Introducing…

By Olivier COCHARD-LABBÉ,

FreeNAS author and project manager until april 2008

static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init,ma

Plan

- What are: a NAS, an appliance and FreeNAS ?
- Features
- Inside FreeNAS
- Example: Encrypted Software RAID 5
- History and Awards

```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma
```

Definition: NAS and Appliance

NAS (Network Attached Storage):

File level data storage using file-level protocol as CIFS, NFS, AFP,etc…

Appliance:

“A computer appliance is generally a separate and discrete hardware component specifically designed to provide a specific computing resource”

Source: Wikipedia

static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init,ma

What is FreeNAS ?

1) Embedded OS specialized in NAS:
   - Based on the m0n0wall “framework”
   - Upgraded to latest FreeBSD stable with the m0n0wall firewall features replaced by NAS features

2) An exercise for discovering FreeBSD, PHP and OSS managment too :-)


```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init,ma
```

Why using FreeBSD ?

- First try with BusyBox
  - I didn't understand how it works (never try to compile a software before)
- Second try using m0n0wall Developers' Handbook
  - Very easy-to-follow handbook for build a m0n0wall from scratch: No source compilation, only file copy
  - Include the WebGUI files and modified rc scripts for appliance (XML config file, RAM drive)


```
static void sync_icache_030(void *address,unsigned int len){unsigned int off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma
```

Features

- Hardware: all supported by FreeBSD
- Minimum: 256Mb RAM and 128Mb flash
- FreeBSD GEOM: RAID, JBOD, Encryption (using cryptographic hardware if available) and ZFS
- Filesystem: UFS, but you can use ext2/3, ntfs or even FAT for data migration
- Integration: LDAP and AD
- Monitoring: SNMP, email report, UPS

static void sync_icache_030(addr_t address,int len){int l,off;char *p;uint32 cacr;off=(uint32)((int)address&(CACHELINE-1));
len+=off;l=len;p=(char*)(address-off);cacr=get_cacr();if(cacr&0x00000004){/*data cache enabled*/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma

Features

- Services:
  - CIFS, NFS, AFP
  - FTP, TFTP, SCP
  - RSYNC, Unison
  - UpnP
  - I-SCSI target
- User Requested Services:
  - Bittorrent client
  - Webserver
  - Itunes/DAAP

static void sync_icache_[illegible](void *address, uint32_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void [illegible](void)
{long_page_directory_entry entry,*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init,ma

Inside FreeNAS

- Light FreeBSD system (about 80MB)
  - Only useful binary and libs are kept
  - Add softwares: samba, ntfs-3g, lighttpd, etc…
- All configuration parameters stored in a XML file
- All rc scripts re-wrote:
  - They re-generate the config files from the xml file

```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma
```

Inside FreeNAS

- System partition:
  - zipped kernel
  - zipped root filesystem (in one file)
  - config file
- Start-up sequence:
  1) Unzip the kernel in RAM
  2) Unzip the root filesystem in RAM
  3) Start rc scripts


```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
```

Inside FreeNAS

- RC scripts:
  - Search XML config file (scan all media)
  - Generate services config file
- Where are the FreeNAS files:
  - /etc/rc* : Customized startup scripts
  - /etc/rc.d/ : Customized services scripts
  - /etc/inc : Global PHP functions
  - /usr/local/www/ : Web GUI pages


```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
```

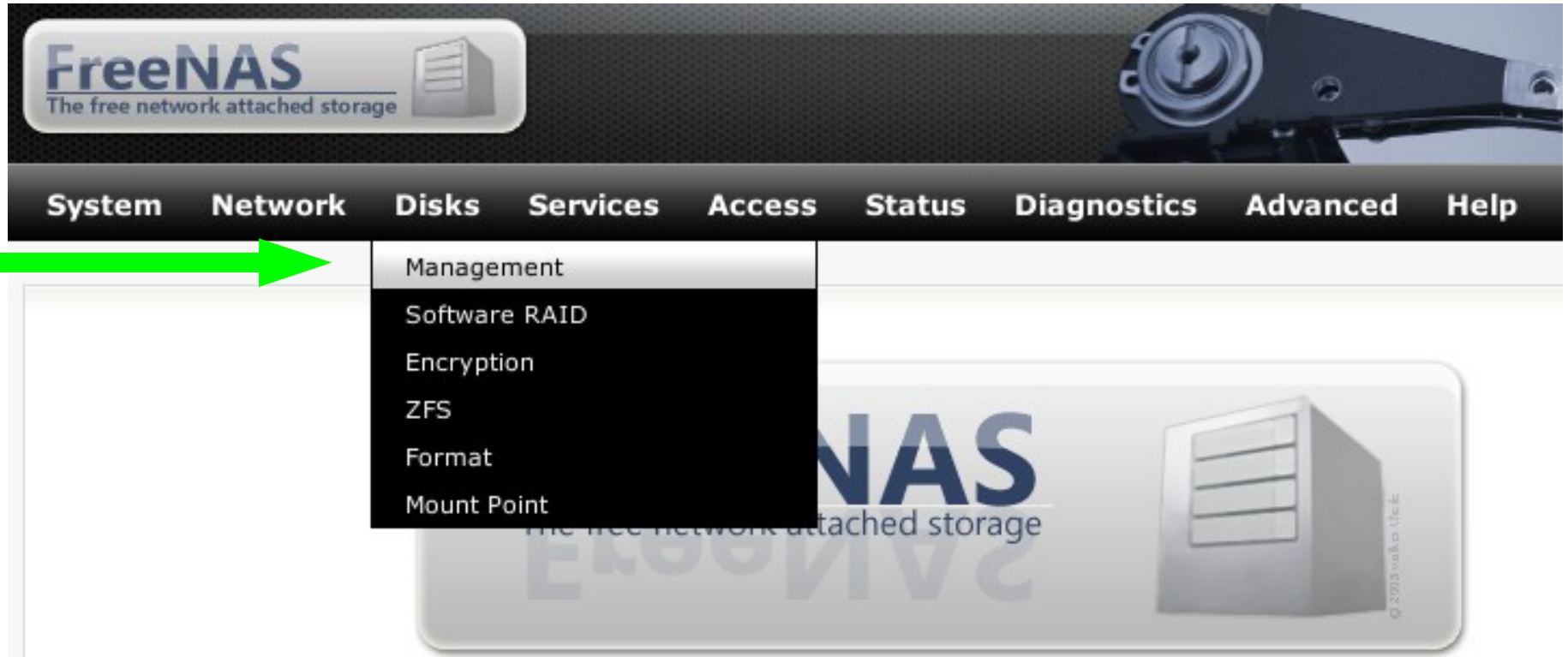
Example: Encrypted software RAID

1) Add your disks
2) Create the software RAID 5 volume
3) Create the encrypted volume by setting your passphrase
4) Format the volume
5) Mount the volume

The passphrase is needed after each reboot


```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init,ma
```

Example: Encrypted software RAID

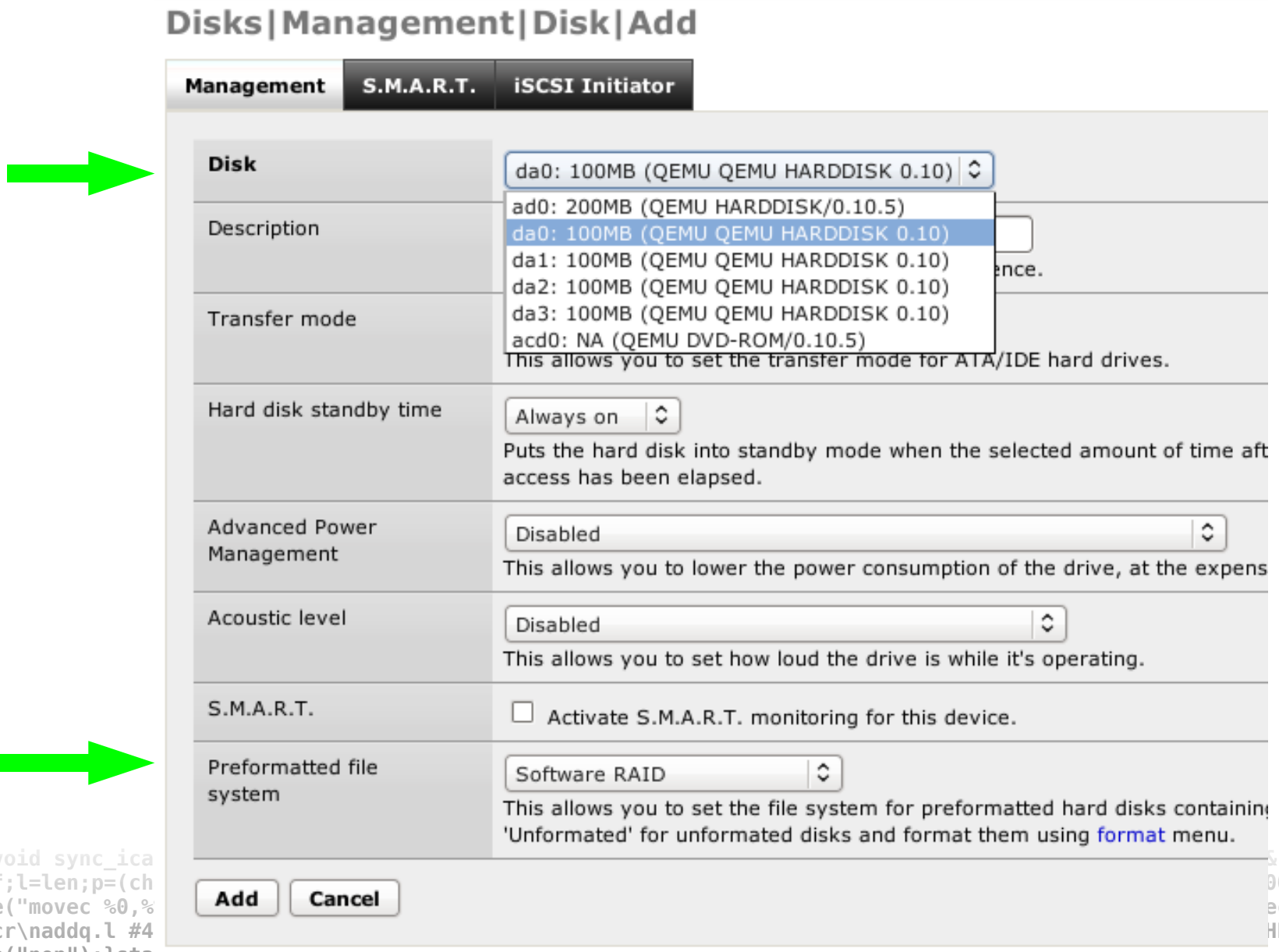



```
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
```

Example: Encrypted software RAID

static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma

Example: Encrypted software RAID

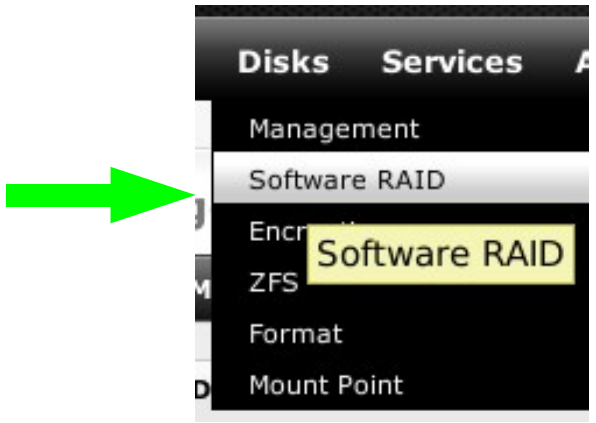


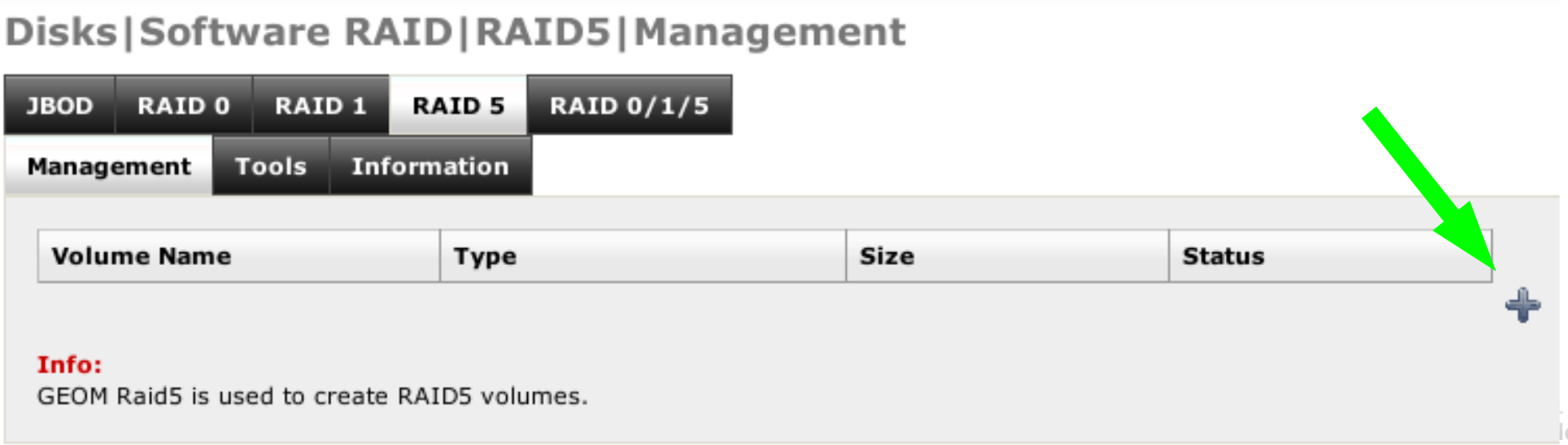

```
static void sync_ica                                                    &(CACHELINE-1);
len+=off;l=len;p=(ch                                                    0000004;/**/do{asm
volatile("movec %0,%                                                    ec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4                                                    HELINE)>0);asm
volatile("nop");}sta
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init,ma
```

Example: Encrypted software RAID


```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma
```

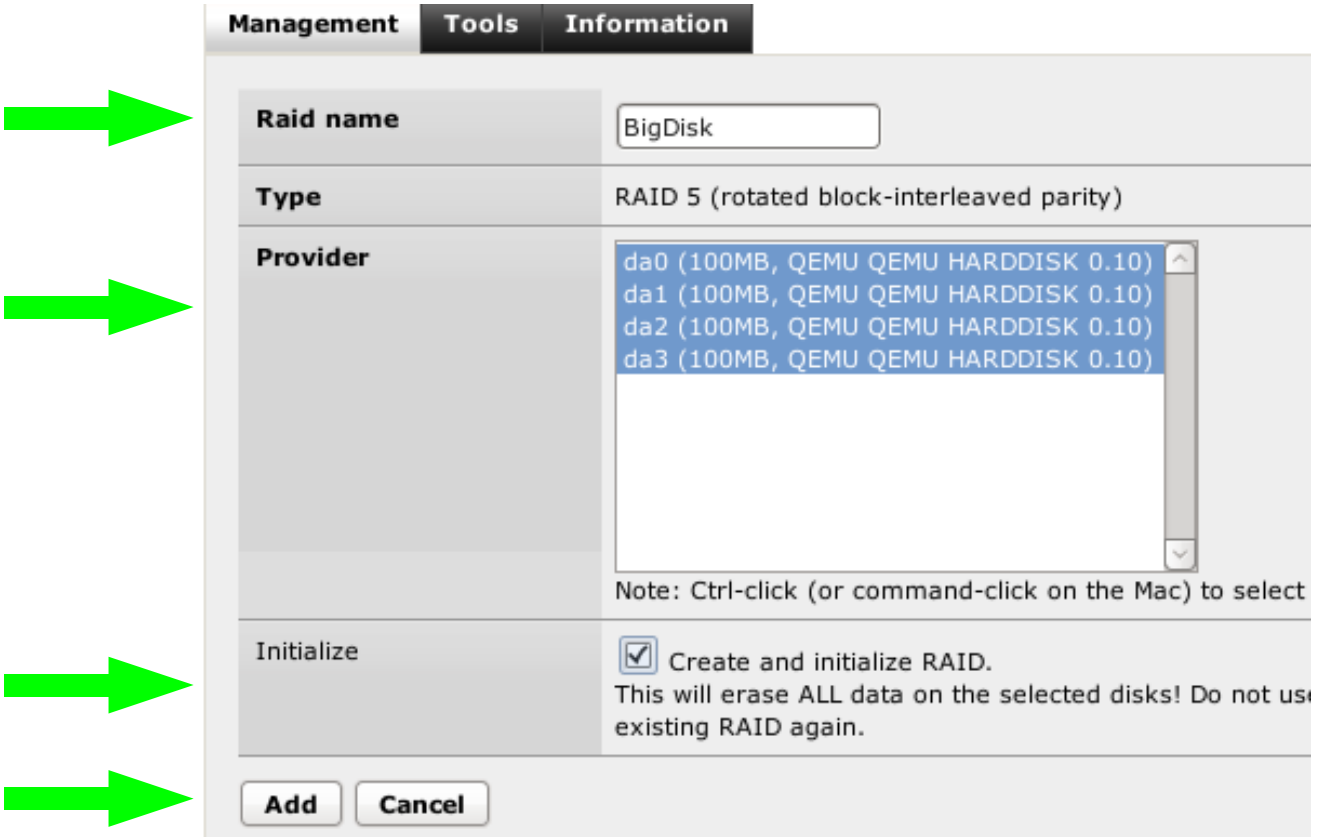
Example: Encrypted software RAID

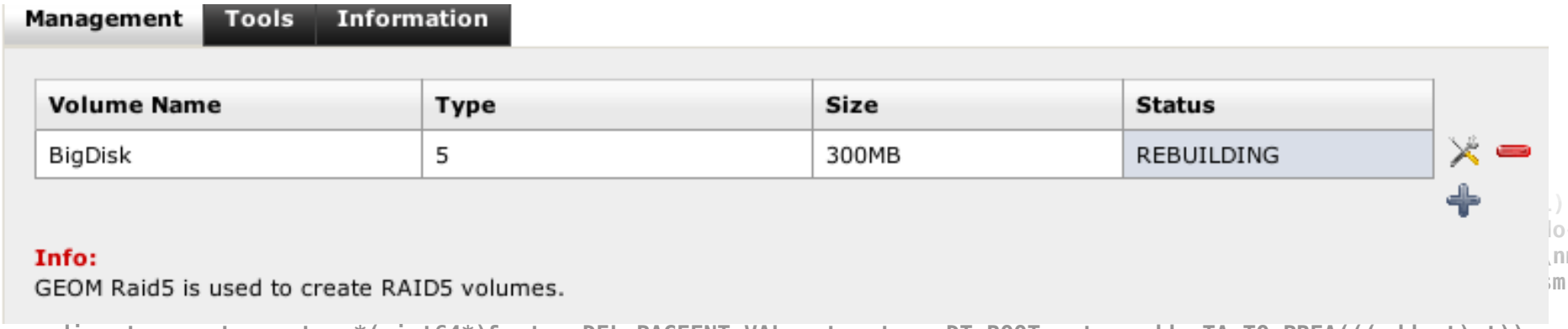




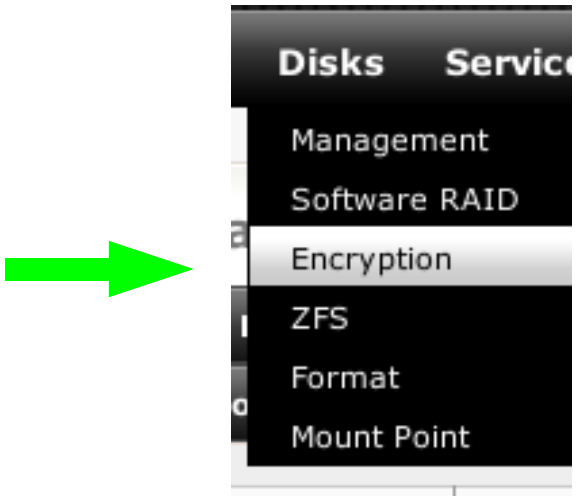

```
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
```

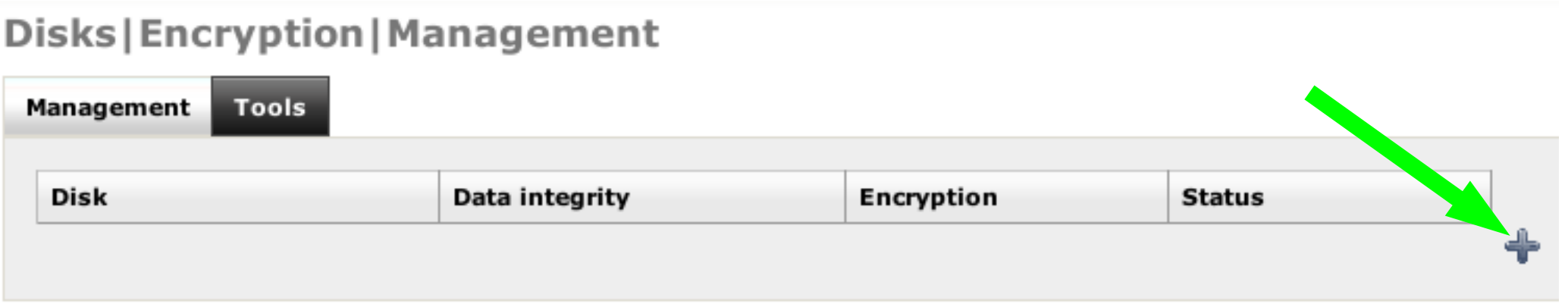
Example: Encrypted software RAID





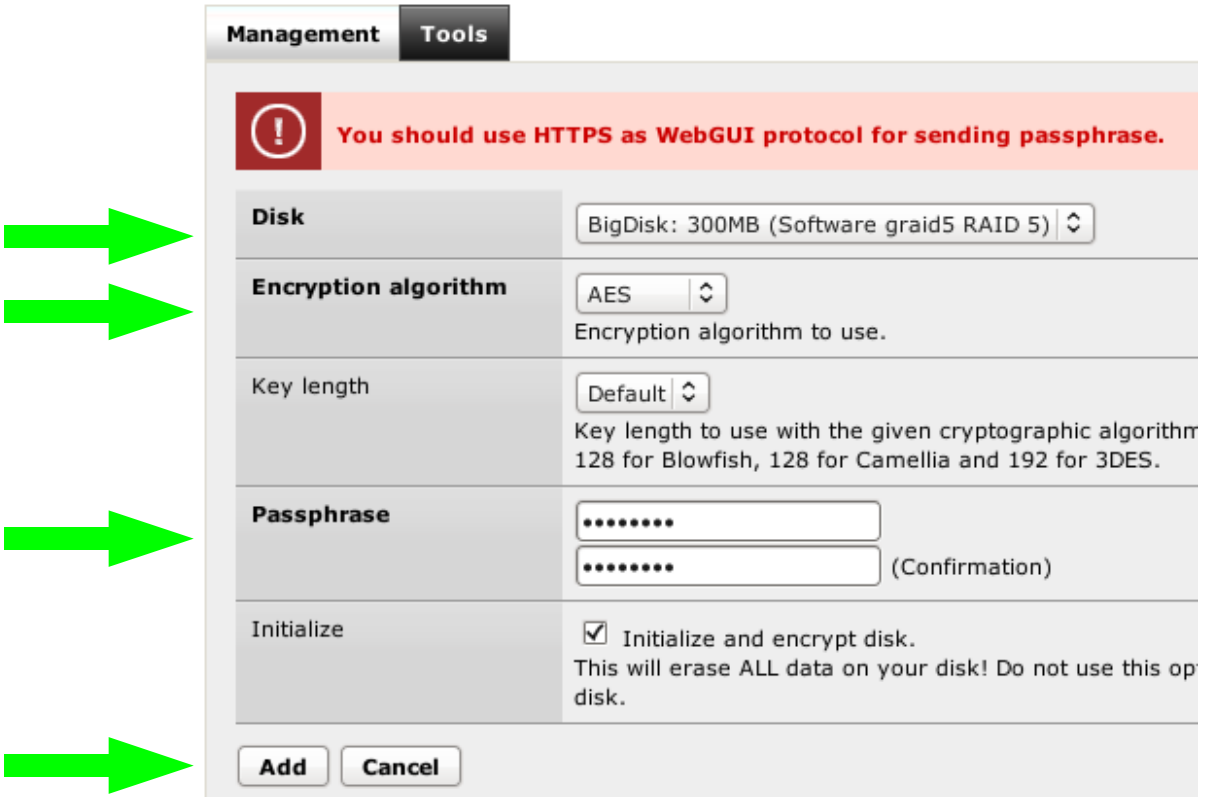
Example: Encrypted software RAID

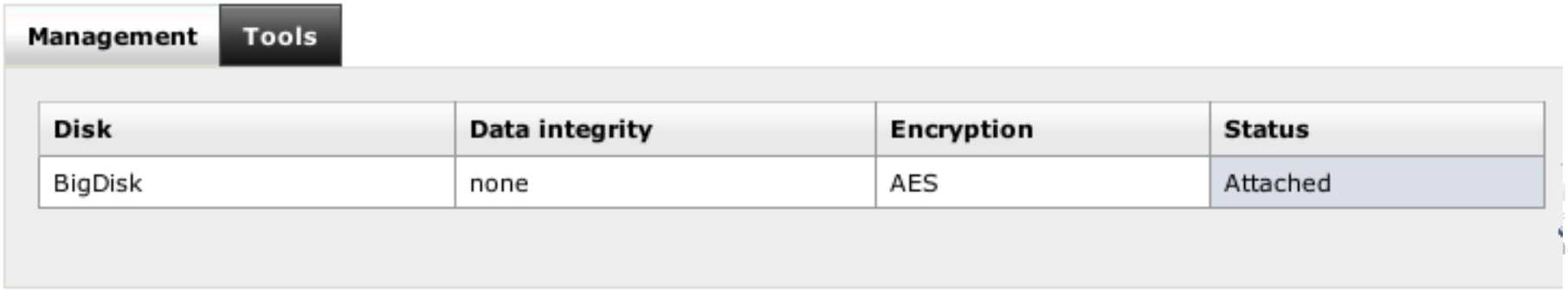





```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init,ma
```

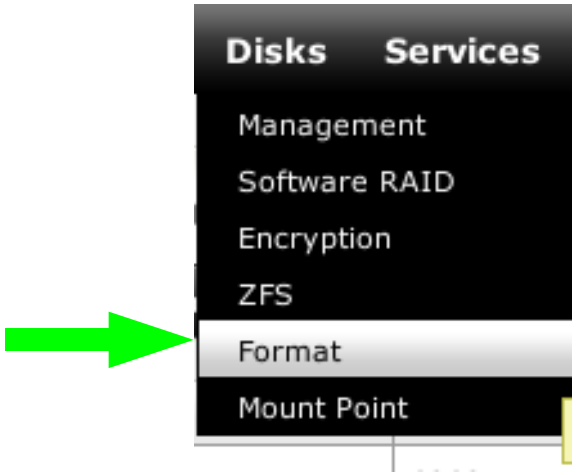
Example: Encrypted software RAID

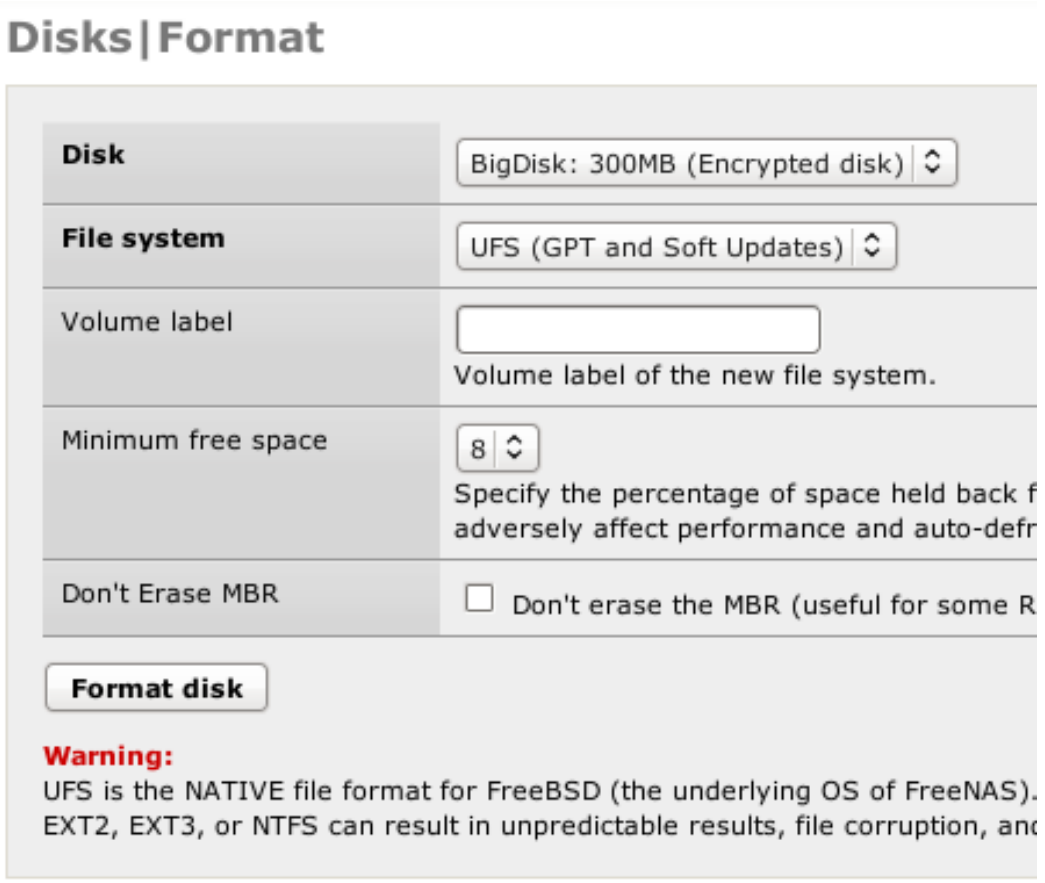




{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%

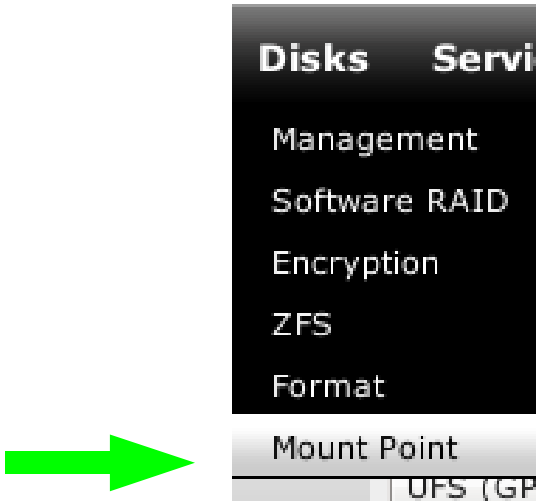
Example: Encrypted software RAID

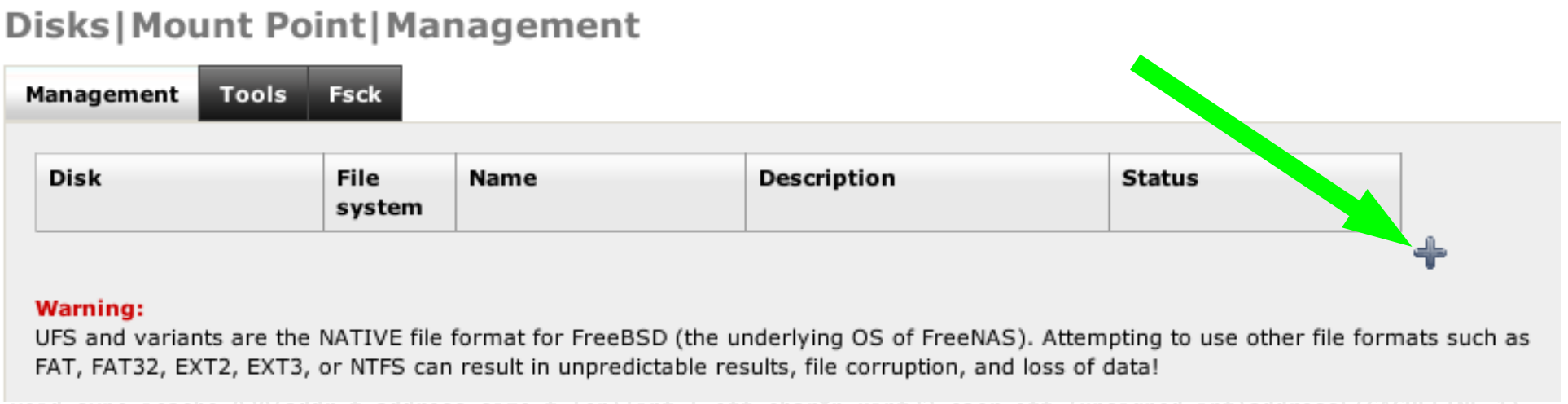





```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma
```

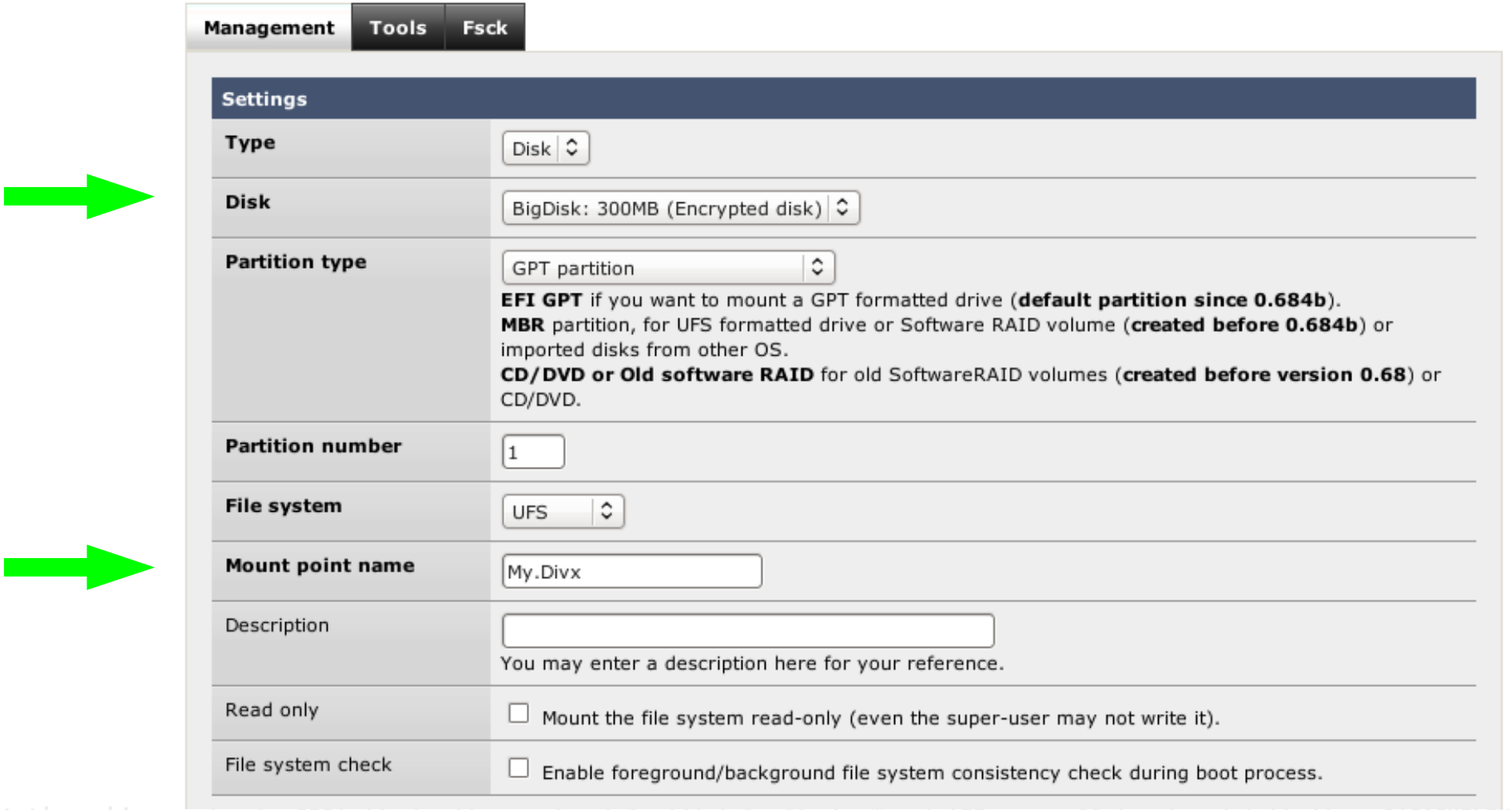
Example: Encrypted software RAID





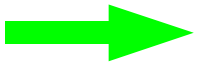

```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma
```

Example: Encrypted software RAID




```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init,ma
```

Example: Encrypted software RAID



Access Restrictions

Owner	root			
Group	wheel			
Mode		**Read**	**Write**	**Execute**
	Owner	☑	☑	☑
	Group	☑	☑	☑
	Others	☑	☑	☑

Add **Cancel**

Warning:
You can't mount the partition '/dev/ad0s1' where the config file is stored.
UFS and variants are the NATIVE file format for FreeBSD (the underlying OS of FreeNAS). Attempting to use other file formats such as FAT, FAT32, EXT2, EXT3, or NTFS can result in unpredictable results, file corruption, and loss of data!

Disk	File system	Name	Description	Status
/dev/raid5 /BigDisk.elip1	ufs	My.Divx		OK


```
static void sync_icache_030(addr_t address,size_t len){int l,off;char*p;uint32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=off;l=len;p=(char*)address-off;asm volatile("nop");asm volatile("movec %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
%srp\n"::"a"((uint64*)&entry));}struct m68k_vm_ops m68030_vm_ops={ m68k_translation_map_get_pgdir,m68k_vm_translation_map_init_ma
```

History and Awards

- Oct 05 First release available
- Aug 06 VMware: Ultimate Virtual Appliance Challenge
- Jan 07 Sourceforge: Project of the month
- Sep 07 InfoWorld: Bossie Award
- Apr 08 Volker Theile became project manager
- Aug 08 InfoWorld: Bossie Award

History and Awards

- June 2009: FreeNAS is a finalist for Sourceforge Community Choice Award 09 in category:
  - Best Project
  - Most Likely to Change the Way You Do Everything

You can vote here:

http://sf.net/community/cca09/

```
static [illegible] 32 cacr;off=(unsigned int)address&(CACHELINE-1);
len+=[illegible] %%cacr,%0":"=r"(cacr):);cacr|=0x00000004;/**/do{asm
volatile("movec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec
%1,%%cacr\naddq.l #4,%0\nmovec %0,%%caar\nmovec %1,%%cacr\n"::"r"(p),"r"(cacr));p+= CACHELINE;} while((l-=CACHELINE)>0);asm
volatile("nop");}static void set_pgdir(void*rt)
{long_page_directory_entry entry;*(uint64*)&entry=DFL_PAGEENT_VAL;entry.type=DT_ROOT;entry.addr=TA_TO_PREA(((addr_t)rt));asm vola
tile("pmove (%0),%%srp\npmove (%0),%
```