Introduction

Update of Install OpenBSD 5.9 updated to openBSD 6.7 Tired of Systemd and some other layers in Linux, I will try to install OpenBSD my x230 Tablet, encrypting partitions with a key stored on a SD card/USBkey. (hello NSA)
I will use an openBSD67.
The x230 is modified, I added a mSATA drive (sd1) and changed the 500GB drive to a 1TB SSD(sd0) .
I removed totally Windows as I don't use it.

====== Downloads ======Lenovo X230T Install - Encrypted Install67.iso take the install*.fs for USB keys

disk Layout

I will use the following schema:

- mSATA (sd1) 256GB for /
- SATA (sd0) 1000GB for swap/var/home

Disk	Partition	Format	Size	Mount Point
1	1	RAID	230GB	/
1	2	none	26GB	fake trim
0	1	swap	16GB	OpenBSD swap
0	2	RAID	950GB	/var, /tmp and /home

Installation

Boot from the iso/usb key
Select the shell (by type '**s**') when arriving on the following:
(I)nstall, (U)pgrade, (A)utoinstall, (S)hell?

Disk partitionning

to know what disks are recognised

```
sysctl hw.disknames
```

We will create the above partitions using fdisk/disklabel tools.

```
fdisk -iy sd1
```

the output is

```
fdisk ; sd1: No such file or directory
```

Let's create the device

```
cd /dev
sh ./MAKEDEV sd0 sd1
```

An run again:

```
fdisk -iy sd1
```

the output is largely nicer:

```
Writing MBR at offset 0.
```

```
fdisk -iy sd0
```

Let's partition sd0 and sd1

```
disklabel -E sd1
sd1> a
partition: [a]
offset: [64]
size: [500103386] 95%
FS type: [4.2BSD] RAID
sd1*> w
sd1> q
```

Done for sd1 (I leave 5% free for TRIM as I got no clear details whether or not I need it or not if it is working or not and well 95% of 256GB should be enough for / )
Let's partition sd0

```
disklabel -E sd0
sd0> a
partition: [a]
offset: [0]
size: [976773168] 16g
FS type: [4.2BSD] swap
sd0> a
partition: [b]
offset: [33559785]
size: [943213383]
FS type: [swap] RAID
sd0*> w
sd0> q
```

Here we do 2 partitions, swap (already encrypted) and a RAID one that will contained /var /tmp and

/home)

SDcard/USBkey partitioning

Plug in your SD card/ USB stick, here it is named sd2

```
cd /dev
sh ./MAKEDEV sd2
dd if=/dev/urandom of=/dev/rsd2c bs=1m
```

Don't forget to create the MBR

```
 fdisk -iy sd2
```

We just need around 1MB for the key so I will just create two very small partitions (d and e).

```
disklabel -E sd2
sd2> a
partition: [a]d
offset: [64]
size: [124735488] 2m
FS type: [4.2BSD] RAID
sd2> a
partition: [a]e
offset: [16065]
size: [124735488] 2m
FS type: [4.2BSD] RAID
> w
> q
```

Let's encrypt

So for now we have sd1(m-sata)/sd0(SSD)/sd2(USB key)
Next step is an all-in-one step: we will encrypt sd1a and sd0b as softraid0 and save the key on the SD card.

```
bioctl -c C -C force -k sd2d -l sd0b softraid0
bioctl -c C -C force -k sd2e -l sd1a softraid0
```

Note: http://www.tedunangst.com/flak/post/OpenBSD-softraid-crypto-boot
Note 2: if you don't put **-C force** you might end up with

```
 openbsd softraid0 chunk already in use
```

Output will be something like:

```
sd3 at scsibus1 targ 1 lun 0: <OPENBSD, SR CRYPTO, 006>
sd3: 937482MB, 512 bytes/sector, 1919964855 sectors
softraid0: CRYPTO volume attached as sd3
```

and

```
sd4 at scsibus1 targ 2 lun 0: <OPENBSD, SR CRYPTO, 006>
sd4: 244190MB, 512 bytes/sector, 50010858 sectors
softraid0: CRYPTO volume attached as sd4
```

We create the devices:

```
cd /dev
sh MAKEDEV sd3 sd4
```

Good practice, erase first MegaByte:

```
dd if=/dev/zero of=/dev/rsd3c bs=1m count=1
dd if=/dev/zero of=/dev/rsd4c bs=1m count=1
```

So now we have sd3 and sd4 which we will use during the installation.

Install

Let's go back to the installation

```
/install
```

Select layout, configure network[em0], root password, sshd, timezone, user(s)…
When at the partitioning point, select the 2 new device (sd3 and sd4 in my case).

```
Which disk is the root disk? [sd0] sd3
Use (W)hole disk MBR, whole disk (G)PT or (E)dit? [whole]W
[… auto partitions schema …]
Use (A)uto layout, (E)dit auto layout, or create (C)ustom layout? [a] c
sd3> a
partition: [a]a
offset: [64]
size: [475106246]
FS type: [4.2BSD]
mount point: [none] /
> w
> q
```

We do the same for sd4 (/var /tmp and /home)

```
which disk do you wish to initialize? [done] sd4
Use (W)hole disk MBR, whole disk (G)PT or (E)dit? [whole]W
```

```
sd4> a
partition: [a]a
offset: [64]
size: [943208216] 20g
FS type: [4.2BSD]
mount point: [none] /var
sd4> a
partition: [b]
offset: [41945696]
size: [901262584] 10g
FS type: [swap] 4.2BSD
mount point: [none] /tmp
sd4> a
partition: [d]
offset: [62910528]
size: [880297752]
FS type: [4.2BSD]
mount point: [none] /home
> w
> q
```

And voilà.

```
which disk do you wish to initialize? [done]
```

Install the sets

http
no proxy
cdn.openbsd.org
/pub/OpenBSD/6.7/amd64
-game*

Installation should finish with a success, but **DO NOT REBOOT YET**

Final touch

We will tweak quickly the fstab to add the swap partition as well as some parameters:

```
sed 's/rw/rw,softdep,noatime/g' /mnt/etc/fstab > /mnt/a
echo '/dev/sd0a none swap sw 0 0' >> /mnt/a
mv /mnt/a /mnt/etc/fstab  Still in ~/.xsession
<code bash>
# activate scroll wheel button
xinput set-prop "/dev/wsmouse" "WS Pointer Wheel Emulation" 0
xinput set-prop "/dev/wsmouse" "WS Pointer Wheel Emulation Axes" 6 7 4 5
xinput set-prop "/dev/wsmouse" "WS Pointer Wheel Emulation Button" 2
xinput set-prop "/dev/wsmouse" "WS Pointer Wheel Emulation Timeout" 50
xinput set-prop "/dev/wsmouse" "WS Pointer Wheel Emulation Inertia" 3
```

```
```

Done, everything is ready! ready to reboot

```
reboot
```

The trick for all this installation is that all sd* MUST have MBR otherwise you get an error like

```
openbsd softraid0: not valid softraid metadata
```

or

```
openbsd installboot: invalid boot record signature 0x0000 @ sector 0 Failed to install bootblocks -hyper
```

Solution for the above errors:

```
 dd if=/dev/zero of=/dev/rsdXc bs=10m count=1
```

then repartition

Backup of the Keys

```
dd if=/dev/rsd2d of=key1.img bs=1m
dd if=/dev/rsd2e of=key2.img bs=1m
```

Switching to -current

Optional, but thrilling!

```
cd /
mv /bsd.rd /bsd.rd.sav
wget http://mirror.switch.ch/ftp/pub/OpenBSD/snapshots/amd64/bsd.rd
```

Reboot

```
reboot
```

At boot prompt type:

```
boot> boot bsd.rd
```

Then use the “upgrade” method, give your keyboard layout, your root device/partition.

When upgrade is done, type again “reboot”.
Once reboot use sysmerge to merge/check modifications

```
sysmerge
```

And voilà ! you are in -current

Laptop tweaks

apm

In order to keep some battery:

```
vi /etc/rc.conf.local
```

add

```
apmd_flags="-A"          # Set apmd(8) to automatic performance adjustment
mode.
apmd_enable="YES"
```

or

```
rcctl enable apmd
rcctl set apmd flags -A
rcctl start apmd
```

desktop reactivity

In /etc/rc.conf.local add:

```
multicast_host=YES
ntpd_flags="-s"
hotplugd_flags=""
```

In /etc/login.conf change:

```
        :datasize-max=512M:\
        :datasize-cur=512M:\
```

by (at least)

```
        :datasize-max=1024M:\
        :datasize-cur=1024M:\
```

Reference: http://www.bsdnow.tv/tutorials/the-desktop-obsd

Xorg

All set nicely

Touchpad

In your ~/.xsession

```
#!/bin/sh

# deactivate touchpad
synclient TouchpadOff=1
```

Scrolling

increase pointer speed

```
xinput set-prop "/dev/wsmouse" "Device Accel Constant Deceleration" 0.4
```

Locales

Still in ~/.xsession

```
# set locale
export LC_CTYPE="en_US.UTF-8"
export LC_MESSAGES="en_US.UTF-8"
```

Network

First download everything needed:

```
syspatch
fw_update
```

ifconfig is your friend!
Simple example:

```
ifconfig iwn0 nwid "my nice wifi" wpakey "4m4z1ngP4$$" up
```

then if all is ok:

```
dhclient iwn0
```

You can scan networks (wifi)

```
ifconfig iwn0 scan
```

Failover network

local DNS

Security

PF

Basic rules in your /etc/pf.conf

```
#       $OpenBSD: pf.conf,v 1.54 2014/08/23 05:49:42 deraadt Exp $
#
# See pf.conf(5) and /etc/examples/pf.conf

set skip on lo

block return    # block stateless traffic
pass            # establish keep-state

# By default, do not permit remote connections to X11
block return in on ! lo0 proto tcp to port 6000:6010

set block-policy drop
match in all scrub (no-df random-id max-mss 1440)
antispoof quick for (egress)
block in quick on egress from { no-route urpf-failed } to any
block in all
pass out quick inet keep state
```

References: http://www.bsdnow.tv/tutorials/the-desktop-obsd

Fingerprint

```
 pkg_add -iv dbus fprintd login_fingerprint libfprint
/etc/rc.d/messagebus start
rcctl enable messagebus
```

Additional Packages

Time to play with pkg_add to put some useful tools:

```
pkg_add -Uu
pkg_add -iv firefox
pkg_add -iv ImageMagick irssi vim zsh rxvt-unicode fvwm2 xscreensaver
rdesktop iftop rsync wget curl figlet dfc git subversion ranger emelfm2 ccze
htop
```

Ports

You can install ports using a snapshot made for you!

```
cd /usr
wget https://mirror.ungleich.ch/pub/OpenBSD/snapshots/ports.tar.gz
tar xvzf ports.tar.gz
```

Finding packages

Two ways:

```
pkg_info yourpackage
```

or

```
cd /usr/ports
make search key=what_you_search
```

Reference: http://www.bsdnow.tv/tutorials/ports-obsd

Thing went wrong when

bootblocks are not installed

```
→ installboot: no OpenBSD partition
Failed to install bootblocks
You will not be able to boot OpenBSD from sd4
```

I had this issue when sd0 didn't had a MBR nor sd3 … dd saved the days :)

References

- http://unix.stackexchange.com/questions/9527/how-should-one-set-up-full-disk-encryption-on-openbsd
- https://ryanak.ca/planet-ubuntu/2013/03/26/Setting-up-full-disk-encryption-in-OpenBSD-5.3.html
- http://geekyschmidt.com/2011/01/19/configuring-openbsd-softraid-fo-encryption
- http://brycv.com/blog/2012/encrypted-root-filesystem-using-softraid-4-on-openbsd-with-an-slc-ssd/
- http://www.bsdnow.tv/tutorials/fde
- http://man.openbsd.org/OpenBSD-current/man8/bioctl.8
- http://geodsoft.com/howto/harden/OpenBSD/no_changes.htm
- http://fsfe.soup.io/post/669752294/emergency-exit-OpenBSD-on-the-Thinkpad-X250
- http://www.openbsd.org/faq/faq4.html
- http://undeadly.org/cgi?action=article&sid=20110530221728
- https://github.com/jhautefeuille/obsdnfo
- http://www.openbsd.org/faq/faq14.html
- http://geekyschmidt.com/2011/03/27/openbsd-laptop-mini-howto
- http://www.tumfatig.net/20150215/bind-nsd-unbound-openbsd-5-6/
- http://www.openbsd.org/faq/faq6.html
- https://www.tumfatig.net/20190311/openbsd-laptop-with-full-disk-encryption/
- https://www.c0ffee.net/blog/openbsd-on-a-laptop/