Encrypted FS

Encrypted partition

```
apt-get install cryptsetup-bin
```

Enable HW acceleration. Which is a bit slower than software :P

NOTE: From Kernel 4.2 cesa driver was completely rewritten to support DMA, and old **mv_cesa** driver was removed in kernel 4.15

Kernel 3.18

```
modprobe mv_cesa
cat /proc/crypto | grep mv_cesa -B 2 -A 7
```

Is providing only:

1. hmac(sha1)
2. sha1
3. cbc(aes)
4. ecb(aes)

There are also additional kernel modules optimised for ARM:

1. sha1_arm
2. aes_arm

Kernel 5.8

```
modprobe mv_cesa
cat /proc/crypto | grep cesa -B 2 -A 7
```

Is providing:

1. hmac(sha1)
2. hmac(md5)
3. sha1
4. md5
5. cbc(aes)
6. ecb(aes)
7. cbc(des3_ede)
8. ecb(des3_ede)
9. cbc(des)

10. ecb(des)

There are also additional kernel modules optimised for ARM:

1. sha1_arm
2. aes_arm

fio benchmark

CESA 4kB:

```
   READ: bw=479KiB/s (491kB/s), 479KiB/s-479KiB/s (491kB/s-491kB/s),
io=384MiB (403MB), run=820117-820117msec
  WRITE: bw=160KiB/s (164kB/s), 160KiB/s-160KiB/s (164kB/s-164kB/s),
io=128MiB (134MB), run=820117-820117msec
```

CESA 64kB:

```
   READ: bw=5663KiB/s (5799kB/s), 5663KiB/s-5663KiB/s (5799kB/s-5799kB/s),
io=382MiB (400MB), run=69037-69037msec
  WRITE: bw=1931KiB/s (1977kB/s), 1931KiB/s-1931KiB/s (1977kB/s-1977kB/s),
io=130MiB (137MB), run=69037-69037msec
```

ARM 4kB:

```
   READ: bw=478KiB/s (490kB/s), 478KiB/s-478KiB/s (490kB/s-490kB/s),
io=384MiB (403MB), run=822072-822072msec
  WRITE: bw=159KiB/s (163kB/s), 159KiB/s-159KiB/s (163kB/s-163kB/s),
io=128MiB (134MB), run=822072-822072msec
```

ARM 64kB:

```
   READ: bw=5622KiB/s (5757kB/s), 5622KiB/s-5622KiB/s (5757kB/s-5757kB/s),
io=382MiB (400MB), run=69540-69540msec
  WRITE: bw=1917KiB/s (1963kB/s), 1917KiB/s-1917KiB/s (1963kB/s-1963kB/s),
io=130MiB (137MB), run=69540-69540msec
```

Benchmark

```
cryptsetup benchmark
```

Algorithm	Key	Encryption	Decryption	accel	kernel
aes-cbc	128b	12.8 MiB/s	13.4 MiB/s		3.18
		13.4 MiB/s	14.1 MiB/s	arm	3.18
		19.7 MiB/s	**20.2 MiB/s**	mv_cesa	3.18
		34,9 MiB/s	**36.2 MiB/s**	marvell_cesa	5.8
serpent-cbc	128b	11.1 MiB/s	11.6 MiB/s		3.18

Algorithm	Key	Encryption	Decryption	accel	kernel
twofish-cbc	128b	13.0 MiB/s	13.4 MiB/s		3.18
aes-cbc	256b	10.1 MiB/s	10.5 MiB/s		3.18
		11.0 MiB/s	11.4 MiB/s	arm	3.18
		18.9 MiB/s	**19.2 MiB/s**	mv_cesa	3.18
		32.0 MiB/s	**33.1 MiB/s**	marvell_cesa	5.8
serpent-cbc	256b	11.1 MiB/s	11.6 MiB/s		3.18
twofish-cbc	256b	13.0 MiB/s	13.4 MiB/s		3.18
aes-xts	256b	13.1 MiB/s	13.3 MiB/s		3.18
		14.6 MiB/s	14.7 MiB/s	arm	3.18
		23.6 MiB/s	**22.5 MiB/s**	marvell_cesa	5.8
serpent-xts	256b	11.5 MiB/s	11.6 MiB/s		3.18
twofish-xts	256b	13.4 MiB/s	13.2 MiB/s		3.18
aes-xts	512b	10.2 MiB/s	10.4 MiB/s		3.18
		11.4 MiB/s	11.8 MiB/s	arm	3.18
		22.5 MiB/s	**23.1 MiB/s**	marvell_cesa	5.8
serpent-xts	512b	11.5 MiB/s	11.6 MiB/s		3.18
twofish-xts	512b	13.4 MiB/s	13.2 MiB/s		3.18

Ciphers benchmark

Each cipher was tested with following steps:

- *luksFormat /dev/sda5*
- *luksOpen /dev/sda5 sda5*
- benchmarks described in table below on /dev/mapper/sda5
- create ext4fs on /dev/mapper/sda5
- the same benchmarks but on mounted ext4 (writing/reading from file).

test	command line	description
hdparm	*hdparm -t /dev/...*	Buffered read test
WR	*dd bs=16M count=128*	Normal buffered transfer, but with sync before exit
WR S		
WR DS		
RD		

REMARKS:

1. For XTS, only half of key is used, so for 128b cipher I need to specify -s 256.
2. Ext4 by default was created with lazy_init, to speed up creation process, but it can make impact on tests.
3. Before each test, flush by *sync && echo 3 > .../drop_caches* was issued.

128b key											
		Block device					**EXT4**				
	acc	**hdparm**	**WR**	**WR S**	**WR DS**	**RD**	**WR**	**WR S**	**WR DS**	**RD**	
cbc-plain	HW	8.82	7.0	6.1	7.3	9.2	8.0	5.5	5.8	9.3	
	SW	11.80	8.2	7.4	8.7	12.40	9.5	6.2	6.4	12.40	

128b key											
		Block device					EXT4				
	acc	hdparm	WR	WR S	WR DS	RD	WR	WR S	WR DS	RD	
	ARM	12.76	8.9	7.2	9.2	13.60	10.2	6.4	6.4	13.60	*
cbc-plain64	HW	8.79	6.9	6.1	7.5	9.2	7.9	5.3	5.6	9.0	
	SW	11.83	8.2	7.4	9.2	12.40	9.5	6.2	6.6	12.40	
	ARM	12.73	8.9	7.2	9.3	13.60	10.2	6.2	6.1	13.60	*
cbc-essiv :sha256	HW	7.7	6.2	5.5	6.9	8.1	7.2	5.2	5.2	8.1	
	SW	9.7	7.8	6.9	8.7	11.40	9.1	6.2	6.5	11.40	
	ARM	12.36	8.7	7.0	9.1	13.20	9.9	6.3	6.2	13.20	*
xts-plain	SW	11.29	8.2	7.4	8.7	11.80	9.5	6.1	6.5	11.90	
	ARM	12.79	9.3	7.5	10.1	13.60	10.6	6.3	5.9	13.70	*
xts-plain64	SW	11.27	8.2	7.4	8.7	11.80	9.5	6.2	6.5	11.70	
	ARM	12.84	9.3	7.5	10.2	13.70	10.6	6.4	6.1	13.70	*
xts-essiv :sha256	SW	10.30	7.9	7.2	8.7	11.10	9.1	6.1	6.5	11.10	
	ARM	12.40	9.1	7.5	9.3	13.20	10.4	6.3	6.1	13.30	*
256b key											
		Block device					EXT4				
	acc	hdparm	WR	WR S	WR DS	RD	WR	WR S	WR DS	RD	
cbc-plain	HW	8.43	6.7	6.1	7.5	8.9	7.7	5.5	5.7	8.9	
	SW	9.17	6.7	6.1	7.4	9.6	7.7	5.5	5.8	9.6	
	ARM	10.32	7.6	6.3	7.9	10.80	8.5	5.5	6.0	10.80	*
cbc-plain64	HW	8.44	6.7	6.1	7.5	8.9	7.7	5.5	5.7	8.8	
	SW	9.15	6.8	6.1	7.5	9.5	7.6	5.5	5.8	9.7	
	ARM	10.24	7.6	6.2	7.8	10.70	8.4	5.1	5.5	10.00	*
cbc-essiv :sha256	HW	7.47	6.0	5.5	6.5	7.8	6.9	5.0	5.2	7.8	
	SW	8.59	6.7	6.1	7.5	9.0	7.5	5.3	5.5	8.9	
	ARM	9.83	7.5	6.2	7.9	10.50	8.3	5.5	5.7	10.60	*
xts-plain	SW	8.70	6.8	6.1	7.5	9.1	7.6	5.5	5.6	9.2	
	ARM	10.09	7.9	6.6	8.5	10.7	8.8	5.2	5.6	10.80	*
xts-plain64	SW	8.70	6.8	6.1	7.5	9.2	7.6	5.5	5.6	9.2	
	ARM	10.14	7.9	6.6	8.4	10.80	8.8	5.4	5.7	10.80	*
xts-essiv :sha256	SW	8.37	6.7	6.1	7.0	8.8	7.3	5.1	5.4	8.4	
	ARM	9.94	7.7	6.3	7.9	10.40	8.5	4.9	5.2	9.7	
without encryption											
		Block device					EXT4				
	acc	hdparm	WR	WR S	WR DS	RD	WR	WR S	WR DS	RD	
/dev/sda5		137	91	33.7	51.7	149	69	13	15	149	

file copy benchmark

Copy using *dd if=src_file of=dst_file conv=fsync*

“It will synchronize output data and metadata just before finishing”

128b key				
	acc	**WR S**	**RD**	
aes-cbc-plain64	HW	5.8	8.1	
	SW	6.4	10.60	
	ARM	6.8	12.00	*
twofish-cbc-plain64	SW	6.5	10.60	
aes-cbc-essiv:sha256	HW	5.4	7.1	
	SW	6.3	10.30	
	ARM	6.6	11.10	
twofish-cbc-essiv:sha256	SW	6.5	10.70	
aes-xts-plain64	SW	6.4	10.20	
	ARM	7.0	12.10	*
twofish-xts-plain64	SW	6.6	11.00	
twofish-xts-essiv:sha256	SW	6.4	10.50	
256b key				
	acc	**WR S**	**RD**	
aes-cbc-plain64	HW	5.8	8.3	
	SW	5.5	8.4	
	ARM	5.9	9.5	*
twofish-cbc-plain64	SW	6.6	11.00	*
aes-cbc-essiv:sha256	HW	5.5	7.3	
	SW	5.4	8.0	
	ARM	5.9	9.6	*
twofish-cbc-essiv:sha256	SW	6.5	10.70	*
aes-xts-plain64	SW	5.5	8.2	
	ARM	6.1	9.4	*
twofish-xts-plain64	SW	6.6	10.90	*
twofish-xts-essiv:sha256	SW	6.3	10.10	*

loaded CPU benchmark

Comparison SW & HW with loaded system

```
stress -v -c 1
```

		Block device					**EXT4**				
	acc	**hdparm**	**WR**	**WR S**	**WR DS**	**RD**	**WR**	**WR S**	**WR DS**	**RD**	
cbc-plain-128	HW	4.71	3.9	3.6	3.8	4.9	4.1	3.2	3.4	4.9	
	SW	6.13	4.4	3.9	5.3	6.5	5.0	4.0	4.0	6.4	
	ARM	6.64	4.8	4.2	5.4	7.0	5.3	4.0	4.2	7.0	*
cbc-plain-256	HW	4.68	3.8	3.4	3.9	4.9	4.0	3.2	3.2	4.9	
	SW	4.73	3.6	3.4	4.0	5.0	4.0	3.2	3.3	5.0	
	ARM	5.31	4.1	3.6	4.4	5.6	4.4	3.4	3.6	5.6	

Twofish cipher

(SW only)

		Block device					EXT4			
	key	**hdparm**	**WR**	**WR S**	**WR DS**	**RD**	**WR**	**WR S**	**WR DS**	**RD**
cbc-plain	128	11.80	8.4	7.4	9.5	12.4	9.6	6.0	6.1	11.5
cbc-essiv:sha256	128	11.35	8.2	7.4	8.7	11.9	9.5	6.2	6.5	11.9
xts-plain	128	11.61	8.4	7.4	9.4	12.2	9.5	6.2	6.6	12.3
xts-essiv:sha256	128	11.06	8.0	7.4	8.7	11.6	9.1	6.2	6.5	11.7
cbc-plain	256	11.82	8.4	7.4	9.5	12.4	9.7	6.5	6.6	12.4
cbc-essiv:sha256	256	11.34	8.2	7.4	8.7	11.9	9.5	6.2	6.6	12.0
xts-plain	256	11.64	8.4	7.4	9.4	12.2	9.6	6.2	6.6	12.3
xts-essiv:sha256	256	11.04	8.0	7.4	8.7	11.6	9.3	6.2	6.5	11.7

SSH performance

Enable low complexity ciphers if device is used locally.

```
ssh -Q cipher localhost | paste -d , -s
```

/etc/ssh/sshd_config

```
# enable all ciphers!
# obtained with ssh -Q cipher localhost | paste -d , -s
Ciphers 3des-cbc,blowfish-cbc,cast128-
cbc,arcfour,arcfour128,arcfour256,aes128-cbc,aes192-cbc,aes256-
cbc,rijndael-cbc@lysator.liu.se,aes128-ctr,aes192-ctr,aes256-
ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com,chacha20-
poly1305@openssh.com
```

cmd	performance	time	Kernel 5.8
(default)	3.1MB/s		
3des-cbc	1.2MB/s	1m28	67.9 MB/s
blowfish-cbc	3.3MB/s	0m30	245.7 MB/s
cast128-cbc	2.9MB/s	0m34	248.8 MB/s
arcfour	4.2MB/s	0m24	425.5 MB/s
arcfour128	-- | --	395.3 MB/s	
arcfour256	4.6MB/s	0m22	425.5 MB/s
aes128-cbc	2.8MB/s	0m37	228.8 MB/s
aes192-cbc	2.9MB/s	0m34	211.4 MB/s
aes256-cbc	2.5MB/s	0m40	192.3 MB/s
rijndael-cbc@lysator.liu.se	2.8MB/s	0m36	192.3 MB/s
aes128-ctr	2.9MB/s	0m35	223.2 MB/s

cmd	performance	time	Kernel 5.8
aes192-ctr	2.9MB/s	0m35	202.8 MB/s
aes256-ctr	2.9MB/s	0m40	191.6 MB/s
aes128-gcm@openssh.com	2.6MB/s	0m39	170.7 MB/s
aes256-gcm@openssh.com	2.2MB/s	0m47	151.7 MB/s
chacha20-poly1305@openssh.com	3.2MB/s	0m32	268.8 MB/s

fsck performance

- Kernel 5.8 CESA: user 1m32,738s sys 0m9,904s
- Kernel 5.8 ARM: user 1m32,642s sys 0m9,177s

Rsync with SSH tunnell performance

When rsyncing huge file, to use delta transfers, rsync needs to scan file to make checksums.

Kernel 5.8 ARM

```
    134,113,800   1%    7.99MB/s    0:16:11
    142,909,200   1%    8.04MB/s    0:16:03
    151,768,800   1%    8.33MB/s    0:15:29
    159,151,800   1%    8.11MB/s    0:15:53
    175,715,400   2%    7.74MB/s    0:16:36
    192,664,200   2%    7.86MB/s    0:16:20
    200,111,400   2%    7.61MB/s    0:16:50
    217,060,200   2%    7.63MB/s    0:16:45
    234,715,200   2%    8.19MB/s    0:15:34
    243,382,200   3%    8.17MB/s    0:15:37
    250,572,600   3%    7.94MB/s    0:16:02
    259,432,200   3%    7.98MB/s    0:15:57
    268,227,600   3%    7.98MB/s    0:15:55
    334,674,600   4%    8.19MB/s    0:15:24

%Cpu(s): 50,0 us, 50,0 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  0,0 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
32375 root      20   0       0      0      0 R  63,2   0,0   1:14.38
kworker/u2:3-kcryptd/253:0
16281 root      20   0   14808   6600   3028 R  36,8   2,7   0:04.58 rsync

%Cpu(s): 30,8 us, 65,4 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  3,8 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
16360 root      20   0       0      0      0 R  57,1   0,0   0:00.35
```

```
kworker/u2:1+kcryptd/253:0
16281 root      20   0   14808   6604   3028 D  28,6   2,7   0:06.54 rsync
%Cpu(s): 29,0 us, 71,0 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  0,0 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
16360 root      20   0       0      0      0 R  58,3   0,0   0:01.74
kworker/u2:1+kcryptd/253:0
16281 root      20   0   14808   6604   3028 R  29,2   2,7   0:07.24 rsync

%Cpu(s): 26,9 us, 73,1 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  0,0 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
31416 root      20   0       0      0      0 R  50,0   0,0   0:06.85
kworker/u2:2+kcryptd/253:0
16281 root      20   0   14808   6608   3028 D  27,3   2,7   0:08.21 rsync

%Cpu(s): 33,3 us, 66,7 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  0,0 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
16360 root      20   0       0      0      0 R  63,6   0,0   0:08.34
kworker/u2:1+kcryptd/253:0
16281 root      20   0   14808   6612   3028 R  27,3   2,7   0:10.61 rsync
```

Kernel 5.8 CESA

```
    178,540,200   2%    7.46MB/s    0:17:15
    185,987,400   2%    7.35MB/s    0:17:29
    209,613,000   2%    7.39MB/s    0:17:20
    271,758,600   3%    7.30MB/s    0:17:24
    287,680,200   3%    7.31MB/s    0:17:20
    334,995,600   4%    7.30MB/s    0:17:16
    358,300,200   4%    7.41MB/s    0:16:57
    761,026,800   9%    7.07MB/s    0:16:51
    769,051,800   9%    7.15MB/s    0:16:38
    777,397,800   9%    7.40MB/s    0:16:03
%Cpu(s): 31,3 us, 68,1 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  0,6 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
29705 root      20   0       0      0      0 R  34,4   0,0   1:40.62
kworker/u2:0+kcryptd/253:0
30062 root     -51   0       0      0      0 S  32,5   0,0   1:10.38 irq/29-
f1030000
32091 root      20   0   14808   6516   2908 D  28,7   2,7   0:24.76 rsync
%Cpu(s): 36,4 us, 63,6 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  0,0 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
29705 root      20   0       0      0      0 R  28,0   0,0   1:41.40
kworker/u2:0+kcryptd/253:0
30062 root     -51   0       0      0      0 S  28,0   0,0   1:11.12 irq/29-
```

```
f1030000
32091 root      20   0   14808   6516   2908 D  24,0   2,7   0:25.41 rsync

%Cpu(s): 26,5 us, 73,5 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  0,0 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
29705 root      20   0       0      0      0 R  30,8   0,0   1:42.31
kworker/u2:0+kcryptd/253:0
30062 root     -51   0       0      0      0 S  26,9   0,0   1:11.98 irq/29-
f1030000
32091 root      20   0   14808   6520   2908 D  19,2   2,7   0:26.17 rsync

%Cpu(s): 29,0 us, 71,0 sy,  0,0 ni,  0,0 id,  0,0 wa,  0,0 hi,  0,0 si,  0,0
st
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
29705 root      20   0       0      0      0 R  25,9   0,0   1:42.76
kworker/u2:0+kcryptd/253:0
30062 root     -51   0       0      0      0 S  22,2   0,0   1:12.41 irq/29-
f1030000
32091 root      20   0   14808   6520   2908 D  18,5   2,7   0:26.54 rsync
```