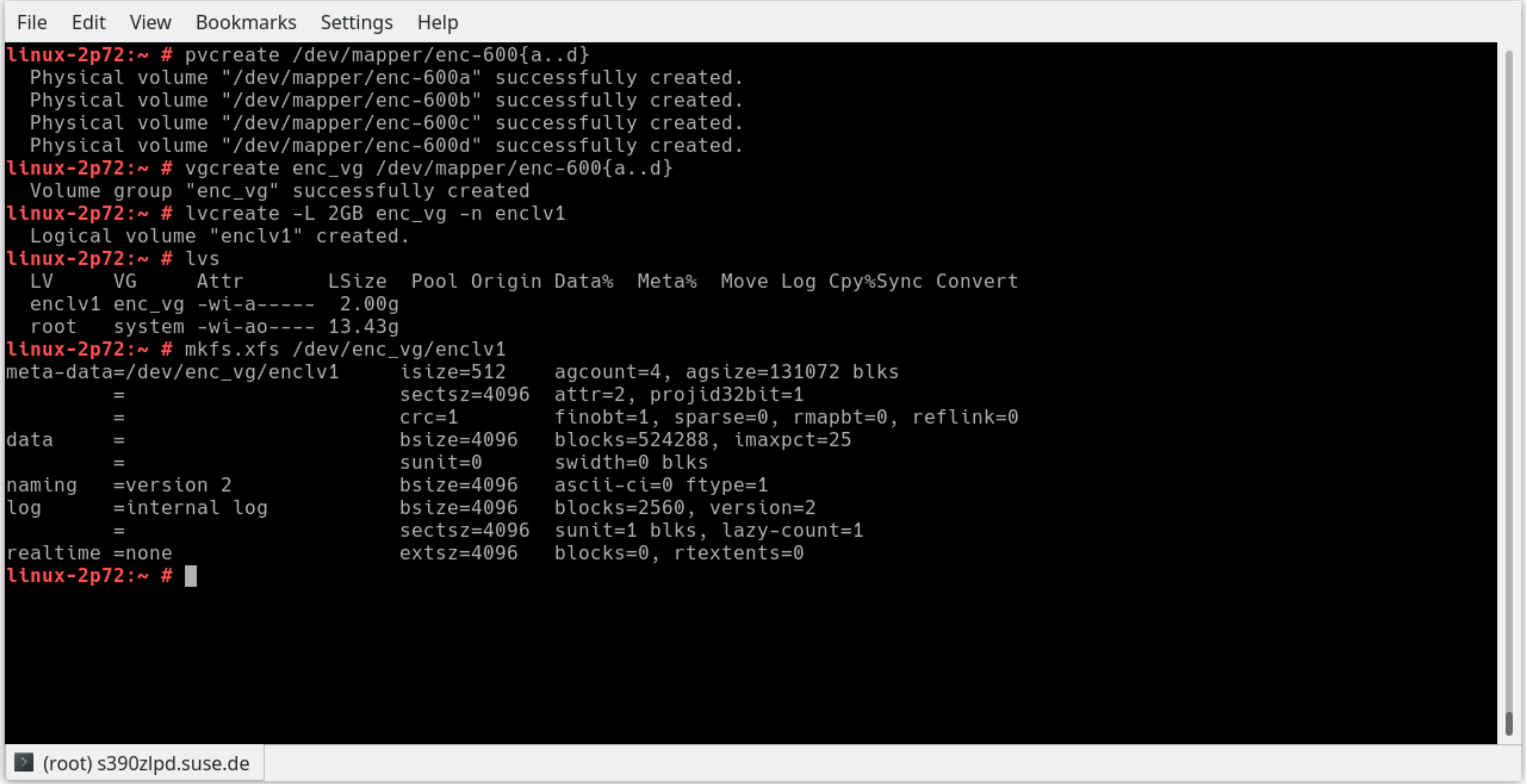
```
linux-2p72:~ # zkey cryptsetup --volumes /dev/disk/by-id/ccw-0X600A-part1
cryptsetup luksFormat --type luks2 --master-key-file '/etc/zkey/repository/xtskey-600a.skey' --key-size 1024 --cipher paes-xt
s-plain64 --sector-size 4096 /dev/disk/by-id/ccw-0X600A-part1
zkey-cryptsetup setvp /dev/disk/by-id/ccw-0X600A-part1
linux-2p72:~ # for dasd in {a..d}; do cryptsetup luksFormat --type luks2 --master-key-file /etc/zkey/repository/xtskey-600${d
asd}.skey --key-size 1024 --cipher paes-xts-plain64 --sector-size 4096 /dev/disk/by-id/ccw-0X600$(echo ${dasd} | tr [a-z] [A-
Z])-part1; done

WARNING!
========
This will overwrite data on /dev/disk/by-id/ccw-0X600A-part1 irrevocably.

Are you sure? (Type uppercase yes): YES
Enter passphrase for /dev/disk/by-id/ccw-0X600A-part1:
Verify passphrase:

WARNING!
========
This will overwrite data on /dev/disk/by-id/ccw-0X600B-part1 irrevocably.

Are you sure? (Type uppercase yes): YES
Enter passphrase for /dev/disk/by-id/ccw-0X600B-part1:
Verify passphrase:
```

Set the LUKS2 verification pattern and open each encrypted partition



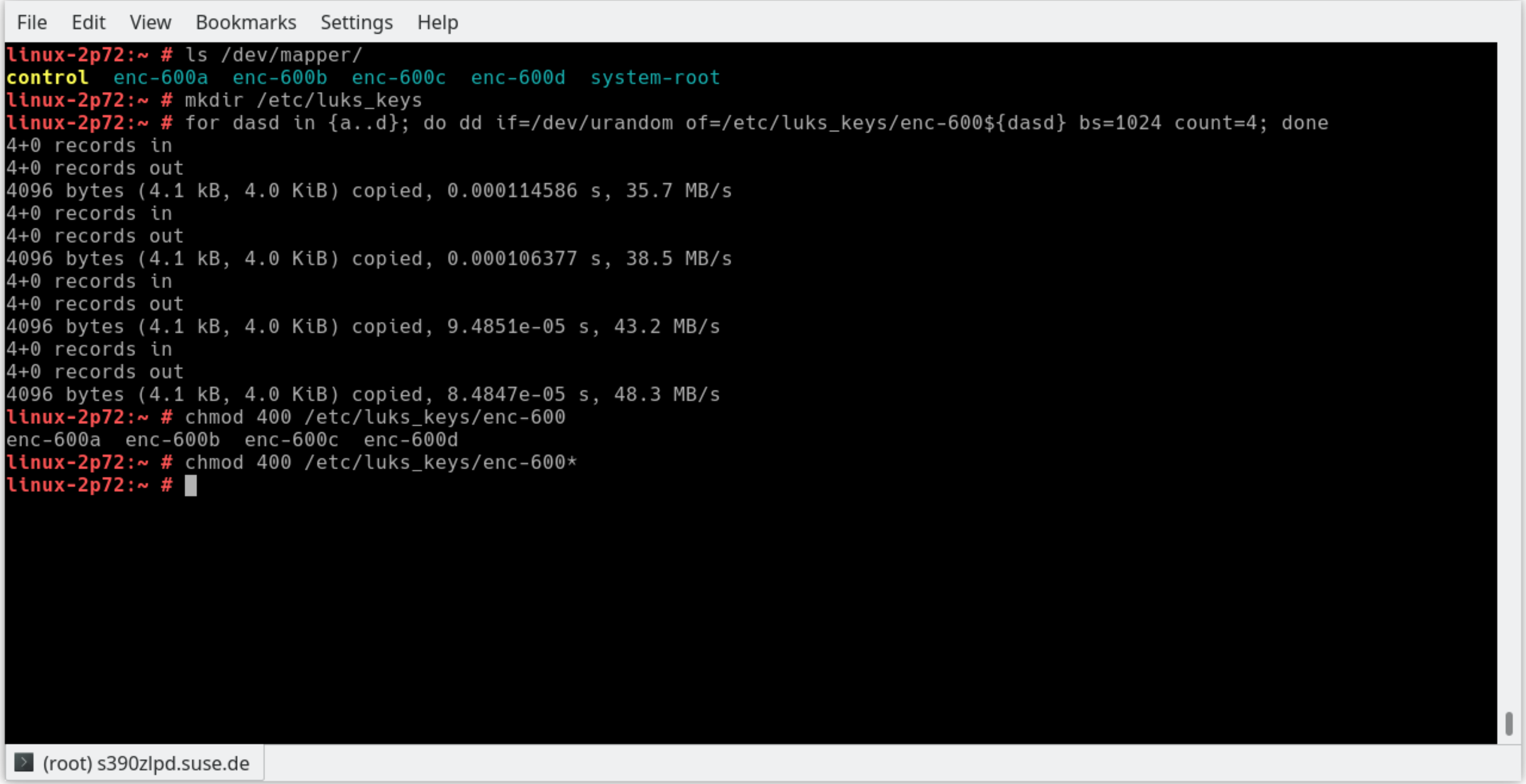
```
linux-2p72:~ # for dasd in {a..d}; do zkey-cryptsetup setvp /dev/disk/by-id/ccw-0X600$(echo ${dasd} | tr [a-z] [A-Z])-part1;
done
Enter passphrase for '/dev/disk/by-id/ccw-0X600A-part1':
Enter passphrase for '/dev/disk/by-id/ccw-0X600B-part1':
Enter passphrase for '/dev/disk/by-id/ccw-0X600C-part1':
Enter passphrase for '/dev/disk/by-id/ccw-0X600D-part1':
linux-2p72:~ # for dasd in {a..d}; do cryptsetup luksOpen /dev/disk/by-id/ccw-0X600$(echo ${dasd} | tr [a-z] [A-Z])-part1 enc
-600${dasd}; done
Enter passphrase for /dev/disk/by-id/ccw-0X600A-part1:
Enter passphrase for /dev/disk/by-id/ccw-0X600B-part1:
Enter passphrase for /dev/disk/by-id/ccw-0X600C-part1:
No key available with this passphrase.
Enter passphrase for /dev/disk/by-id/ccw-0X600C-part1:
Enter passphrase for /dev/disk/by-id/ccw-0X600D-part1:
linux-2p72:~ # ls /dev/mapper/
control  enc-600a  enc-600b  enc-600c  enc-600d  system-root
linux-2p72:~ #
```

Create LVM logical volume and format with XFS



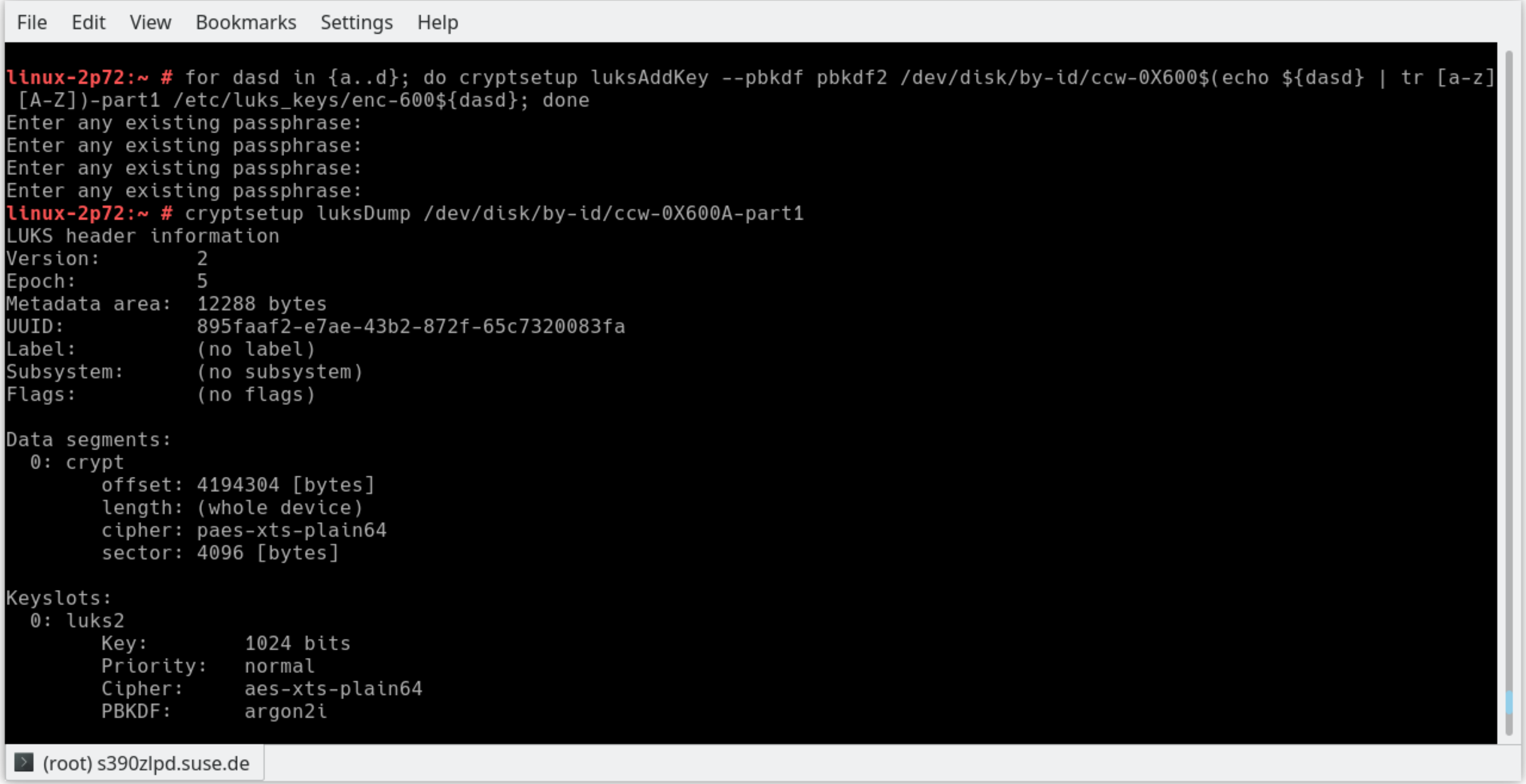
```
linux-2p72:~ # pvcreate /dev/mapper/enc-600{a..d}
  Physical volume "/dev/mapper/enc-600a" successfully created.
  Physical volume "/dev/mapper/enc-600b" successfully created.
  Physical volume "/dev/mapper/enc-600c" successfully created.
  Physical volume "/dev/mapper/enc-600d" successfully created.
linux-2p72:~ # vgcreate enc_vg /dev/mapper/enc-600{a..d}
  Volume group "enc_vg" successfully created
linux-2p72:~ # lvcreate -L 2GB enc_vg -n enclv1
  Logical volume "enclv1" created.
linux-2p72:~ # lvs
  LV     VG     Attr       LSize  Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  enclv1 enc_vg -wi-a-----  2.00g
  root   system -wi-ao---- 13.43g
linux-2p72:~ # mkfs.xfs /dev/enc_vg/enclv1
meta-data=/dev/enc_vg/enclv1     isize=512    agcount=4, agsize=131072 blks
         =                       sectsz=4096  attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=0, rmapbt=0, reflink=0
data     =                       bsize=4096   blocks=524288, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=4096  sunit=1 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
linux-2p72:~ #
```

Create a key file to automatically open encrypted partition without a passphrase



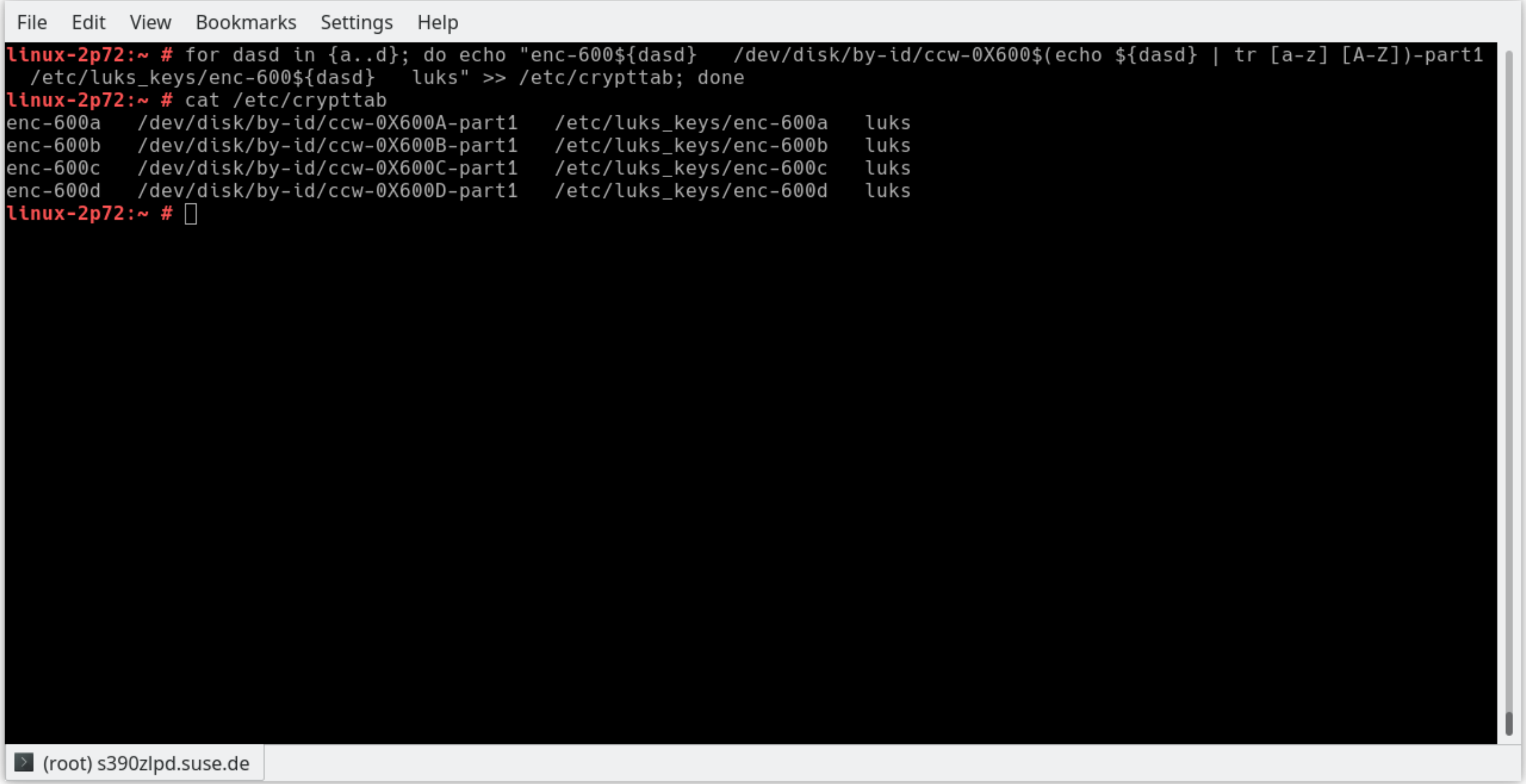
```
linux-2p72:~ # ls /dev/mapper/
control  enc-600a  enc-600b  enc-600c  enc-600d  system-root
linux-2p72:~ # mkdir /etc/luks_keys
linux-2p72:~ # for dasd in {a..d}; do dd if=/dev/urandom of=/etc/luks_keys/enc-600${dasd} bs=1024 count=4; done
4+0 records in
4+0 records out
4096 bytes (4.1 kB, 4.0 KiB) copied, 0.000114586 s, 35.7 MB/s
4+0 records in
4+0 records out
4096 bytes (4.1 kB, 4.0 KiB) copied, 0.000106377 s, 38.5 MB/s
4+0 records in
4+0 records out
4096 bytes (4.1 kB, 4.0 KiB) copied, 9.4851e-05 s, 43.2 MB/s
4+0 records in
4+0 records out
4096 bytes (4.1 kB, 4.0 KiB) copied, 8.4847e-05 s, 48.3 MB/s
linux-2p72:~ # chmod 400 /etc/luks_keys/enc-600
enc-600a  enc-600b  enc-600c  enc-600d
linux-2p72:~ # chmod 400 /etc/luks_keys/enc-600*
linux-2p72:~ #
```

Add the key file to the LUKS2 header



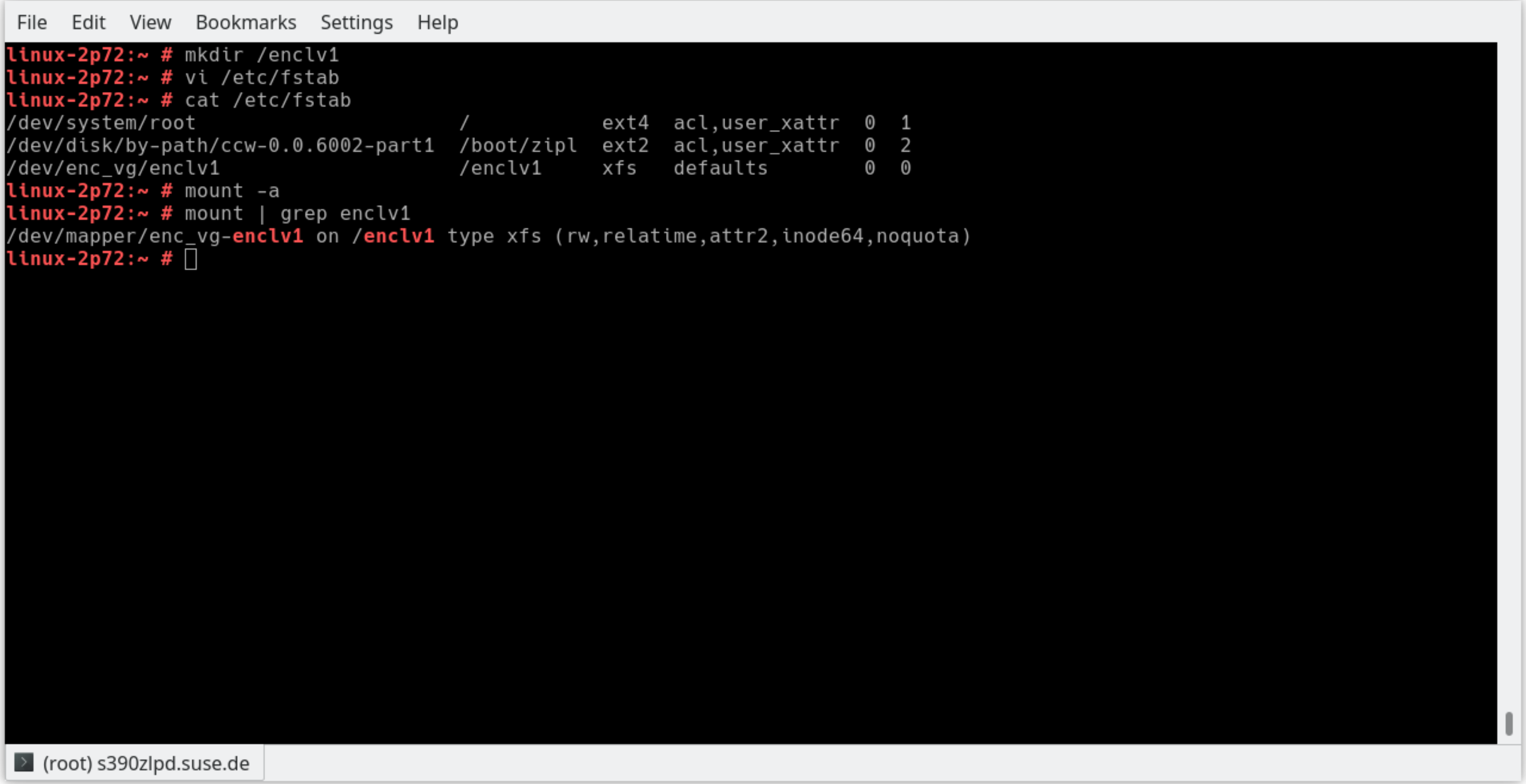
```
linux-2p72:~ # for dasd in {a..d}; do cryptsetup luksAddKey --pbkdf pbkdf2 /dev/disk/by-id/ccw-0X600$(echo ${dasd} | tr [a-z]
 [A-Z])-part1 /etc/luks_keys/enc-600${dasd}; done
Enter any existing passphrase:
Enter any existing passphrase:
Enter any existing passphrase:
Enter any existing passphrase:
linux-2p72:~ # cryptsetup luksDump /dev/disk/by-id/ccw-0X600A-part1
LUKS header information
Version:        2
Epoch:          5
Metadata area:  12288 bytes
UUID:           895faaf2-e7ae-43b2-872f-65c7320083fa
Label:          (no label)
Subsystem:      (no subsystem)
Flags:          (no flags)

Data segments:
  0: crypt
        offset: 4194304 [bytes]
        length: (whole device)
        cipher: paes-xts-plain64
        sector: 4096 [bytes]

Keyslots:
  0: luks2
        Key:        1024 bits
        Priority:   normal
        Cipher:     aes-xts-plain64
        PBKDF:      argon2i
```

A crypttab file is needed to auto open encrypted partitions



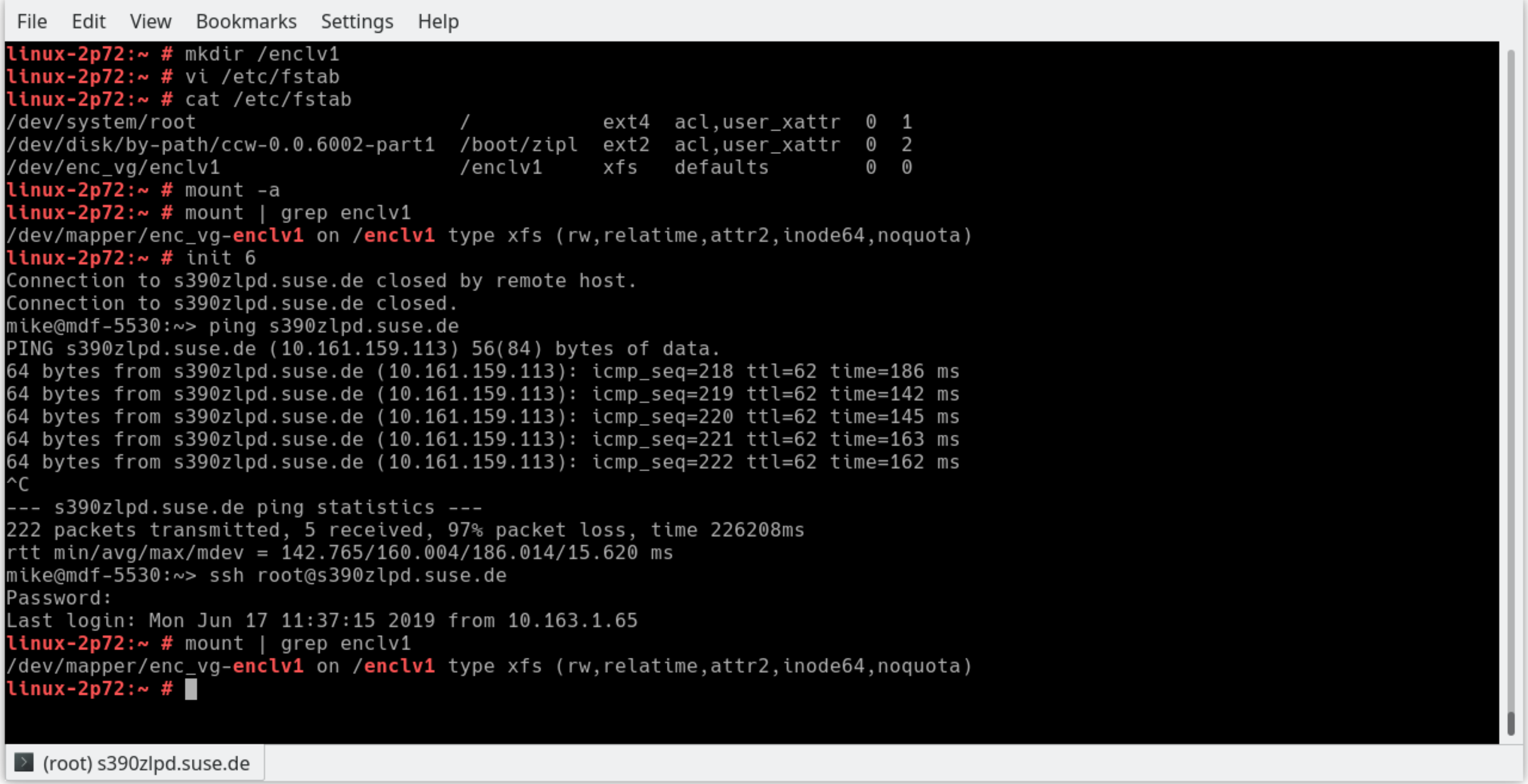
```
linux-2p72:~ # for dasd in {a..d}; do echo "enc-600${dasd}   /dev/disk/by-id/ccw-0X600$(echo ${dasd} | tr [a-z] [A-Z])-part1
  /etc/luks_keys/enc-600${dasd}   luks" >> /etc/crypttab; done
linux-2p72:~ # cat /etc/crypttab
enc-600a   /dev/disk/by-id/ccw-0X600A-part1   /etc/luks_keys/enc-600a   luks
enc-600b   /dev/disk/by-id/ccw-0X600B-part1   /etc/luks_keys/enc-600b   luks
enc-600c   /dev/disk/by-id/ccw-0X600C-part1   /etc/luks_keys/enc-600c   luks
enc-600d   /dev/disk/by-id/ccw-0X600D-part1   /etc/luks_keys/enc-600d   luks
linux-2p72:~ # 
```

Mount the formatted logical volume



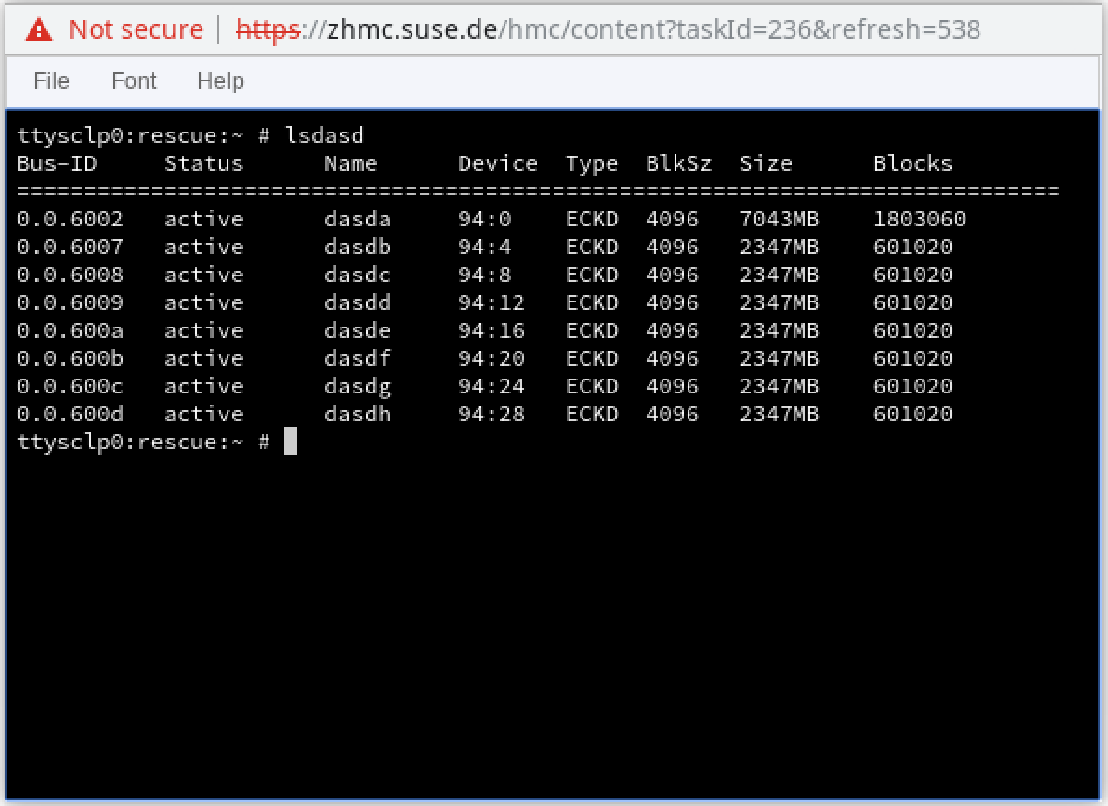
```
linux-2p72:~ # mkdir /enclv1
linux-2p72:~ # vi /etc/fstab
linux-2p72:~ # cat /etc/fstab
/dev/system/root                    /           ext4  acl,user_xattr  0  1
/dev/disk/by-path/ccw-0.0.6002-part1  /boot/zipl  ext2  acl,user_xattr  0  2
/dev/enc_vg/enclv1                  /enclv1     xfs   defaults        0  0
linux-2p72:~ # mount -a
linux-2p72:~ # mount | grep enclv1
/dev/mapper/enc_vg-enclv1 on /enclv1 type xfs (rw,relatime,attr2,inode64,noquota)
linux-2p72:~ #
```

Reboot to verify encrypted partitions are opened and logical volume mounted



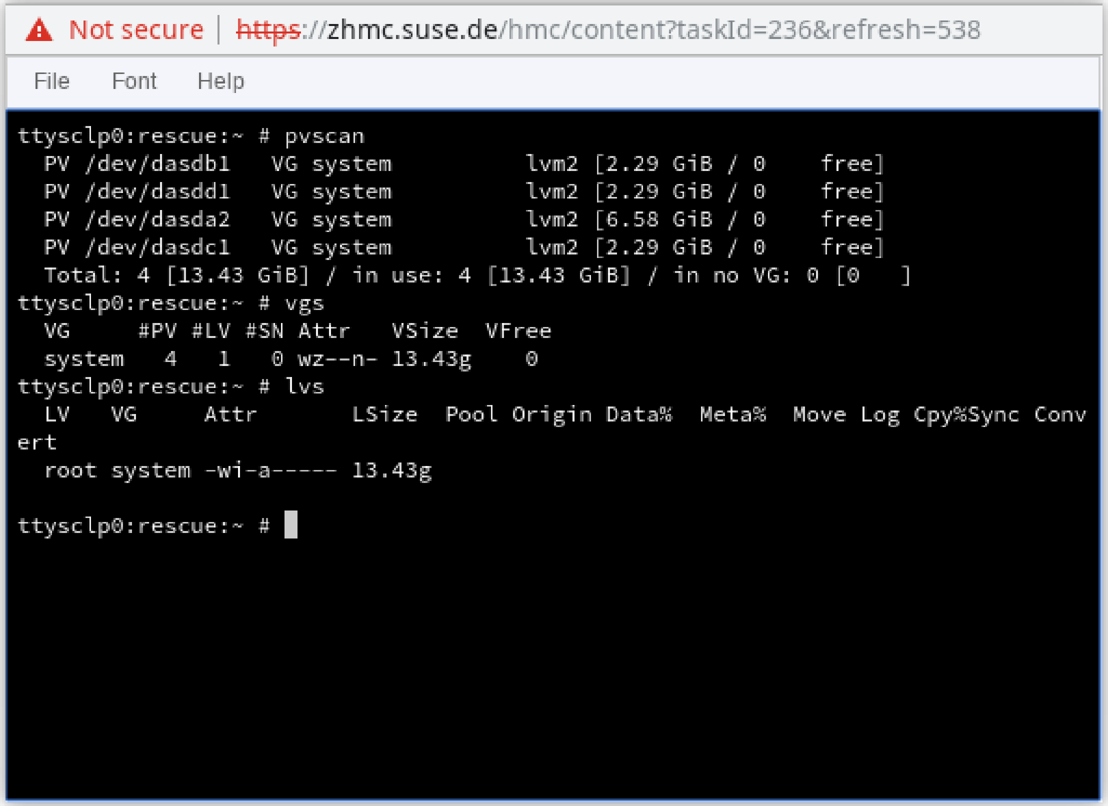
```
linux-2p72:~ # mkdir /enclv1
linux-2p72:~ # vi /etc/fstab
linux-2p72:~ # cat /etc/fstab
/dev/system/root                      /           ext4  acl,user_xattr  0  1
/dev/disk/by-path/ccw-0.0.6002-part1  /boot/zipl  ext2  acl,user_xattr  0  2
/dev/enc_vg/enclv1                    /enclv1     xfs   defaults        0  0
linux-2p72:~ # mount -a
linux-2p72:~ # mount | grep enclv1
/dev/mapper/enc_vg-enclv1 on /enclv1 type xfs (rw,relatime,attr2,inode64,noquota)
linux-2p72:~ # init 6
Connection to s390zlpd.suse.de closed by remote host.
Connection to s390zlpd.suse.de closed.
mike@mdf-5530:~> ping s390zlpd.suse.de
PING s390zlpd.suse.de (10.161.159.113) 56(84) bytes of data.
64 bytes from s390zlpd.suse.de (10.161.159.113): icmp_seq=218 ttl=62 time=186 ms
64 bytes from s390zlpd.suse.de (10.161.159.113): icmp_seq=219 ttl=62 time=142 ms
64 bytes from s390zlpd.suse.de (10.161.159.113): icmp_seq=220 ttl=62 time=145 ms
64 bytes from s390zlpd.suse.de (10.161.159.113): icmp_seq=221 ttl=62 time=163 ms
64 bytes from s390zlpd.suse.de (10.161.159.113): icmp_seq=222 ttl=62 time=162 ms
^C
--- s390zlpd.suse.de ping statistics ---
222 packets transmitted, 5 received, 97% packet loss, time 226208ms
rtt min/avg/max/mdev = 142.765/160.004/186.014/15.620 ms
mike@mdf-5530:~> ssh root@s390zlpd.suse.de
Password:
Last login: Mon Jun 17 11:37:15 2019 from 10.163.1.65
linux-2p72:~ # mount | grep enclv1
/dev/mapper/enc_vg-enclv1 on /enclv1 type xfs (rw,relatime,attr2,inode64,noquota)
linux-2p72:~ #
```

SLES rescue system with OS and encrypted DASDs enabled



```
ttysclp0:rescue:~ # lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.6002   active      dasda     94:0    ECKD  4096   7043MB    1803060
0.0.6007   active      dasdb     94:4    ECKD  4096   2347MB    601020
0.0.6008   active      dasdc     94:8    ECKD  4096   2347MB    601020
0.0.6009   active      dasdd     94:12   ECKD  4096   2347MB    601020
0.0.600a   active      dasde     94:16   ECKD  4096   2347MB    601020
0.0.600b   active      dasdf     94:20   ECKD  4096   2347MB    601020
0.0.600c   active      dasdg     94:24   ECKD  4096   2347MB    601020
0.0.600d   active      dasdh     94:28   ECKD  4096   2347MB    601020
ttysclp0:rescue:~ #
```

Operating system volume group is available but encrypted is NOT!

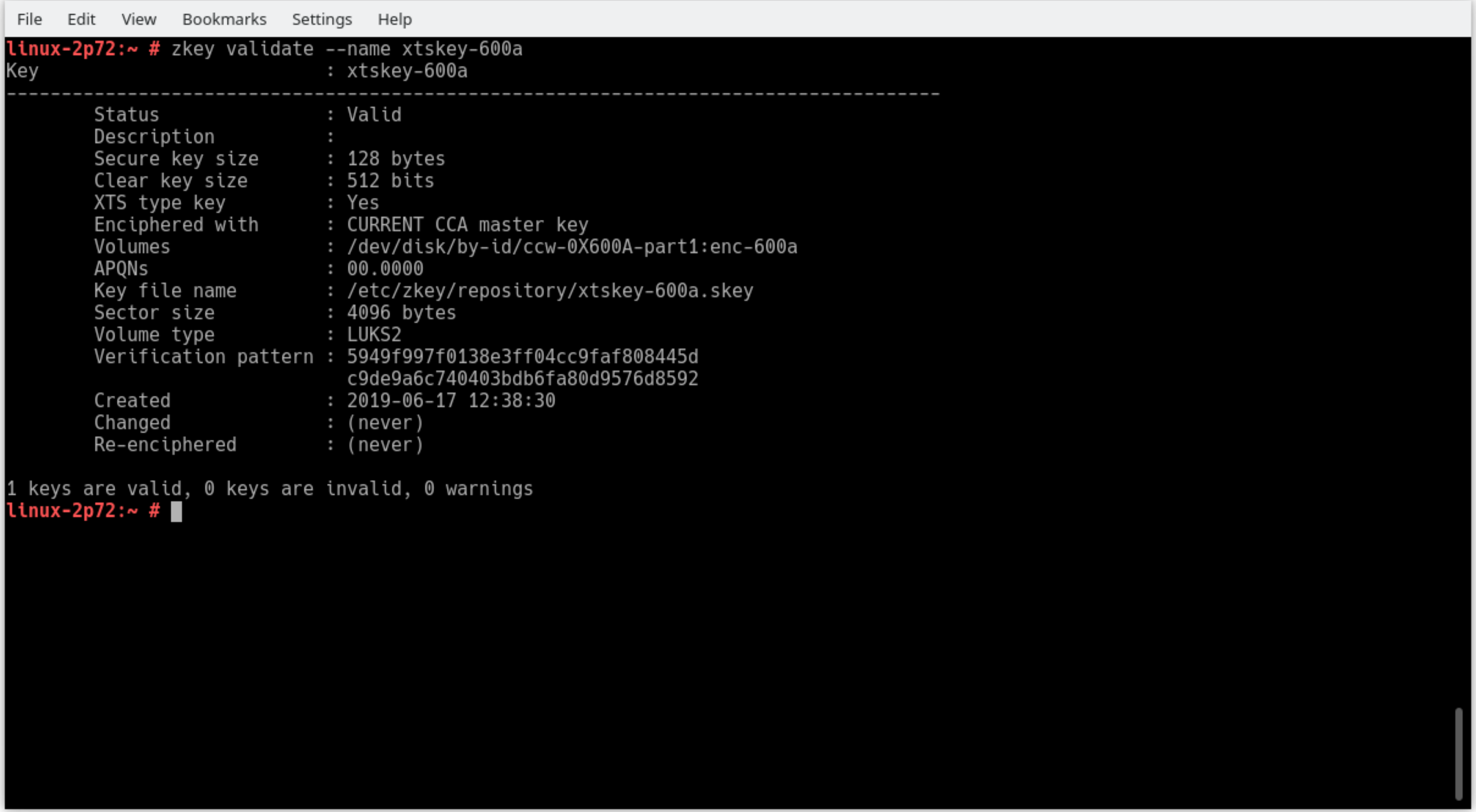


```
ttysclp0:rescue:~ # pvscan
  PV /dev/dasdb1   VG system          lvm2 [2.29 GiB / 0    free]
  PV /dev/dasdd1   VG system          lvm2 [2.29 GiB / 0    free]
  PV /dev/dasda2   VG system          lvm2 [6.58 GiB / 0    free]
  PV /dev/dasdc1   VG system          lvm2 [2.29 GiB / 0    free]
  Total: 4 [13.43 GiB] / in use: 4 [13.43 GiB] / in no VG: 0 [0   ]
ttysclp0:rescue:~ # vgs
  VG     #PV #LV #SN Attr   VSize  VFree
  system   4   1   0 wz--n- 13.43g    0
ttysclp0:rescue:~ # lvs
  LV   VG     Attr       LSize  Pool Origin Data%  Meta%  Move Log Cpy%Sync Conv
ert
  root system -wi-a----- 13.43g

ttysclp0:rescue:~ #
```

Working with data-at-rest encryption

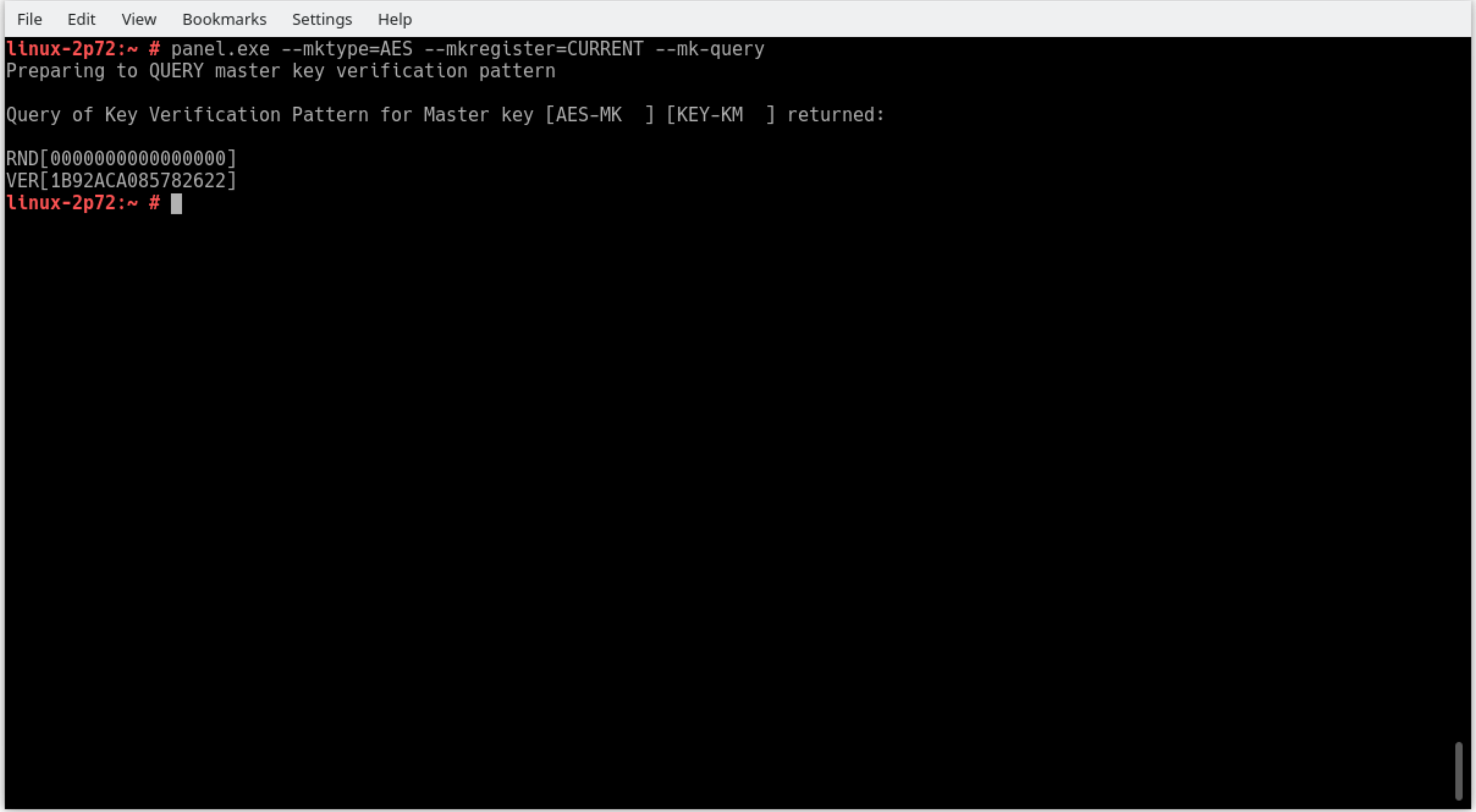
Re-encipher secure keys

Secure key enciphered with CURRENT master key



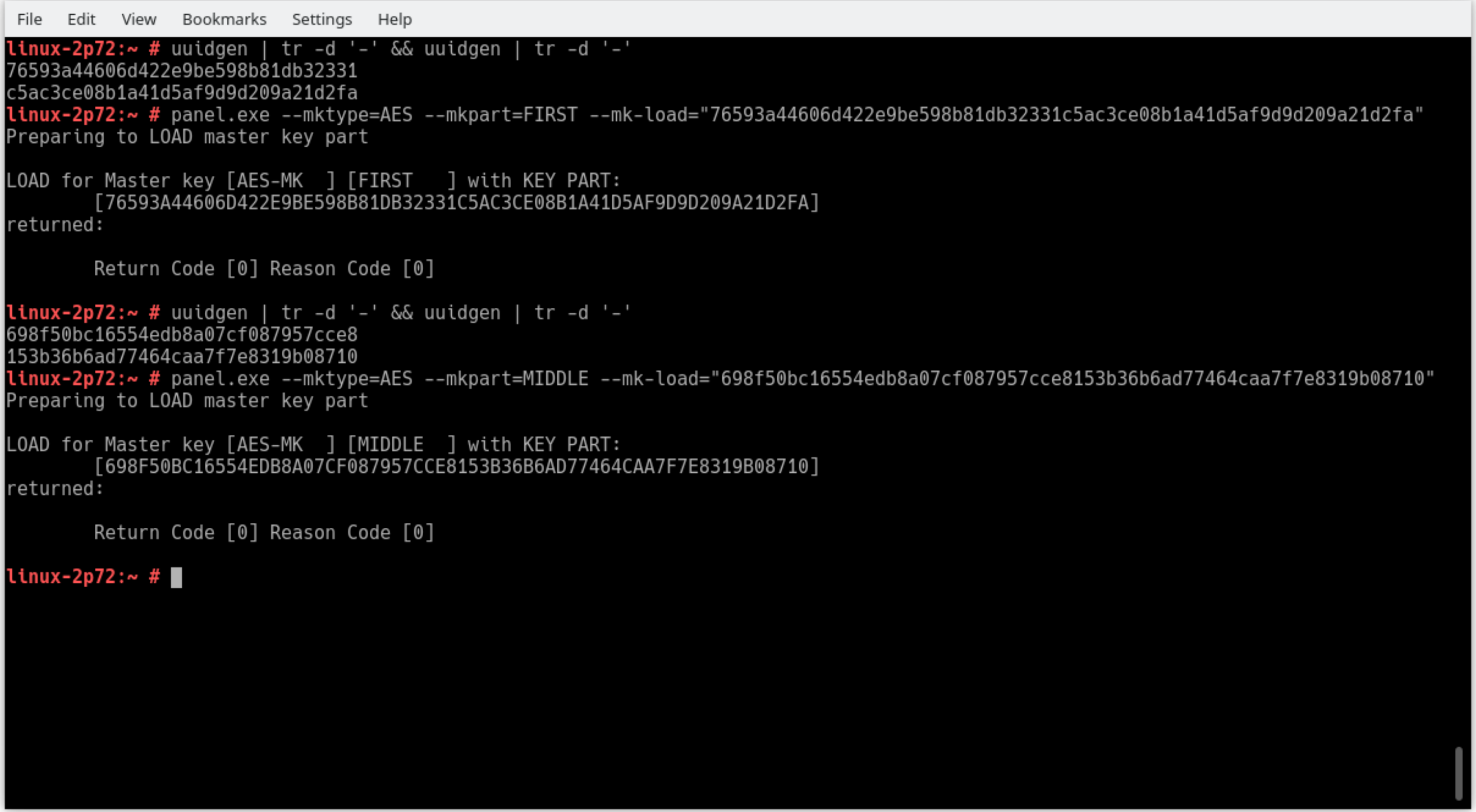
```
linux-2p72:~ # zkey validate --name xtskey-600a
Key                           : xtskey-600a
-------------------------------------------------------------------------------------
        Status                : Valid
        Description           :
        Secure key size       : 128 bytes
        Clear key size        : 512 bits
        XTS type key          : Yes
        Enciphered with       : CURRENT CCA master key
        Volumes               : /dev/disk/by-id/ccw-0X600A-part1:enc-600a
        APQNs                 : 00.0000
        Key file name         : /etc/zkey/repository/xtskey-600a.skey
        Sector size           : 4096 bytes
        Volume type           : LUKS2
        Verification pattern  : 5949f997f0138e3ff04cc9faf808445d
                                c9de9a6c740403bdb6fa80d9576d8592
        Created               : 2019-06-17 12:38:30
        Changed               : (never)
        Re-enciphered         : (never)

1 keys are valid, 0 keys are invalid, 0 warnings
linux-2p72:~ #
```

CURRENT master key verified



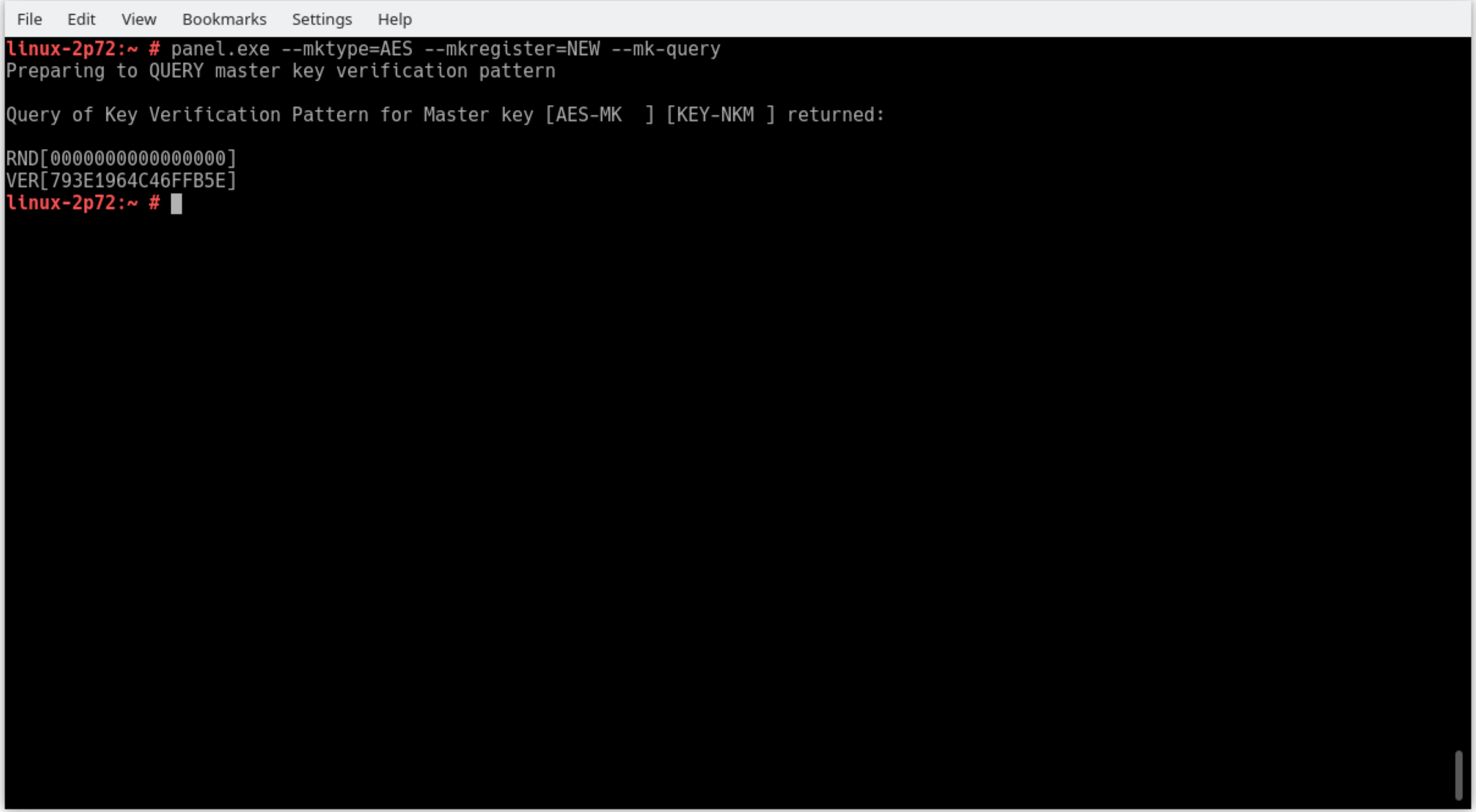
```
linux-2p72:~ # panel.exe --mktype=AES --mkregister=CURRENT --mk-query
Preparing to QUERY master key verification pattern

Query of Key Verification Pattern for Master key [AES-MK   ] [KEY-KM   ] returned:

RND[0000000000000000]
VER[1B92ACA085782622]
linux-2p72:~ #
```

Load NEW master key



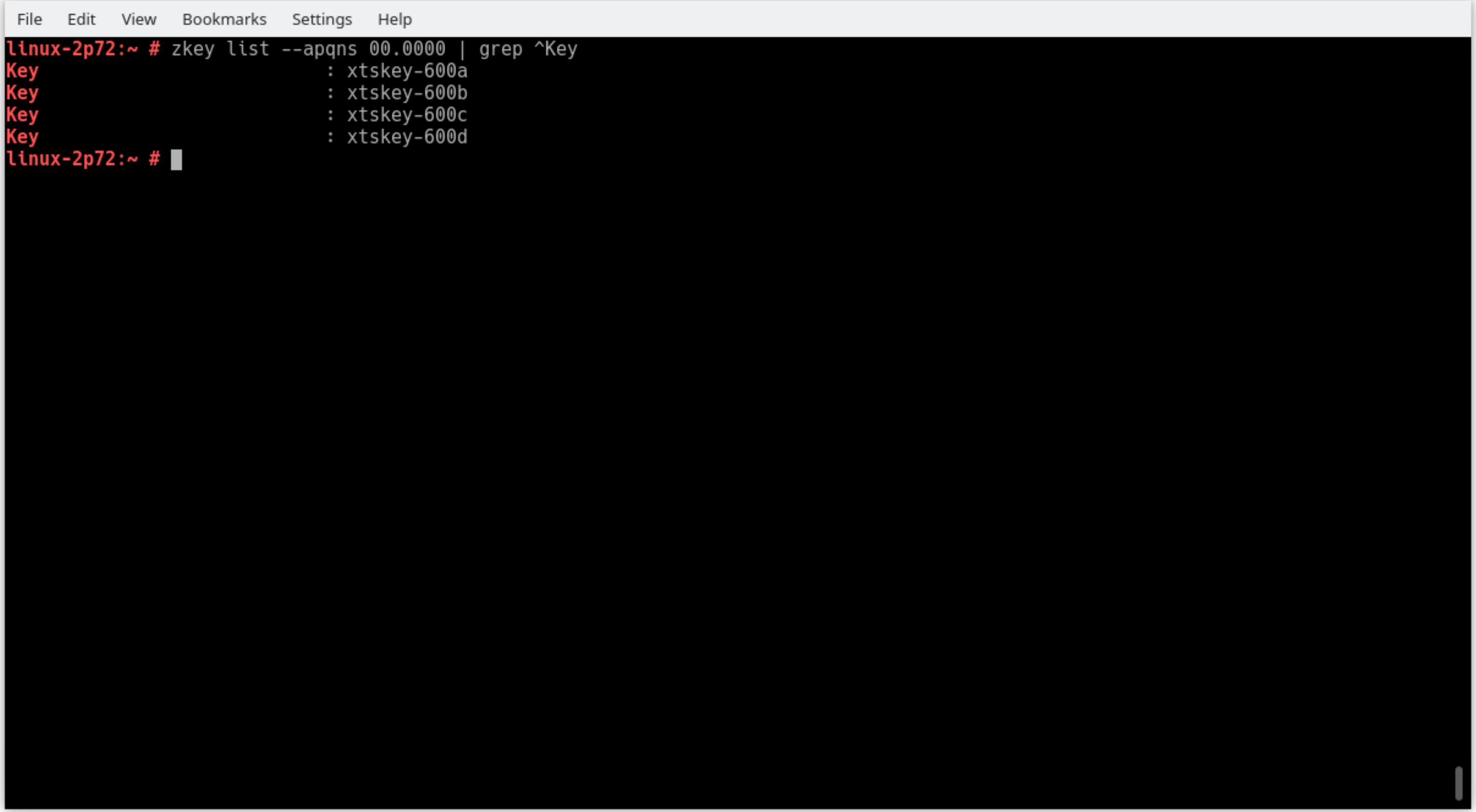
```
linux-2p72:~ # uuidgen | tr -d '-' && uuidgen | tr -d '-'
76593a44606d422e9be598b81db32331
c5ac3ce08b1a41d5af9d9d209a21d2fa
linux-2p72:~ # panel.exe --mktype=AES --mkpart=FIRST --mk-load="76593a44606d422e9be598b81db32331c5ac3ce08b1a41d5af9d9d209a21d2fa"
Preparing to LOAD master key part

LOAD for Master key [AES-MK  ] [FIRST   ] with KEY PART:
        [76593A44606D422E9BE598B81DB32331C5AC3CE08B1A41D5AF9D9D209A21D2FA]
returned:

        Return Code [0] Reason Code [0]

linux-2p72:~ # uuidgen | tr -d '-' && uuidgen | tr -d '-'
698f50bc16554edb8a07cf087957cce8
153b36b6ad77464caa7f7e8319b08710
linux-2p72:~ # panel.exe --mktype=AES --mkpart=MIDDLE --mk-load="698f50bc16554edb8a07cf087957cce8153b36b6ad77464caa7f7e8319b08710"
Preparing to LOAD master key part

LOAD for Master key [AES-MK  ] [MIDDLE  ] with KEY PART:
        [698F50BC16554EDB8A07CF087957CCE8153B36B6AD77464CAA7F7E8319B08710]
returned:

        Return Code [0] Reason Code [0]

linux-2p72:~ #
```

NEW master key verified



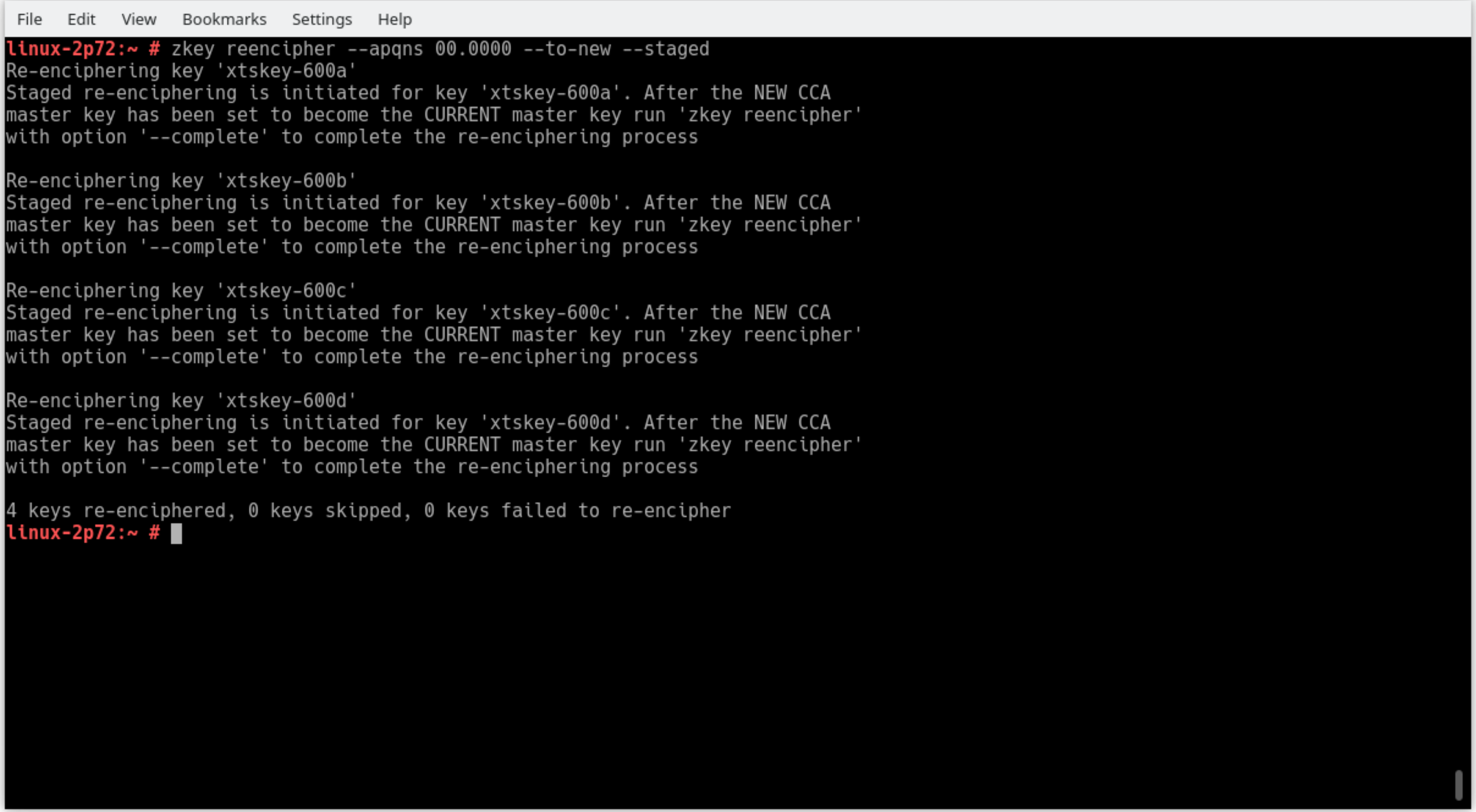
```
linux-2p72:~ # panel.exe --mktype=AES --mkregister=NEW --mk-query
Preparing to QUERY master key verification pattern

Query of Key Verification Pattern for Master key [AES-MK  ] [KEY-NKM ] returned:

RND[0000000000000000]
VER[793E1964C46FFB5E]
linux-2p72:~ #
```

Secure keys ciphered with card 00 domain 0000



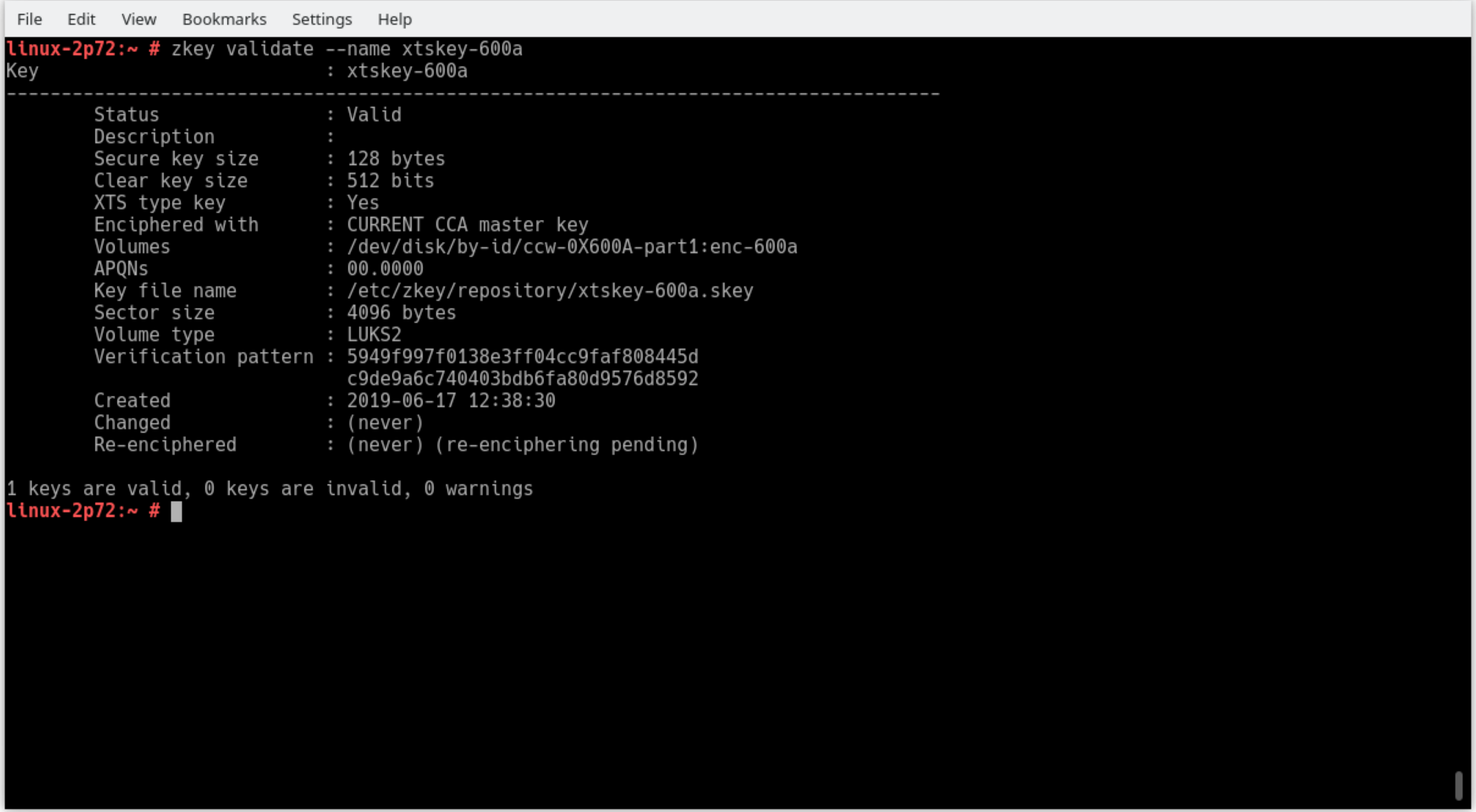
```
linux-2p72:~ # zkey list --apqns 00.0000 | grep ^Key
Key                          : xtskey-600a
Key                          : xtskey-600b
Key                          : xtskey-600c
Key                          : xtskey-600d
linux-2p72:~ #
```

Begin staged re-encipher process



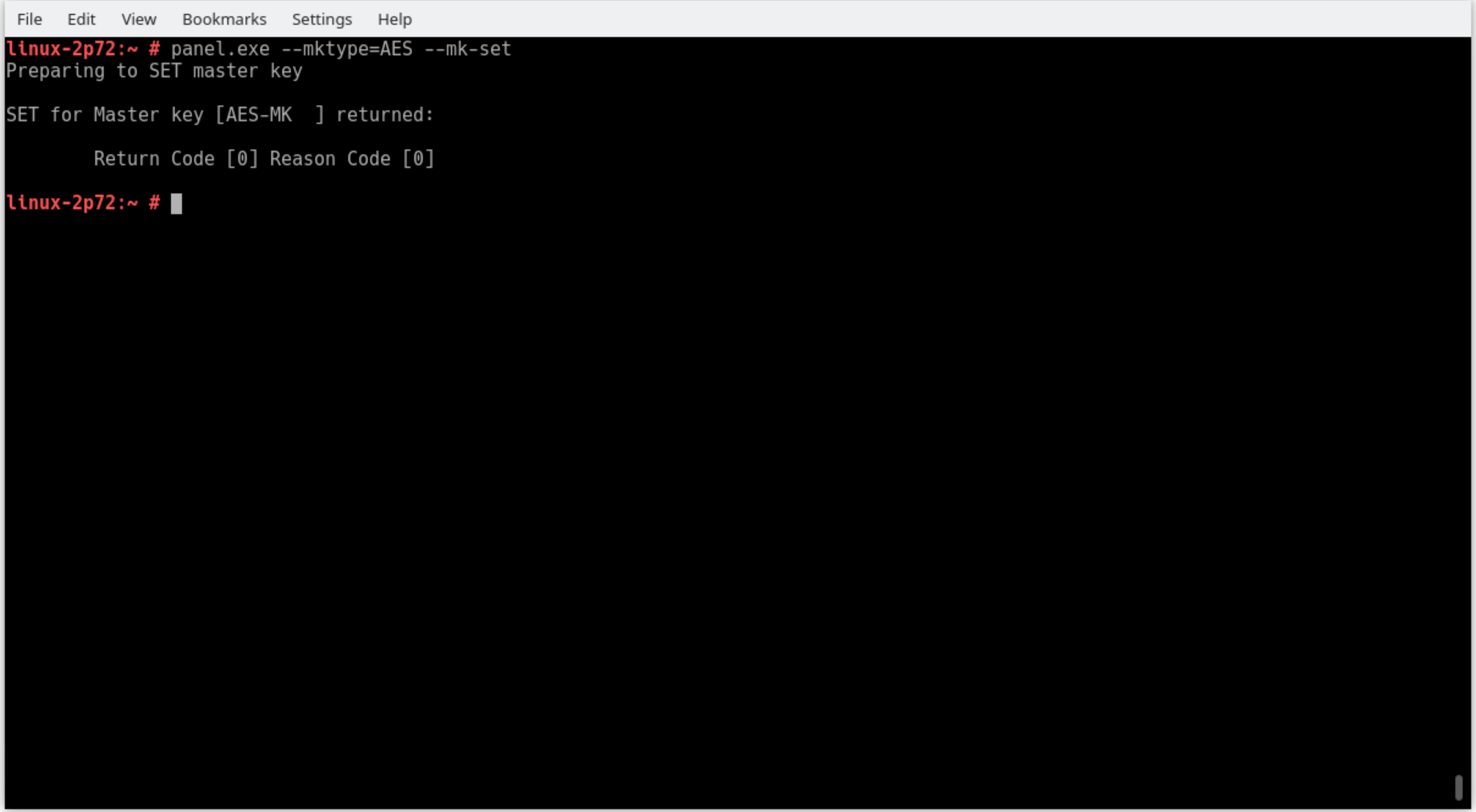
```
linux-2p72:~ # zkey reencipher --apqns 00.0000 --to-new --staged
Re-enciphering key 'xtskey-600a'
Staged re-enciphering is initiated for key 'xtskey-600a'. After the NEW CCA
master key has been set to become the CURRENT master key run 'zkey reencipher'
with option '--complete' to complete the re-enciphering process

Re-enciphering key 'xtskey-600b'
Staged re-enciphering is initiated for key 'xtskey-600b'. After the NEW CCA
master key has been set to become the CURRENT master key run 'zkey reencipher'
with option '--complete' to complete the re-enciphering process

Re-enciphering key 'xtskey-600c'
Staged re-enciphering is initiated for key 'xtskey-600c'. After the NEW CCA
master key has been set to become the CURRENT master key run 'zkey reencipher'
with option '--complete' to complete the re-enciphering process

Re-enciphering key 'xtskey-600d'
Staged re-enciphering is initiated for key 'xtskey-600d'. After the NEW CCA
master key has been set to become the CURRENT master key run 'zkey reencipher'
with option '--complete' to complete the re-enciphering process

4 keys re-enciphered, 0 keys skipped, 0 keys failed to re-encipher
linux-2p72:~ #
```

Secure key showing re-encipher pending



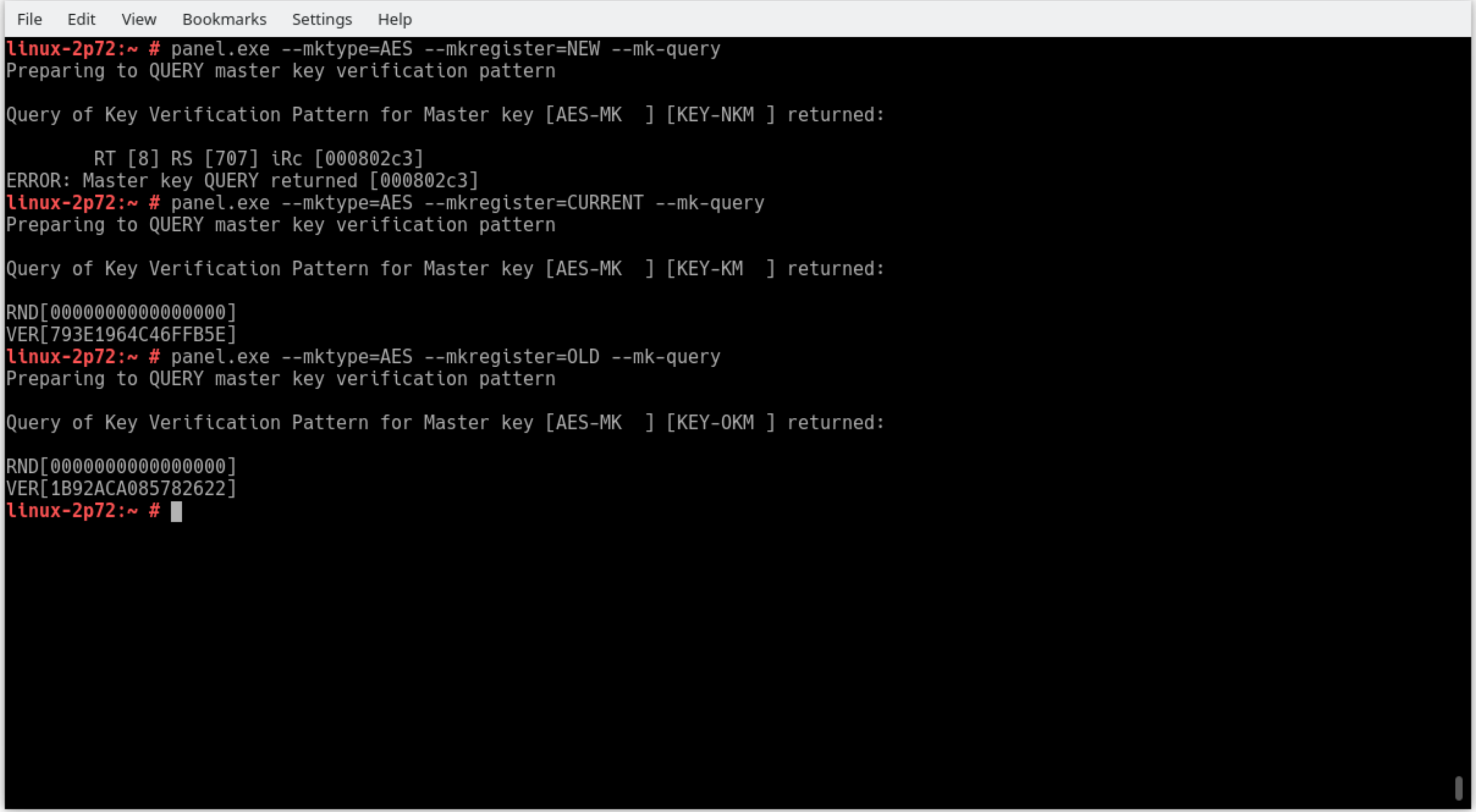
```
linux-2p72:~ # zkey validate --name xtskey-600a
Key                          : xtskey-600a
-----------------------------------------------------------------------------------
        Status               : Valid
        Description          :
        Secure key size      : 128 bytes
        Clear key size       : 512 bits
        XTS type key         : Yes
        Enciphered with      : CURRENT CCA master key
        Volumes              : /dev/disk/by-id/ccw-0X600A-part1:enc-600a
        APQNs                : 00.0000
        Key file name        : /etc/zkey/repository/xtskey-600a.skey
        Sector size          : 4096 bytes
        Volume type          : LUKS2
        Verification pattern : 5949f997f0138e3ff04cc9faf808445d
                               c9de9a6c740403bdb6fa80d9576d8592
        Created              : 2019-06-17 12:38:30
        Changed              : (never)
        Re-enciphered        : (never) (re-enciphering pending)

1 keys are valid, 0 keys are invalid, 0 warnings
linux-2p72:~ # 
```

Promote NEW key to CURRENT key



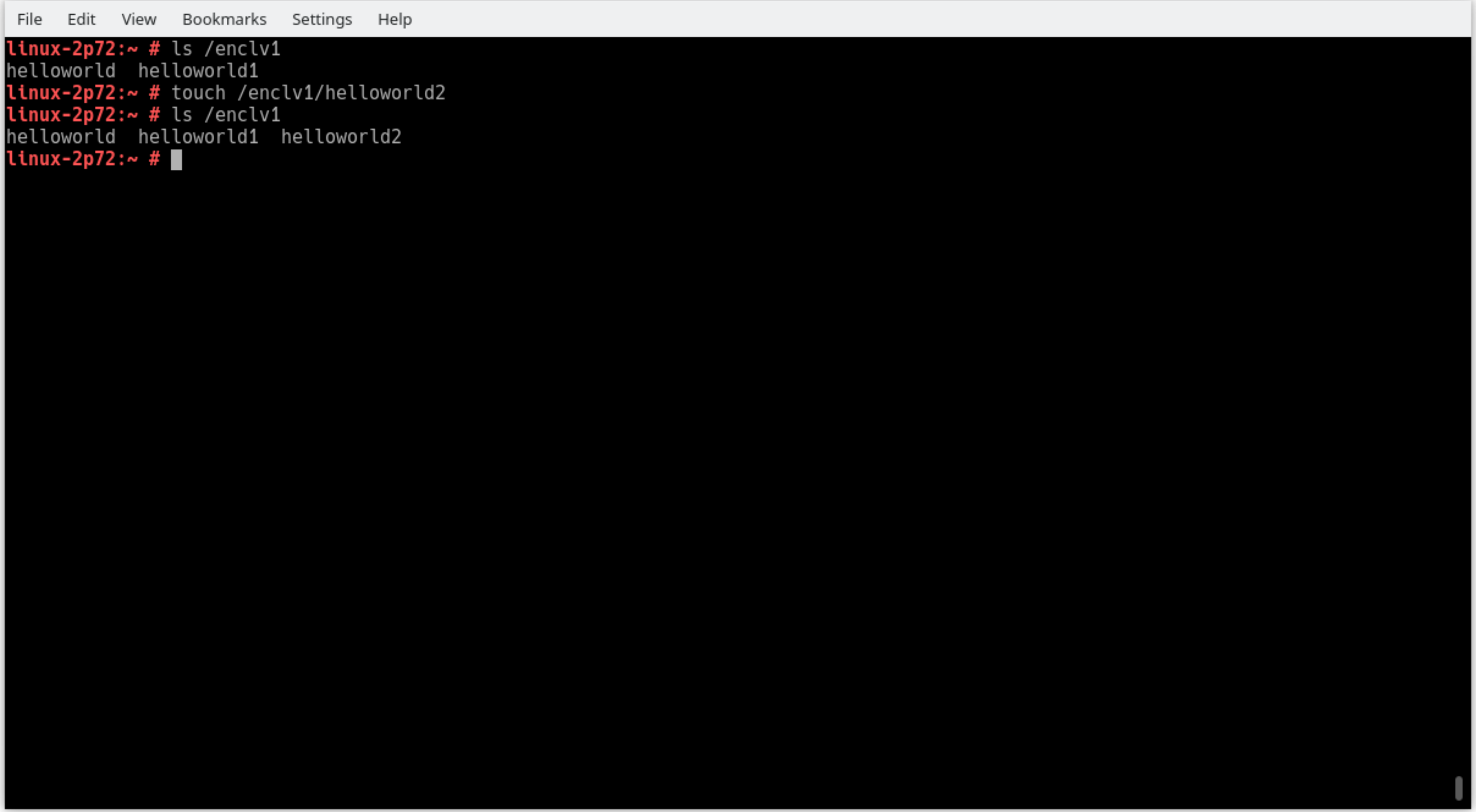
```
linux-2p72:~ # panel.exe --mktype=AES --mk-set
Preparing to SET master key

SET for Master key [AES-MK  ] returned:

        Return Code [0] Reason Code [0]

linux-2p72:~ #
```

NEW key is now the CURRENT key
CURRENT key is now the OLD key



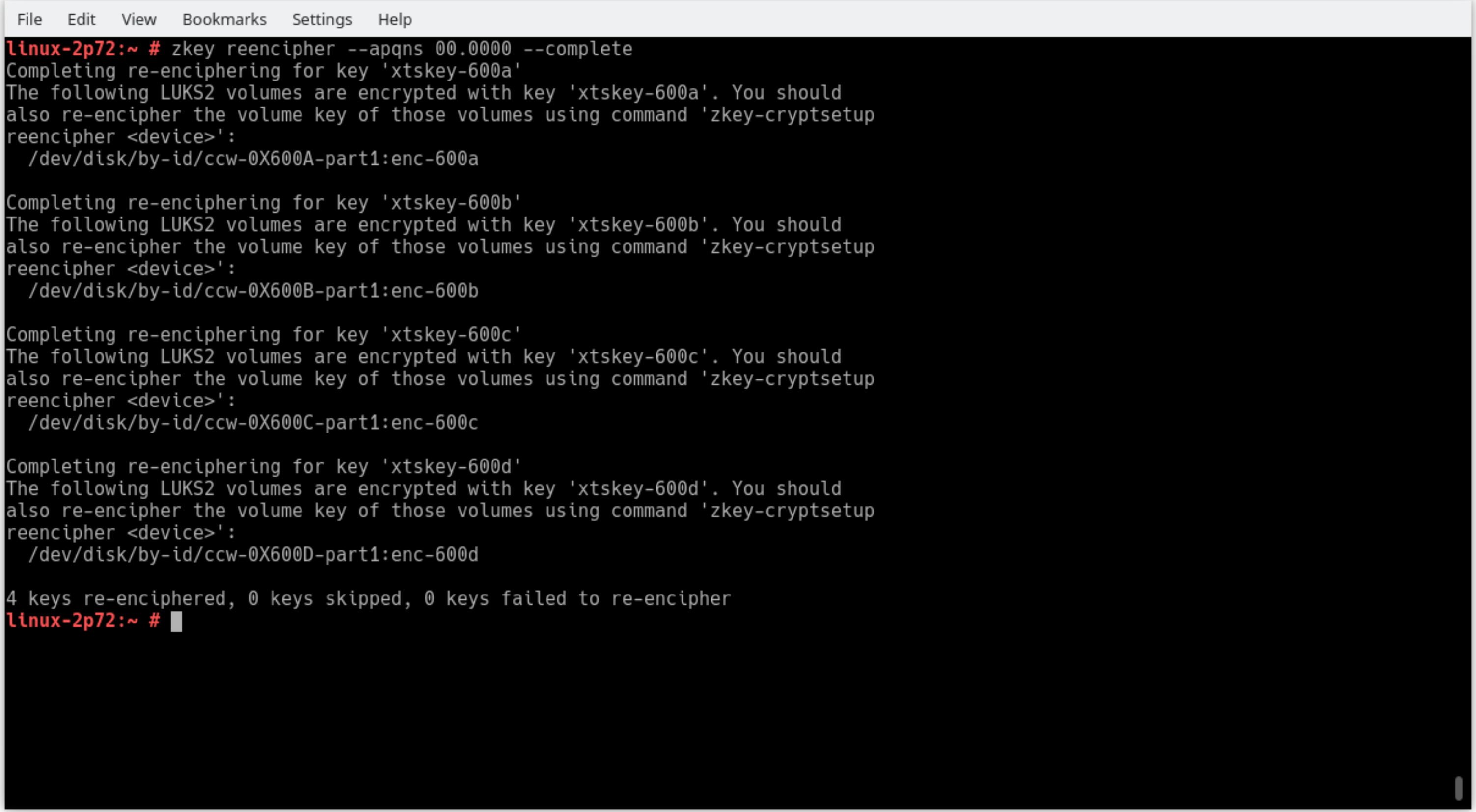
```
linux-2p72:~ # panel.exe --mktype=AES --mkregister=NEW --mk-query
Preparing to QUERY master key verification pattern

Query of Key Verification Pattern for Master key [AES-MK  ] [KEY-NKM ] returned:

        RT [8] RS [707] iRc [000802c3]
ERROR: Master key QUERY returned [000802c3]
linux-2p72:~ # panel.exe --mktype=AES --mkregister=CURRENT --mk-query
Preparing to QUERY master key verification pattern

Query of Key Verification Pattern for Master key [AES-MK  ] [KEY-KM  ] returned:

RND[0000000000000000]
VER[793E1964C46FFB5E]
linux-2p72:~ # panel.exe --mktype=AES --mkregister=OLD --mk-query
Preparing to QUERY master key verification pattern

Query of Key Verification Pattern for Master key [AES-MK  ] [KEY-OKM ] returned:

RND[0000000000000000]
VER[1B92ACA085782622]
linux-2p72:~ # 
```

Still able to write using OLD secure key



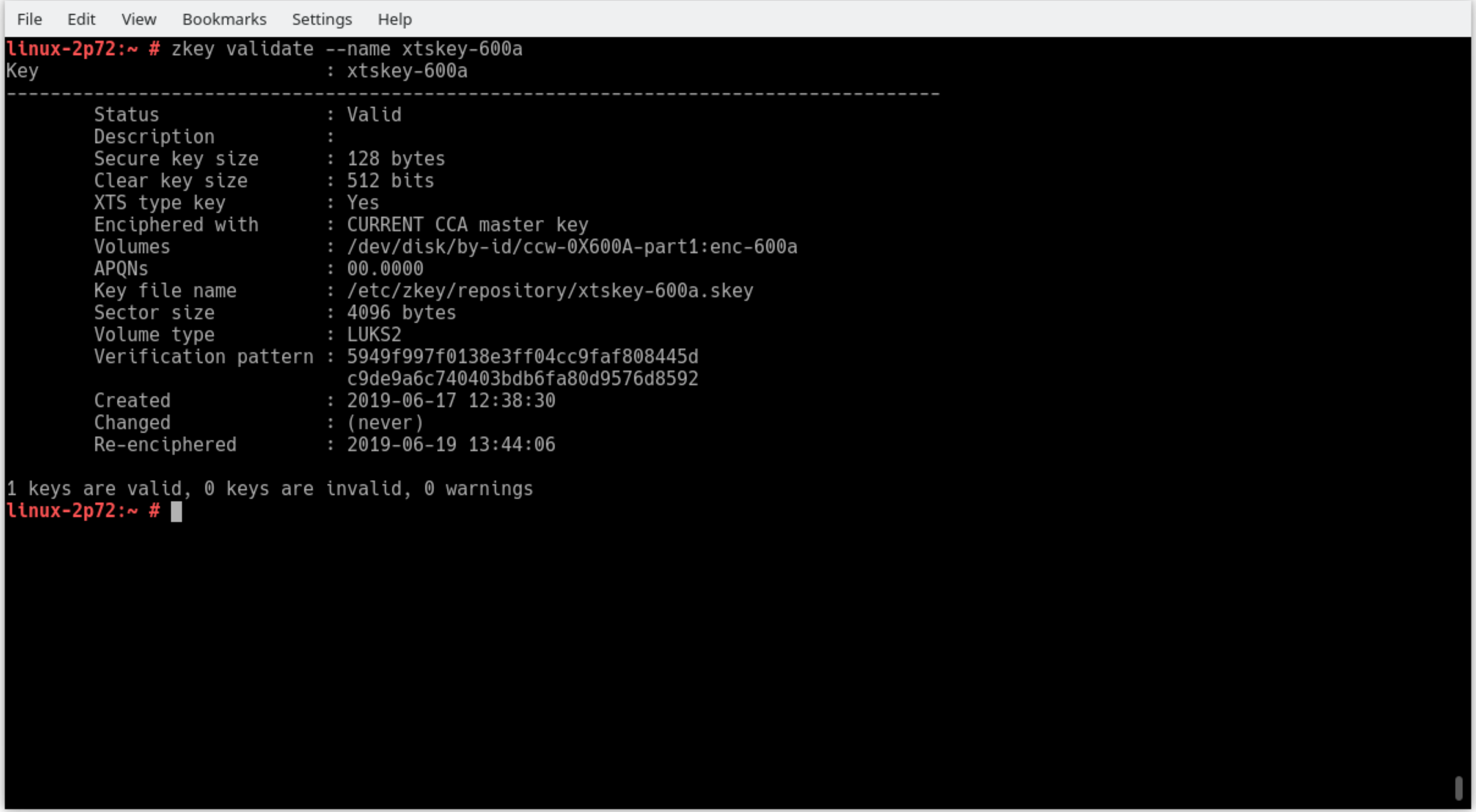
```
linux-2p72:~ # ls /enclv1
helloworld  helloworld1
linux-2p72:~ # touch /enclv1/helloworld2
linux-2p72:~ # ls /enclv1
helloworld  helloworld1  helloworld2
linux-2p72:~ #
```

Complete the re-encipher process



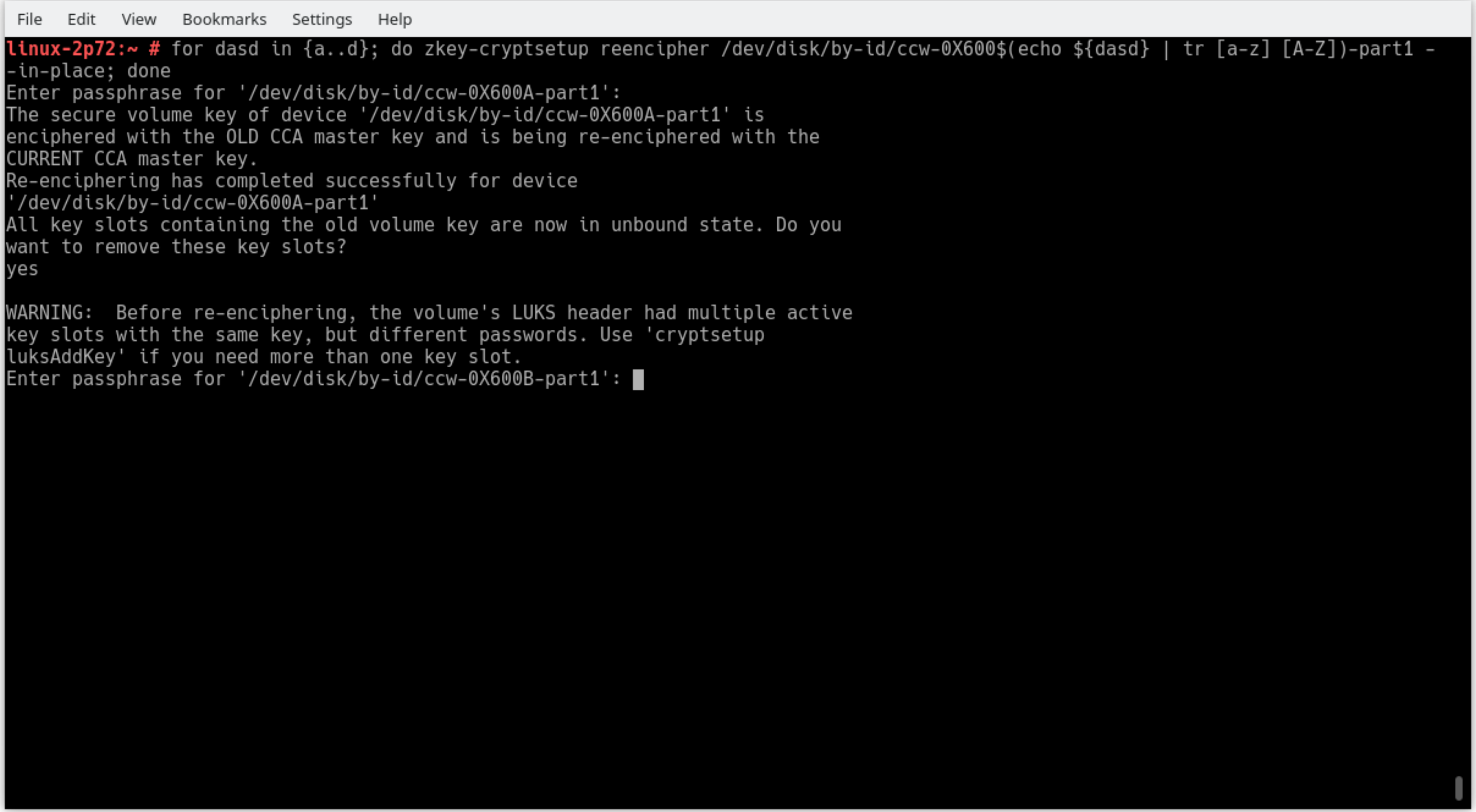
```
linux-2p72:~ # zkey reencipher --apqns 00.0000 --complete
Completing re-enciphering for key 'xtskey-600a'
The following LUKS2 volumes are encrypted with key 'xtskey-600a'. You should
also re-encipher the volume key of those volumes using command 'zkey-cryptsetup
reencipher <device>':
  /dev/disk/by-id/ccw-0X600A-part1:enc-600a

Completing re-enciphering for key 'xtskey-600b'
The following LUKS2 volumes are encrypted with key 'xtskey-600b'. You should
also re-encipher the volume key of those volumes using command 'zkey-cryptsetup
reencipher <device>':
  /dev/disk/by-id/ccw-0X600B-part1:enc-600b

Completing re-enciphering for key 'xtskey-600c'
The following LUKS2 volumes are encrypted with key 'xtskey-600c'. You should
also re-encipher the volume key of those volumes using command 'zkey-cryptsetup
reencipher <device>':
  /dev/disk/by-id/ccw-0X600C-part1:enc-600c

Completing re-enciphering for key 'xtskey-600d'
The following LUKS2 volumes are encrypted with key 'xtskey-600d'. You should
also re-encipher the volume key of those volumes using command 'zkey-cryptsetup
reencipher <device>':
  /dev/disk/by-id/ccw-0X600D-part1:enc-600d

4 keys re-enciphered, 0 keys skipped, 0 keys failed to re-encipher
linux-2p72:~ #
```

The secure key has been re-enciphered



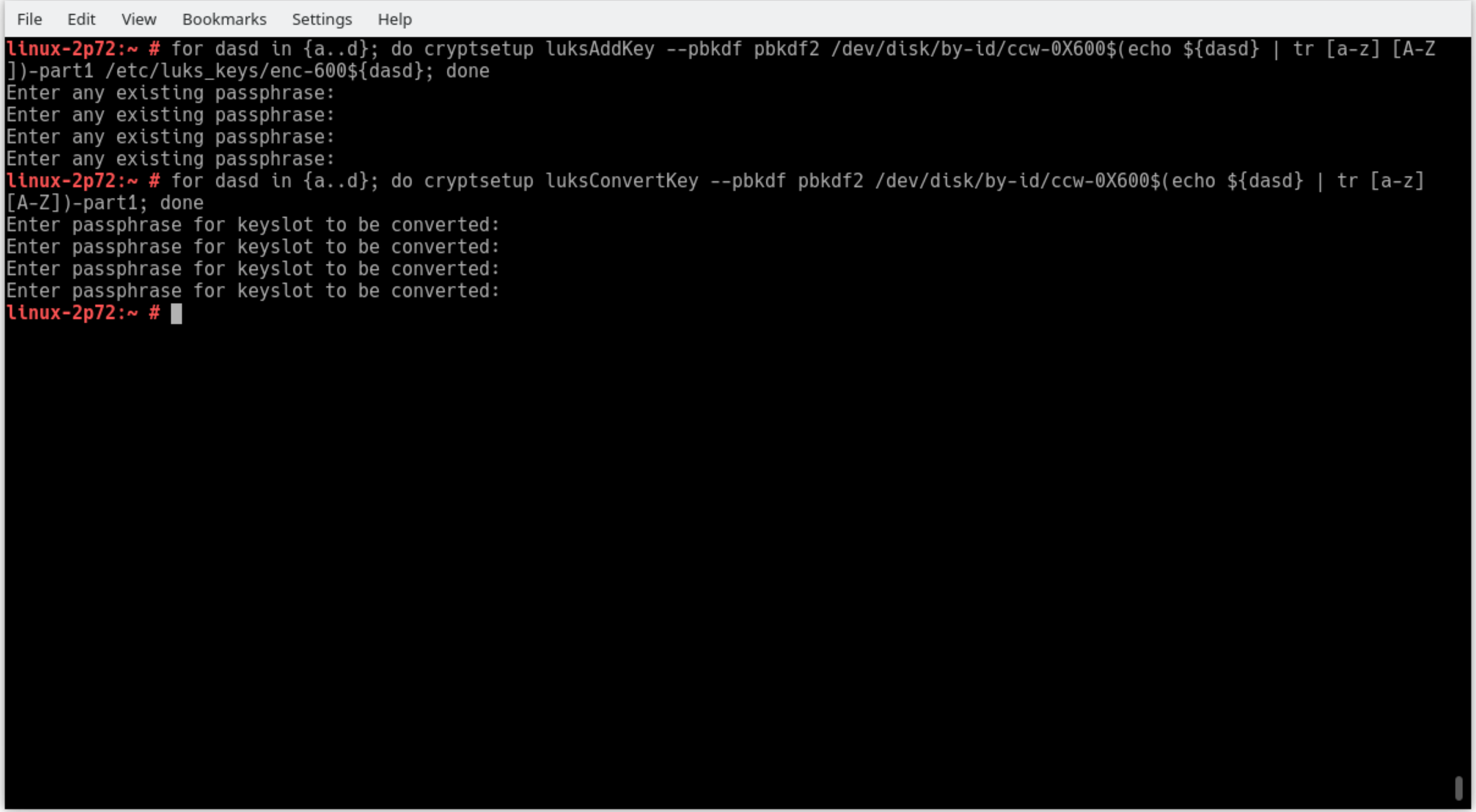
```
linux-2p72:~ # zkey validate --name xtskey-600a
Key                          : xtskey-600a
------------------------------------------------------------------------------------
        Status               : Valid
        Description          :
        Secure key size      : 128 bytes
        Clear key size       : 512 bits
        XTS type key         : Yes
        Enciphered with      : CURRENT CCA master key
        Volumes              : /dev/disk/by-id/ccw-0X600A-part1:enc-600a
        APQNs                : 00.0000
        Key file name        : /etc/zkey/repository/xtskey-600a.skey
        Sector size          : 4096 bytes
        Volume type          : LUKS2
        Verification pattern : 5949f997f0138e3ff04cc9faf808445d
                               c9de9a6c740403bdb6fa80d9576d8592
        Created              : 2019-06-17 12:38:30
        Changed              : (never)
        Re-enciphered        : 2019-06-19 13:44:06

1 keys are valid, 0 keys are invalid, 0 warnings
linux-2p72:~ #
```

Re-encipher the LUKS2 header for each partition



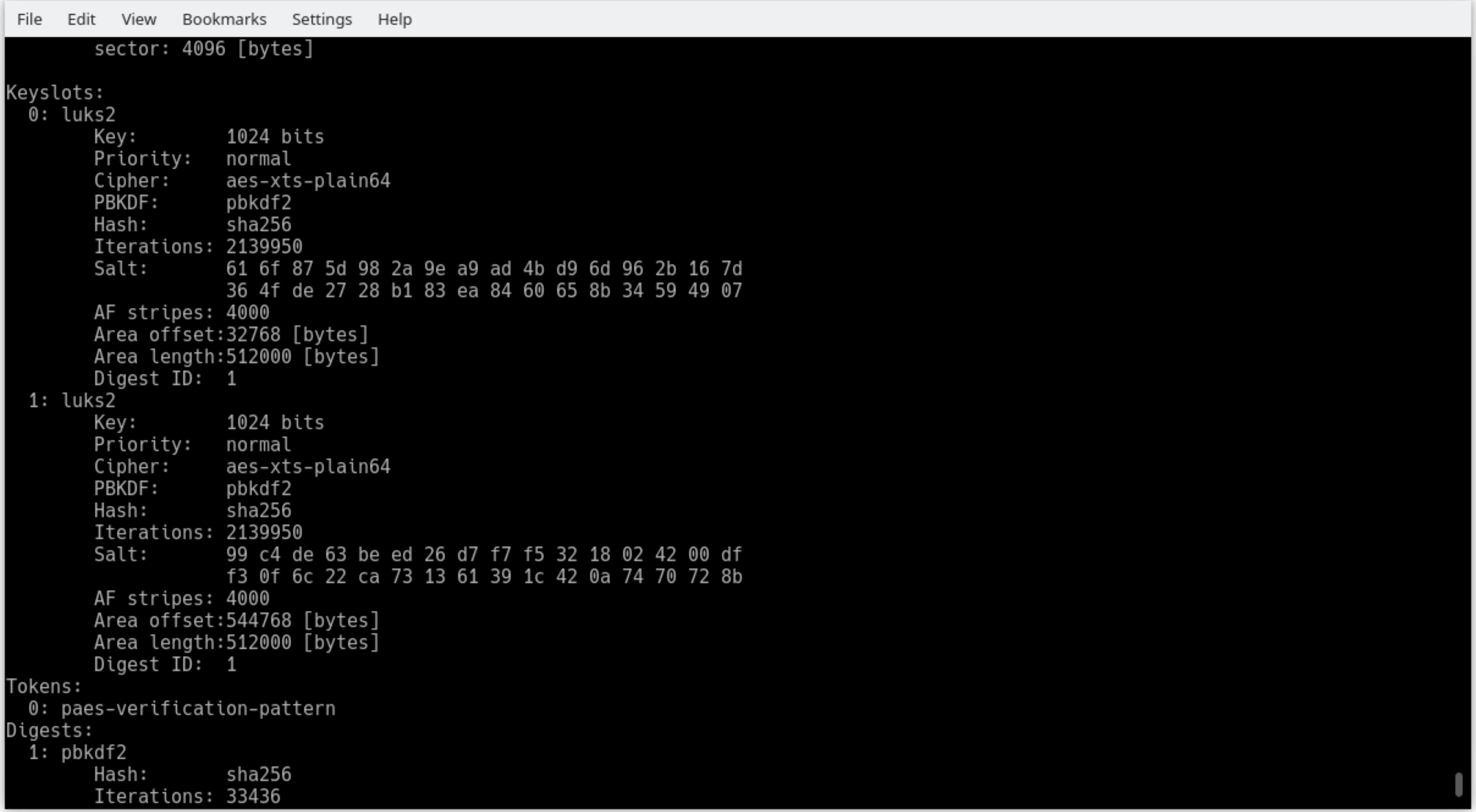
```
linux-2p72:~ # for dasd in {a..d}; do zkey-cryptsetup reencipher /dev/disk/by-id/ccw-0X600$(echo ${dasd} | tr [a-z] [A-Z])-part1 -
-in-place; done
Enter passphrase for '/dev/disk/by-id/ccw-0X600A-part1':
The secure volume key of device '/dev/disk/by-id/ccw-0X600A-part1' is
enciphered with the OLD CCA master key and is being re-enciphered with the
CURRENT CCA master key.
Re-enciphering has completed successfully for device
'/dev/disk/by-id/ccw-0X600A-part1'
All key slots containing the old volume key are now in unbound state. Do you
want to remove these key slots?
yes

WARNING:  Before re-enciphering, the volume's LUKS header had multiple active
key slots with the same key, but different passwords. Use 'cryptsetup
luksAddKey' if you need more than one key slot.
Enter passphrase for '/dev/disk/by-id/ccw-0X600B-part1':
```

Re-add keys for auto opening



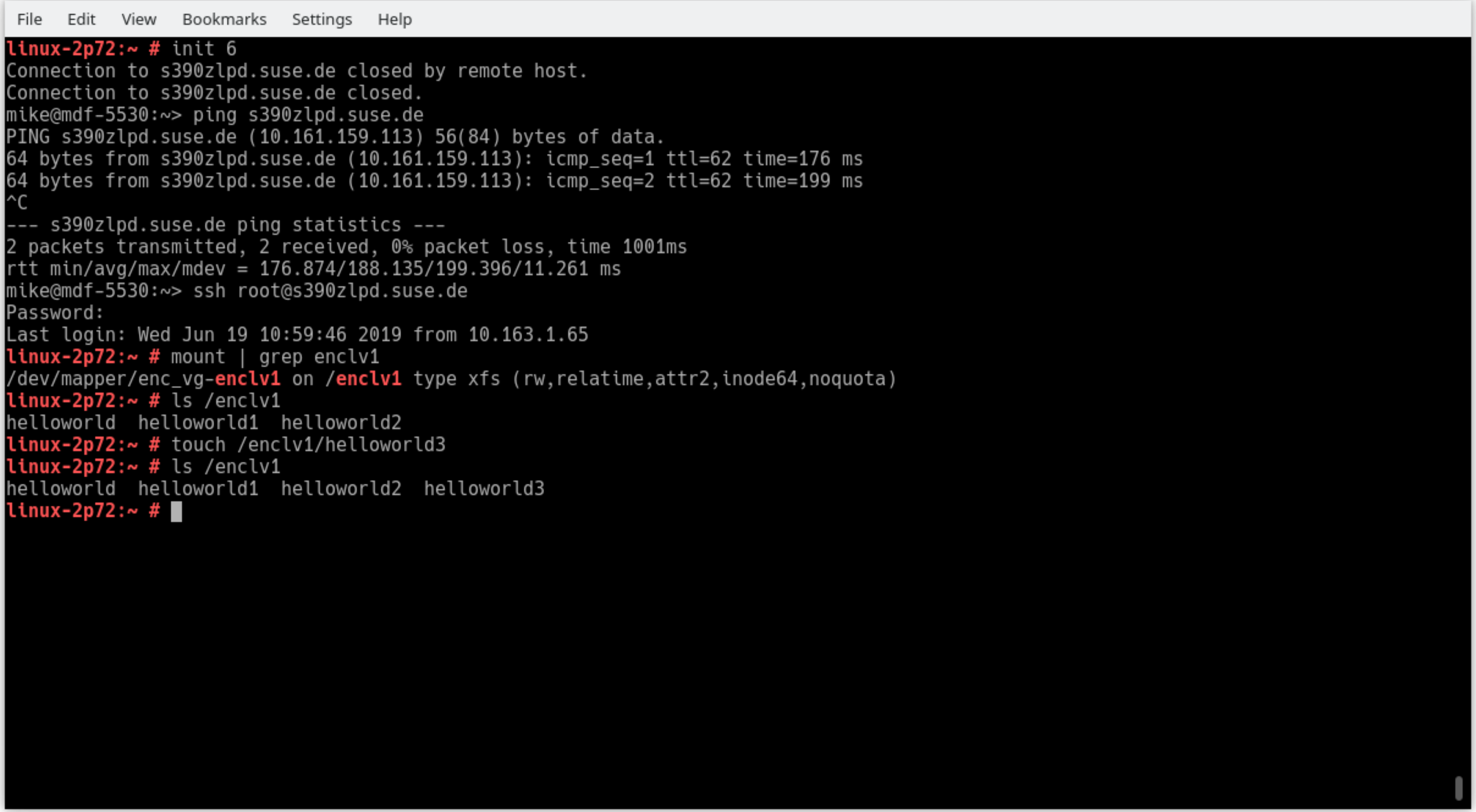
```
linux-2p72:~ # for dasd in {a..d}; do cryptsetup luksAddKey --pbkdf pbkdf2 /dev/disk/by-id/ccw-0X600$(echo ${dasd} | tr [a-z] [A-Z
])-part1 /etc/luks_keys/enc-600${dasd}; done
Enter any existing passphrase:
Enter any existing passphrase:
Enter any existing passphrase:
Enter any existing passphrase:
linux-2p72:~ # for dasd in {a..d}; do cryptsetup luksConvertKey --pbkdf pbkdf2 /dev/disk/by-id/ccw-0X600$(echo ${dasd} | tr [a-z]
[A-Z])-part1; done
Enter passphrase for keyslot to be converted:
Enter passphrase for keyslot to be converted:
Enter passphrase for keyslot to be converted:
Enter passphrase for keyslot to be converted:
linux-2p72:~ #
```

cryptsetup luksdump showing two new keyslots after re-enciphered LUKS2 header



```
        sector: 4096 [bytes]

Keyslots:
  0: luks2
        Key:        1024 bits
        Priority:   normal
        Cipher:     aes-xts-plain64
        PBKDF:      pbkdf2
        Hash:       sha256
        Iterations: 2139950
        Salt:       61 6f 87 5d 98 2a 9e a9 ad 4b d9 6d 96 2b 16 7d
                    36 4f de 27 28 b1 83 ea 84 60 65 8b 34 59 49 07
        AF stripes: 4000
        Area offset:32768 [bytes]
        Area length:512000 [bytes]
        Digest ID:  1
  1: luks2
        Key:        1024 bits
        Priority:   normal
        Cipher:     aes-xts-plain64
        PBKDF:      pbkdf2
        Hash:       sha256
        Iterations: 2139950
        Salt:       99 c4 de 63 be ed 26 d7 f7 f5 32 18 02 42 00 df
                    f3 0f 6c 22 ca 73 13 61 39 1c 42 0a 74 70 72 8b
        AF stripes: 4000
        Area offset:544768 [bytes]
        Area length:512000 [bytes]
        Digest ID:  1
Tokens:
  0: paes-verification-pattern
Digests:
  1: pbkdf2
        Hash:       sha256
        Iterations: 33436
```

Auto open and writing to encrypted volume continues to work after reboot



```
linux-2p72:~ # init 6
Connection to s390zlpd.suse.de closed by remote host.
Connection to s390zlpd.suse.de closed.
mike@mdf-5530:~> ping s390zlpd.suse.de
PING s390zlpd.suse.de (10.161.159.113) 56(84) bytes of data.
64 bytes from s390zlpd.suse.de (10.161.159.113): icmp_seq=1 ttl=62 time=176 ms
64 bytes from s390zlpd.suse.de (10.161.159.113): icmp_seq=2 ttl=62 time=199 ms
^C
--- s390zlpd.suse.de ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 176.874/188.135/199.396/11.261 ms
mike@mdf-5530:~> ssh root@s390zlpd.suse.de
Password:
Last login: Wed Jun 19 10:59:46 2019 from 10.163.1.65
linux-2p72:~ # mount | grep enclv1
/dev/mapper/enc_vg-enclv1 on /enclv1 type xfs (rw,relatime,attr2,inode64,noquota)
linux-2p72:~ # ls /enclv1
helloworld  helloworld1  helloworld2
linux-2p72:~ # touch /enclv1/helloworld3
linux-2p72:~ # ls /enclv1
helloworld  helloworld1  helloworld2  helloworld3
linux-2p72:~ # 
```

What is next?

What is next?

Working on documenting the migration of a SLES install from unencrypted to encrypted

Videos and How-to guide(s)

YAST installer support for protected key dm-crypt?

Include examples of configuring data-in-flight encryption

SUSE®

We adapt. You succeed.