

**Full encrypted Backup server to (r)sync with the primary server**

I promised to write further about my project, and here I'm. First I needed to setup the Primary Server, you can read how in this article: [Setup FreeBSD Server with full HDD encryption](#). Then I had to insert a second Power source as you can read in the article: [Use a Compaq 200 Watt Power Supply \(PSU\) as a second power source](#) to power 9 harddisks in my system (5x IDE and 4x Sata).

Now I'am finally ready to setup my beast!

This Manual will destroy all your files on your harddisk, if you fondle around with hardware or Software you are responsible! You have been warned!!

1

### ***Connect all the harddrive's and a cdrom drive***

Connect all the harddrive's and a cdrom drive to your system, startup the system and boot from the [FreeBSD 8.1](#) DVD.

2

### ***Choose standard***

Choose a standard installation, and the choose user from the menu.

3

### ***Configure your Boot Harddisk***

Configure your BOOT harddrive, the name is usually: ad0, Make a slice of 12 GB. If your boot harddisk is over 40GB, you can change this value to a higher one, but dont use all the space in one slice.

4

### ***Quit and make the mount points***

Choose {Q} uit and choose OK to make the mount points. An example:

```
1: ad0s1a - / - 2000MB - UFS2 - Y
2: ad0s1b - swap - 1000MB - SWAP -
3: ad0s1d - /var - 2000MB - UFS2+S - Y
4: ad0s1e - /tmp - 1000MB - UFS2+S - Y
5: ad0s1f - /usr - Rest - UFS2+S - Y
```

If you have a bigger hdd, use the following:

```
1: ad0s1a - / - 5000MB - UFS2 - Y
2: ad0s1b - swap - 2000MB - SWAP -
3: ad0s1d - /var - 5000MB - UFS2+S - Y
4: ad0s1e - /tmp - 1000MB - UFS2+S - Y
```

5: ad0s1f - /usr - Rest - UFS2+S - Y

5

### ***Choose BootMgr***

Choose {Q}uit and choose Boot Manager (Other than this will give me errors)

Choose {OK}

6

### ***Ports Collection***

Say yes if FreeBSD asks to install the ports collection, choose to install from CD/DVD (The one you inserted in your cdrom-drive).

A Picture of my Monster:

[Backup-Monster.jpg](#) (398 KB) ([File Type Details](#))

My Monster of loch ness with 5x IDE and 4x Sata Hdd



7

### ***Sure to write partitions?***

FreeBSD will ask if you are Sure to write all the configured file systems, answer {Yes}!

Please wait until the installation is finished!

8

## ***Congratulations! You now have FreeBSD***

Congratulations! You now have FreeBSD installed on your system, choose {OK}

9

## ***Configure Ethernet...***

Configure Ethernet or SLIP/PPP network devices? Answer: {Yes}

On my machine I use a separate network card, onboard Network devices seem to always give me headache. I Choose fxp1

10

## ***A Few Network questions***

IPv6 -> Answer: No

DHCP -> Answer: No.

Separate screen to enter the LAN credentials.

Host:BSD02

Domain: wayward.nl

IPv4 Gateway:10.30.0.100 (My Router address)

Name Server: 10.30.0.100 (My Router address, if you have a domain controller that provides DNS you can enter it here)

IPv4 Address: 10.30.0.3

Would you like to bring the fxp1 interface up right now?: {Yes}

function as a network gateway?: {No}

Configure inetd and the network.... {No}

Would you like to enable SSH login? {Yes}

Do you want to have anonymous FTP access to this machine? {No}

configure NFS Server {No}

This machine NFS client {No}

Customize your system console settings? {No}

Time Zone: {Yes}

Select local or UTC... {No}

Time Zone Selector: {8} Europe

Countries in Europe: {34} Netherlands  
CEST look reasonable? {Yes}

PS/2, serial or bus Mouse? {No} (FreeBSD picks it up along the way)

FreeBSD package collection, Browse the collection now? {No}

Additional accounts to the system? {No} (The user will not get a home directory when you create it in the install routine, this can be a pain)

Set Root Password: {Ok}

Enter a password twice, and keep this Password different than the password you are going to use on the encrypted part.

Chance to Set any last options? {No}

**Exit the installation.**

Remove the media: {Ok}

System will reboot.

11

### ***Create a encrypted part of the boot hdd***

Type:

1: Sysinstall

Choose: Configure --> Fdisk

In my case the boot hdd is ad0, I choose ad0 (Place an X and then {OK})

If you are confronted with Geometry, I choose {Yes}.

In Fdisk, press {C} and use up the rest of the hdd, press {W},  
Choose BootMgr and then press {Q} to leave.

There is an {X} in front of ad0, choose {OK}

Press {X}, and {Exit Install} to Exit sysinstall.

12

### ***Shutdown the Backup server***

My Backup server has a problem with the RocketRaid card that the computer will always startup, even when I tell him to Power down. Instead I use the following command:

1: shutdown -h NOW

When the system is halted, I pull out the powercord, and then I switch off the [secondary power supply](#).

## ***Preparing temporary hdd***

13

### ***Start from the secondary HDD***

In the BIOS of my Primary server I could say from which IDE drive the system must start, unfortunately the compaq has no option for this.

We need to disconnect the primary hdd and connect a harddisk configured as slave on the IDE controller. Start the system and insert the FreeBSD DVD in the cdrom drive.

14

### ***Secondary Installation Steps***

Start the system from the CDrom and choose:  
**Standard installation.**

A program to partition your harddisk will be started, select {OK}

15

### ***Create a slice***

You will be asked which harddisk you wish to work on, in my case the hdd is called ad1, I choose {ad1}.

Delete any existing Slices with the {D} key.

Create a New Slice and use the full hdd,

Press {Q} to leave this program.

16

### ***Choose Boot Manager***

Since the Compaq BIOS has no option for selected the harddrive we **need** the BootMgr, I choose {BootMgr}.

17

## ***Make partitions***

an {X} is still in front of ad1, select {OK}

Some instructions will be given, select {OK}

Choose {A} and the partitions will be filled in, for the secondary hdd it is not very important to have a good proportioned harddrive since you are going to use it once.

Press {Q} to leave

18

## ***Choose the installation***

Choose: {User} Binaries and doc only.

You will be asked in which language you want documentation. Choose the correct one and then {OK}

Install the ports collection? {No}

Choose {OK}

Install from FreeBSD CD/DVD.

A warning appears that all be overwritten, choose {Yes}

The installation is started, please wait....

19

## ***Answer some questions***

Configure Ethernet or SLIP/PP network devices? {NO}

function as a network gateway? {NO}

configure inetd and the network services that it provides? {NO}

like to enable SSH login? {YES}

Do you want anonymous FTP access? {NO}

NFS Server? {NO}

NFS Client {NO}

customize your system console settings? {NO}

Time Zone? {YES}

CMOS clock set to UTC... {NO}, choose: 8. Europe, Netherlands

CET reasonable? {YES}

PS/2, serial or BUS mouse? {NO}

FreeBSD package collection? {NO}

additional accounts to the system? {YES} --> Add a user, then use {X} Exit.

20

### ***Enter the root password***

Keep this password different then from the encrypted part of the hdd.

Visit general configuration menu for a change to set any last options? {NO}

{X} Exit Install

Are you sure? {Yes}

Be sure to remove the media from the drive {OK}

21

### ***Turn of system when the BIOS screen is visible***

Turn of system when the BIOS screen is visible.

### ***Creating the encrypted part of the hdd***

22

### ***Connect both harddisk drive's on the system***

Connect both harddisk drives to the primary IDE cable, so there is a Master drive (The one you are going to use in the future) and a secondary hdd on the primary IDE cable (The temporary hdd).

23

### ***Choose F5 (Other drive)***

At bootup you will be presented to boot from the hdd:

1: F1Start FreeBSD  
2: F5 Drive 1

3:

choose F5 to switch from Primary hdd to Secondary HDD (On the Primary IDE Cable).  
Then choose F1 to actually boot (Or wait a few seconds)

24

### ***Check the available devices***

Login as root

Go to the devices directory:

```
1: cd /dev/
```

and get the directory dump on your screen:

```
1: ls
```

Look for a drive with **s2** at the end, on my machine the drive is called: *ad0s2*.

25

### ***Create Encrypted Part of HDD***

To make the second Slice of the Boot harddrive encrypted type:

```
1: geli init -b -s 4096 -l 256 /dev/ad0s2
```

You will be asked to enter a passphrase, enter this twice.

*My passphrase is the same as the head server and has 7 words in it, make it hard for another to crack and easy for you to remember. Be sure to use Capital and low characters.*

26

### ***Attach encrypted Slice***

Type:

```
1: geli attach /dev/ad0s2
```

Message will appear:

```
1: GEOM_ELI: Device ad0s2.eli created
2: GEOM_ELI: Encryption: AES-CBC 256
3: GEOM_ELI:      Crypto: software
```

27

## **Create Partitions on the encrypted drive**

To make the necessary partitions/Labels on the encrypted part we are going to use `bsdlabel`:

```
1: bsdlabel -w /dev/ad0s2.eli
```

```
1: bsdlabel -e /dev/ad0s2.eli
```

After the last line you will be presented with an editor, make it look like this:

```
# /dev/ad0s2.eli:
1: 8 partitions:
2: #      size              offset          fstype          [fsize  bsize
3:      bps/cpg]
4: a:    125000              0              4.2BSD          0        0
5: b:    118164             125000         swap            0        0
6: c:    1418759            0              0              unused    0
7:      0      # don't edit
8: d:    309082             243164         4.2BSD          0        0
9: e:    125000             552246         4.2BSD          0        0
   f:      *                677246         4.2BSD          0        0
```

*I=Insert [ESC=end Insert], x remove character.*

**Do not change the letter c!**

I use this setup, because my hdd is 100 GB:

```
# /dev/ad0s2.eli:
1: 8 partitions:
2: #      size              offset          fstype          [fsize  bsize
3:      bps/cpg]
4: a:    500000              0              4.2BSD          0        0
5: b:    472656             500000         swap            0        0
6: c:    ??????            0              unused          0        0
7: # don't edit
8: d:    618164             972656         4.2BSD          0        0
9: e:    250000             1590820        4.2BSD          0        0
   f:      *                1840820        4.2BSD          0        0
```

28

## **Save the labels**

Press once on {ESC}, then type `:w {ENTER}`, and leave `:q {ENTER}`

29

## **Check if the new (encrypted) devices are created**

```
1: cd /dev
```

```
1: ls
```

Are there any **.eli** devices? If Yes, Go ON!

30

### ***Format the encrypted partitions/labels***

```
1: newfs -i 1024 /dev/ad0s2.elia
```

The parameter **-i** will make it possible to write a lot of small files on this partition

We don't need to format the swap partition, so we go on to:

```
1: newfs /dev/ad0s2.elid
```

```
1: newfs /dev/ad0s2.elie
```

With the label mounted as **/usr** it is important to be able to write a lot of small files:

```
1: newfs -i 1024 /dev/ad0s2.elif
```

31

### ***Create the directory for root mountpoint***

```
1: mkdir /fixed
```

Then mount it:

```
1: mount /dev/ad0s2.elia /fixed
```

32

### ***Create all other directory's needed for FreeBSD OS***

```
1: mkdir /fixed/var
```

```
1: mkdir /fixed/tmp
```

```
1: mkdir /fixed/usr
```

33

### ***Mount the encrypted slices***

```
1: mount /dev/ad0s2.elid /fixed/var
```

```
1: mount /dev/ad0s2.elie /fixed/tmp
```

```
1: mount /dev/ad0s2.elif /fixed/usr
```

34

### ***Copy the FreeBSD OS to the encrypted part of the drive***

Prepare the destination location (encrypted part)

```
1: /bin/sh
```

```
1: export DESTDIR=/fixed/
```

```
1: /bin/csh
```

Mount cdrom drive:

```
1: mount /cdrom
```

Change to the correct directory:

```
1: cd /cdrom/8.1-RELEASE/base
```

```
1: ./install.sh
```

You are about to extract the base distribution into **/fixed/** - are you SURE you want to do this over your installed system (y/n)? If **/fixed/** is mentioned, press: {y}

35

### ***Install the kernel***

```
1: cd /cdrom/8.1-RELEASE/kernels
```

```
1: ./install.sh GENERIC
```

36

### ***Install the help pages***

```
1: cd /cdrom/8.1-RELEASE/manpages
```

```
1: ./install.sh
```

```
1: cd /cdrom/8.1-RELEASE/catpages
```

```
1: ./install.sh
```

37

### ***Copy the boot directory to the future boot drive***

First we need to mount the future boot drive:

```
1: mount /dev/ad0s1 /mnt
```

Copy the boot directory to the boot drive:

```
1: cp -Rpv /fixed/boot /mnt
```

38

### ***Speed up the boot process***

To Speed up the boot process we compress a few files:

```
1: cd /mnt/boot/kernel
```

```
1: gzip kernel geom_eli.ko acpi.ko
```

39

### ***Make FreeBSD startup from the encrypted part***

To let FreeBSD boot from the un-encrypted part of the hdd and process the startup from the encrypted part we change the fstab file.

```
1: vi /mnt/etc/fstab
```

Make the Fstab file look like this:

#	Device	Mountpoint	Fstype	Options	Dump
	Pass#				
	/dev/ad0s2.elib	none	swap	sw	0
1:	0				
2:	/dev/ad0s2.elia	/	ufs	rw	1
3:	1				
4:	/dev/ad0s2.elie	/tmp	ufs	rw	2
5:	2				
6:	/dev/ad0s2.elif	/usr	ufs	rw	2
7:	2				
	/dev/ad0s2.elid	/var	ufs	rw	2
	2				
	/dev/acd0	/cdrom	cd9660	ro,noauto	0
	0				

Save the file and exit.

### **Create the necessary directory's**

If you have a floppydrive:

```
1: mkdir /fdd
```

```
1: mkdir /mnt/fdd
```

```
1: mkdir /fixed/fdd
```

For the cdrom drive:

```
1: mkdir /cdrom
```

```
1: mkdir /mnt/cdrom
```

```
1: mkdir /fixed/cdrom
```

### **Copy fstab to encrypted part**

We also need to copy the fstab file from the unencrypted part to the encrypted part:

```
1: cp /mnt/etc/fstab /fixed/etc
```

### **Let FreeBSD ask for the passphrase at bootup**

```
1: echo geom_eli_load="\YES\" >> /mnt/boot/loader.conf
```

### **Copy Unencrypted boot to encrypted part**

Since we are going to use striping of FreeBSD we need some files that the install we did on the encrypted part does not have, we need to copy the unencrypted boot back to the encrypted boot directory.

```
1: cp -Rpnv /mnt/boot /fixed
```

*Wait for all the files to be copied*

## ***ALL STEPS DONE!!!!???***

**Are you sure that you have done all the above steps??**

```
1: shutdown -h NOW
```

45

### ***Disconnect the slave hdd***

Power down the Server, disconnect all the power to the machine and disconnect the Slave hdd from the IDE Cable.

### ***Test the FreeBSD encrypted version***

45

### ***See if you can login without a password***

If everything was going well you have to enter the passphrase that you have typed in the steps before. Then if you login with root, you will not be presented with a password. If this happens you know you are on the encrypted part of the hdd.

Since the installation is basic you need to configure everything by hand before it will work. Also a warning about a name server will pop by, that's because the network device is not configured yet.

46

### ***(optional) Connect your Harddisk drives to your Raid Controller***

```
1: shutdown -h NOW
```

When the machine's says the system is halted turn off your system and disconnect the powercable's.

When you are using a HPT374 like me, take note that Seagate ST3500630A (Barracuda) does not work together with Hitachi Deskstar IDE hdd's on the Rocketraid 454. I could not make a RAID 0/ JBOD or Mirror, so I have choosen todo this the software way.

Also sometimes a LED keeps on when the machine is started in FreeBSD and then ad6 hdd is not present in the /dev directory.

```
1: shutdown -r NOW
```

I have to restart it until all the Led's are off or a device is not detected. (Can anyone tell me what this is? It happens after I type in my Phassphrase)

It seems to me, when I wait too long with typing the correct passphrase that this happens??!!

47

### ***Enable networking***

Type:

```
1: Sysinstall
```

Select: Configure --> Networking --> Interfaces

In my case I select: {fxp1}

IPv6: {No}

DHCP {No}

I type in my credentials.

Bring the interface up now? {Yes}

{X} Exit

{X} Exit

{X} Exit Install

Reboot:

```
1: Shutdown -r NOW
```

48

### ***Enable SSH***

To work faster and from every pc I enable SSH login, type:

```
1: sysinstall
```

Choose: Configure --> Networking --> (Scroll down with arrow keys) sshd, choose {Ok}

{X} Exit --> {X} Exit Install

49

## **Check rc.conf for ssh**

Sysinstall is nice and easy, but you should know what it does. So we are going to check /etc/rc.conf for ssh

```
1: vi /etc/rc.conf
```

Check if you see the tag: sshd\_enable="YES", if so ssh will be enabled at next bootup.

```
1: shutdown -r NOW
```

50

## ***(optional) A little detour***

Yes! I did a little detour on this one, I tried to connect 4 IDE harddrives to one controller and 4 sata drives to a sata controller. Sad thing is, it was not stable, I have left this peace in this manual so you can learn from it. I bought 2 Sata drives of 2 TeraByte and now I'm using 4 x 2 TB Sata drives as one big volume.

### **Make a Stripe set (Raid0) with FreeBSD"]**

If you are using one controller card with 4 IDE drives and another controller card with 4 Sata drive's and you want one big volume you can use FreeBSD to stripe with the GEOM software.

### **Enable striping driver"]**

```
1: vi /boot/defaults/loader.conf
```

Search Geom\_Stripe and type "YES" instead of "NO"

**Save the file with ":w!"** (The **i** is to write a read-only file, only possible as root user)

### **Do the same for unencrypted part"]**

First mount the unencrypted part of the hdd

```
1: mount /dev/ad0s1a /mnt
```

```
1: vi //mnt/boot/defaults/loader.conf
```

Search Geom\_Stripe and type "YES" instead of "NO"

**Save the file with ":w!"**

### **Reboot to activate striping"]**

```
1: shutdown -r NOW
```

*And YES! it is native!*

### **Create first striping set"]**

I explained about the problem between the Seagate and the barracuda, so I solve this the software way. We are going to create a striping set from ad4,6,8 and 10. (The names may be different on your system, check the /dev directory).

```
1: gstripe label -v ide0 /dev/ad4 /dev/ad6 /dev/ad8 /dev/ad10
```

*This will create a striping set with the name ide0. I will give some errors that it will not use the intire hdd capacity, but that is common with RAID 0 sets, all volume's must be the exact same size.*

### **Check your striping set"]**

You can check your striping set by:

```
1: cd /dev/strip
```

```
1: ls
```

and search for ide0

### **Stripe over stripe"]**

Striping over striping, it's unbelievable that this is possible, in windows I would be afraid what will happen with the data, but on my FreeBSD box... I'am confident!

My other Rocket Raid (Model 1740) has created 3 striping sets so we have 4 times 2 TB of striping sets, create a second striping set:

```
1: gstripe label -v big0 /dev/strip/ide0 /dev/da0 /dev/da1 /dev/da2
```

*/dev/dax is the most common name for a hardware striping set, I have seen this in FreeBSD 8.x and in a VirtualMachine enviroment with iSCSI. Once I have seen arx in FreeBSD6.2*

### **Make the stripe (big0) encrypted"]**

And yes we want to encrypt this too! When I format the big0 volume it is somewhat slow, but the only thing this system has todo is duplicate data and share it when disaster strike's, so I don't care.

```
1: geli init -b -s 4096 -l 256 /dev/strip/big0
```

Type your secret passphrase twice.

### **Attach the big0 array**

```
1: geli attach /dev/strip/big0
```

Type type the passphrase and the usual confirmation will be shown.

### **Label the striped encrypted drive"]**

```
1: bsdlabel -w /dev/stripe/big0.eli
```

```
1: bsdlabel -e /dev/stripe/big0.eli
```

[x] to delete characters, Press {I} to edit/insert, Change unused behind a: to 4.2BSD, [ESC], :w, :q

### **Format the encrypted .elia drive"]**

```
1: newfs /dev/stripe/big0.elia
```

When I look at the drive I see the leds making disco, so I know all drive's are being used to stripe!

*When working with older stuff it helps to connect all the led's, for 20,- Euro's you got 10 of them with a wire and a little connector*

### **Mount it and check it"]**

```
1: mount /dev/stripe/big0.elia /encrypt_a
```

```
1: df -h
```

A list with the mounted drives will be presented, check if the size checks out.

### **Mount at startup"]**

```
1: vi /etc/fstab
```

Add the following line at the end of the file:

```
1: /dev/stripe/big0.elia /encrypt_a ufs rw 2 2
```

Save the file and exit vi.

### **Copy the new fstab to the unencrypted part"]**

```
1: mount /dev/ad0s1a /mnt
```

```
1: cp /etc/fstab /mnt/etc/fstab
```

## ***Install Rsync on your FreeBSD machine***

## **Encrypt the big volume**

My Big volume that are 4 S-Ata harddisks on one Sata Controller is called **/dev/da0**, the name for your array could be different, check the name first:

```
1: cd /dev
```

```
1: ls
```

*Search for da0 (or da1, da2, etc...) or ar0 (ar1, ar2, etc..) and use this device name to encrypt the big volume*

Encrypt the big volume with:

```
1: geli init -b -s 4096 -l 256 /dev/da0
```

Enter the passphrase twice.

52

## **Attach and format the big volume (da0)**

First we need to attach the encrypted device:

```
1: geli attach /dev/da0
```

Enter the passphrase you defined in the last step.

Make a label for the attached device:

```
1: bsdlabel -w /dev/da0.eli
```

```
1: bsdlabel -e /dev/da0.eli
```

[x] to delete characters, Press {I} to edit/insert, Change unused behind a: to 4.2BSD, [ESC], :w, :q

It needs to look like this:

```
1: # /dev/da0.eli
2: 8 partitions:
3: #      size      offset  fstype  [fsize  bsize  bps/cpg]
4:  a:  1953431549    2      4.2BSD  0       0       0
5:  c:  1953431549    0      unused  0       0               #"raw" part,
6: don't edit
```

*the offset will be different, this is the size of your volume.*

**Don't forget the save the file and then Exit**

Check if there are new devices in your /dev directory

```
1: cd /dev
```

```
1: ls
```

You should see devices like da0.elia or ar0.elia.

Format the new device:

```
1: newfs /dev/stripe/big0.elia
```

This may take a while, please wait!

53

## ***Mount the encrypted volume***

First make a directory where you can mount the volume:

```
1: mkdir /encrypt_a
```

Mount the big device:

```
1: mount /dev/stripe/da0.elia /encrypt_a
```

Check the size with command **df**:

```
1: df -h
```

Result:

1:	Filesystem	size	Used	Avail	Capacity	Mounted on
2:	/dev/ad0s2.elia	1.8G	315M	1.3G	19%	/
3:	devfs	1.0K	1.0K	0B	100%	/dev
4:	/dev/ad0s2.eli	961M	24K	884M	0%	/tmp
5:	/dev/ad0s2.elif	64G	1.7G	47G	3%	/usr
6:	/dev/ad0s2.elid	2.3G	834M	1.3G	38%	/var
7:	/dev/da0.elia	7.2T	2.6T	4.0T	40%	/encrypt_a
8:						

Your figures are probably different, but this is a good way to check if all the sizes are correct.

54

## ***Mount the new device at bootup***

Edit fstab

```
1: vi /etc/fstab
```

Add the following line at the bottom:

```
1: /dev/da0.elia /encrypt_a ufs rw 2 2
```

55

## ***Install rsync***

Now we have a device where we can but all the data from the primary server we need to sync it to the backup server.

You need to have rsync installed on the primary server, read here how you can do it!

To install Rsync we start sysinstall.

```
1: sysinstall
```

Select Configure --> Packages --> CD/DVD --> net --> rsync-3.x.x

Select {OK} --> {Install}

You will be shown what you have selected, select {OK}

Installation will commence.

{X} Exit

{X} Exit Install

Take out FreeBSD CD/DVD

56

## ***Setup rsyncd***

```
1: vi /usr/local/etc/rsyncd.conf
```

Remove # before "UID" & "GID" and change "nobody" to "rsync", the file will look like this:

```
1: # rsyncd.conf - Example file, see rsyncd.conf(5)
2: #
3:
4: # Set this if you want to stop rsync daemon with rc.d scripts
5: pid file = /var/run/rsyncd.pid
6:
7: # Edit this file before running rsync daemon!!
8:
9: uid = rsync
10: gid = rsync
11: #use chroot = no
12: #max connections = 4
13: #syslog facility = local5
14:
15: #[ftp]
16: #      path = /var/ftp/pub
```

```

17: #         comment = whole ftp area (approx 6.1 GB)
18:
19: #[smbaftp]
20: #         path = /var/ftp/pub/samba
21: #         comment = Samba ftp area (approx 300 MB)
22:
23: #[rsyncftp]
24: #         path = /var/ftp/pub/rsync
25: #         comment = rsync ftp area (approx 6 MB)
26:
27: #[sambawww]
28: #         path = /public_html/samba
29: #         comment = Samba WWW pages (approx 240 MB)
30:
31: #[cvs]
32: #         path = /data/cvs
33: #         comment = CVS repository (requires authentication)
34: #         auth users = tridge, susan
35: #         secrets file = /usr/local/etc/rsyncd.secrets

```

On the other side (the [primary server](#)) the file will look like this:

```

1: # rsyncd.conf - Example file, see rsyncd.conf(5)
2: #
3:
4: # Set this if you want to stop rsync daemon with rc.d scripts
5: pid file = /var/run/rsyncd.pid
6:
7: # Edit this file before running rsync daemon!!
8:
9: uid = rsync
10: gid = rsync
11: use chroot = no
12: max connections = 4
13: syslog facility = local5
14: pid file = /var/run/rsyncd.pid
15: auth users = roland, speciaal, copycop
16: secrets file = /usr/local/etc/rsyncd.secrets
17:
18: [test]
19:     path = /encrypt_a/tmp/
20:     comment = Test to sync the samba tmp directory
21:
22: [encrypt_a]
23:     path = /encrypt_a/
24:     comment = Shared Directory Tree
25:
26: #[ftp]
27: #         path = /var/ftp/pub
28: #         comment = whole ftp area (approx 6.1 GB)
29:
30: #[smbaftp]
31: #         path = /var/ftp/pub/samba
32: #         comment = Samba ftp area (approx 300 MB)
33:
34: #[rsyncftp]
35: #         path = /var/ftp/pub/rsync
36: #         comment = rsync ftp area (approx 6 MB)
37:
38: #[sambawww]

```

```
39: #      path = /public_html/samba
40: #      comment = Samba WWW pages (approx 240 MB)
41:
42: #[cvs]
43: #      path = /data/cvs
44: #      comment = CVS repository (requires authentication)
45: #      auth users = tridge, susan
46: #      secrets file = /usr/local/etc/rsyncd.secrets
```

57

### ***Create user rsync***

```
1: adduser
```

Fill in all the credentials.

58

### ***Create a batch file on the backup server***

```
1: vi /usr/local/bin/rsyncd.bat
```

It must contain:

```
#!/bin/sh
1: /usr/local/bin/rsync -avz --stats --delete
2: copycop@10.30.0.4::encrypt_a /encrypt_a --password-file
   /usr/local/etc/copycop.rsyncd
```

59

### ***Make the batch file startable***

```
1: chmod 0760 /usr/local/bin/rsyncd.bat
```

60

### ***Create the password file***

Now create the password file for copycop

```
1: vi /usr/local/etc/copycop.rsyncd
```

Type the password in the file and save it!

61

## ***Change the rights***

Change the rights of the file's so not everybody can read them.

```
1: chmod 0640 /usr/local/etc/copycop.rsyncd
```

```
1: chmod 0640 /usr/local/etc/rsyncd.conf
```

62

## ***Start Rsync to test syncing the data***

```
1: /usr/local/bin/rsyncd.bat
```

You should see:

```
1: recieving file list ...
```

and then popping a lot of file's on your screen! (And ofcourse the led's playing disco, as a matter a fact I feel Disco! Weeeh!!!)

If you get an error like: "rsync error error starting client-server protocol code 5", check the password you have used in the password file on the primary and backup server. Also check on the primary server if host allow contains the correct IP Adress.

63

## ***More pointers about errors with rsync***

If you get the following errors:

```
1: rsync error: some files/attrs were not transferred (see previous errors)
   (code23) at main.c(1508) [generator=3.0.7]
```

Scroll back and see on wich directory's you don't have access, change the rights of those directory's on the [primary server](#).

The error you should see an error in the copy files tree like this: send\_files failed to open "dir/ectory/" (in encrypt\_a): Permission denied (13)

64

## ***Setting up NTP for time sync***

Open rc.conf

```
1: vi /etc/rc.conf
```

Add the following to lines at the bottom:

```
1: ntp_date="10.30.0.4"
```

This is the IP address of your head server, so all systems in your network al synced together!

```
1: ntpdate_enable="YES"
```

## **Installation of Samba**

=====

What is the use of a backup server, when you cannot access it from windows, I will set it up samba so that you can only read from it!

65

### **Install Samba from the ports**

Insert your FreeBSD 8.1 install CD/DVD in your cd drive and type:

```
1: sysinstall
```

Wait for FreeBSD to complete the operation.

66

### **Edit smb.conf to configure samba**

```
1: vi /usr/local/etc/smb.conf
```

Here is an example of smb.conf file:

```
1:  # This is the main Samba configuration file. You should read the
2:  # smb.conf(5) manual page in order to understand the options listed
3:  # here. Samba has a huge number of configurable options (perhaps too
4:  # many!) most of which are not shown in this example
5:  #
6:  # For a step to step guide on installing, configuring and using samba,
7:  # read the Samba-HOWTO-Collection. This may be obtained from:
8:  # http://www.samba.org/samba/docs/Samba-HOWTO-Collection.pdf
9:  #
10: # Many working examples of smb.conf files can be found in the
11: # Samba-Guide which is generated daily and can be downloaded from:
12: # http://www.samba.org/samba/docs/Samba-Guide.pdf
13: #
14: # Any line which starts with a ; (semi-colon) or a # (hash)
15: # is a comment and is ignored. In this example we will use a #
16: # for commentry and a ; for parts of the config file that you
17: # may wish to enable
18: #
19: # NOTE: Whenever you modify this file you should run the command
20: "testparm"
21: # to check that you have not made any basic syntactic errors.
22: #
23: #===== Global Settings
24: =====
25: [global]
26:
27: # workgroup = NT-Domain-Name or Workgroup-Name, eg: MIDEARTH
28:     workgroup = Wayward
29:
30: # server string is the equivalent of the NT Description field
```

```
31:     server string = BSD02 Samba Server
32:
33: ## Samba Time Server?
34: #
35:     time server =yes
36:
37: ## getpeername failed. Error was socket is not connected, solution:
38: #
39: smb ports = 139
40:
41: # Security mode. Defines in which mode Samba will operate. Possible
42: # values are share, user, server, domain and ads. Most people will want
43: # user level security. See the Samba-HOWTO-Collection for details.
44:     security = user
45:
46: # This option is important for security. It allows you to restrict
47: # connections to machines which are on your local network. The
48: # following example restricts access to two C class networks and
49: # the "loopback" interface. For more examples of the syntax see
50: # the smb.conf man page
51:     hosts allow = 10.30.0. 127.
52:
53: # If you want to automatically load your printer list rather
54: # than setting them up individually then you'll need this
55:     load printers = no
56:
57: # you may wish to override the location of the printcap file
58:     printcap name = /dev/null
59:
60: # on SystemV system setting printcap name to lpstat should allow
61: # you to automatically obtain a printer list from the SystemV spool
62: # system
63: ;     printcap name = lpstat
64:
65: # It should not be necessary to specify the print system type unless
66: # it is non-standard. Currently supported print systems include:
67: # bsd, cups, sysv, plp, lprng, aix, hpux, qnx
68:     printing = bsd
69:
70: # Uncomment this if you want a guest account, you must add this to
71: /etc/passwd
72: # otherwise the user "nobody" is used
73: ;     guest account = pcguest
74:
75: # this tells Samba to use a separate log file for each machine
76: # that connects
77:     log file = /var/log/samba34/log.%m
78:
79: # Put a capping on the size of the log files (in Kb).
80:     max log size = 50
81:
82: # Use password server option only with security = server
83: # The argument list may include:
84: #     password server = My_PDC_Name [My_BDC_Name] [My_Next_BDC_Name]
85: # or to auto-locate the domain controller/s
86: #     password server = *
87: ;     password server = <NT-Server-Name>
88:
89: # Use the realm option only with security = ads
90: # Specifies the Active Directory realm the host is part of
91: ;     realm = MY_REALM
```

```
92:
93: # Backend to store user information in. New installations should
94: # use either tdbsam or ldapsam. smbpasswd is available for backwards
95: # compatibility. tdbsam requires no further configuration.
96: ;   passdb backend = tdbsam
97:
98: # Using the following line enables you to customise your configuration
99: # on a per machine basis. The %m gets replaced with the netbios name
100: # of the machine that is connecting.
101: # Note: Consider carefully the location in the configuration file of
102: #       this line. The included file is read at that point.
103: ;   include = /usr/local/etc/smb.conf.%m
104:
105: # Most people will find that this option gives better performance.
106: # See the chapter 'Samba performance issues' in the Samba HOWTO
107: Collection
108: # and the manual pages for details.
109: # You may want to add the following on a Linux system:
110: ;   socket options = SO_RCVBUF=8192 SO_SNDBUF=8192
111:
112: # Configure Samba to use multiple interfaces
113: # If you have multiple network interfaces then you must list them
114: # here. See the man page for details.
115: ;   interfaces = 192.168.12.2/24 192.168.13.2/24
116:
117: # Browser Control Options:
118: # set local master to no if you don't want Samba to become a master
119: # browser on your network. Otherwise the normal election rules apply
120: ;   local master = no
121:
122: # OS Level determines the precedence of this server in master browser
123: # elections. The default value should be reasonable
124: ;   os level = 33
125:
126: # Domain Master specifies Samba to be the Domain Master Browser. This
127: # allows Samba to collate browse lists between subnets. Don't use this
128: # if you already have a Windows NT domain controller doing this job
129: ;   domain master = yes
130:
131: # Preferred Master causes Samba to force a local browser election on
132: startup
133: # and gives it a slightly higher chance of winning the election
134: ;   preferred master = yes
135:
136: # Enable this if you want Samba to be a domain logon server for
137: # Windows95 workstations.
138: ;   domain logons = yes
139:
140: # if you enable domain logons then you may want a per-machine or
141: # per user logon script
142: # run a specific logon batch file per workstation (machine)
143: ;   logon script = %m.bat
144: # run a specific logon batch file per username
145: ;   logon script = %U.bat
146:
147: # Where to store roving profiles (only for Win95 and WinNT)
148: #       %L substitutes for this servers netbios name, %U is username
149: #       You must uncomment the [Profiles] share below
150: ;   logon path = \\%L\Profiles\%U
151:
152: # Windows Internet Name Serving Support Section:
```

```
153: # WINS Support - Tells the NMBD component of Samba to enable it's WINS
154: Server
155: ;   wins support = yes
156:
157: # WINS Server - Tells the NMBD components of Samba to be a WINS Client
158: #   Note: Samba can be either a WINS Server, or a WINS Client, but NOT
159: both
160: ;   wins server = w.x.y.z
161:
162: # WINS Proxy - Tells Samba to answer name resolution queries on
163: # behalf of a non WINS capable client, for this to work there must be
164: # at least one WINS Server on the network. The default is NO.
165: ;   wins proxy = yes
166:
167: # DNS Proxy - tells Samba whether or not to try to resolve NetBIOS
168: names
169: # via DNS nslookups. The default is NO.
170:   dns proxy = no
171:
172: # Charset settings
173: ;   display charset = koi8-r
174: ;   unix charset = koi8-r
175: ;   dos charset = cp866
176:
177: # Use extended attributes to store file modes
178: ;   store dos attributes = yes
179: ;   map hidden = no
180: ;   map system = no
181: ;   map archive = no
182:
183: # Use inherited ACLs for directories
184: ;   nt acl support = yes
185: ;   inherit acls = yes
186: ;   map acl inherit = yes
187:
188: # These scripts are used on a domain controller or stand-alone
189: # machine to add or delete corresponding unix accounts
190: ;   add user script = /usr/sbin/useradd %u
191: ;   add group script = /usr/sbin/groupadd %g
192: ;   add machine script = /usr/sbin/adduser -n -g machines -c Machine
193: -d /dev/null -s /bin/false %u
194: ;   delete user script = /usr/sbin/userdel %u
195: ;   delete user from group script = /usr/sbin/deluser %u %g
196: ;   delete group script = /usr/sbin/groupdel %g
197:
198:
199: #===== Share Definitions
200: =====
201: [homes]
202:   comment = Home Directories
203:   browseable = no
204:   writable = yes
205:
206: # Un-comment the following and create the netlogon directory for Domain
207: Logons
208: ; [netlogon]
209: ;   comment = Network Logon Service
210: ;   path = /usr/local/samba/lib/netlogon
211: ;   guest ok = yes
212: ;   writable = no
213: ;   share modes = no
```

```
214:
215:
216: # Un-comment the following to provide a specific roving profile share
217: # the default is to use the user's home directory
218: ;[Profiles]
219: ;   path = /usr/local/samba/profiles
220: ;   browseable = no
221: ;   guest ok = yes
222:
223:
224: # NOTE: If you have a BSD-style print system there is no need to
225: # specifically define each individual printer
226: [printers]
227:   comment = All Printers
228:   path = /var/spool/samba34
229:   browseable = no
230: # Set public = yes to allow user 'guest account' to print
231:   guest ok = no
232:   writable = no
233:   printable = yes
234:
235: # This one is useful for people to share files
236: ;[tmp]
237: ;   comment = Temporary file space
238: ;   path = /tmp
239: ;   read only = no
240: ;   public = yes
241:
242: # A publicly accessible directory, but read only, except for people in
243: # the "staff" group
244: ;[public]
245: ;   comment = Public Stuff
246: ;   path = /home/samba
247: ;   public = yes
248: ;   writable = yes
249: ;   printable = no
250: ;   write list = @staff
251:
252: # Other examples.
253: #
254: # A private printer, usable only by fred. Spool data will be placed in
255: # fred's
256: # home directory. Note that fred must have write access to the spool
257: # directory,
258: # wherever it is.
259: ;[fredsprn]
260: ;   comment = Fred's Printer
261: ;   valid users = fred
262: ;   path = /homes/fred
263: ;   printer = fred's_printer
264: ;   public = no
265: ;   writable = no
266: ;   printable = yes
267:
268: # A private directory, usable only by fred. Note that fred requires
269: # write
270: # access to the directory.
271: ;[fredsdir]
272: ;   comment = Fred's Service
273: ;   path = /usr/somewhere/private
274: ;   valid users = fred
```

```
275: ; public = no
276: ; writable = yes
277: ; printable = no
278:
279: # a service which has a different directory for each machine that
280: connects
281: # this allows you to tailor configurations to incoming machines. You
282: could
283: # also use the %U option to tailor it by user name.
284: # The %m gets replaced with the machine name that is connecting.
285: ;[pchome]
286: ; comment = PC Directories
287: ; path = /usr/pc/%m
288: ; public = no
289: ; writable = yes
290:
291: # A publicly accessible directory, read/write to all users. Note that
292: all files
293: # created in the directory by users will be owned by the default user,
294: so
295: # any user with access can delete any other user's files. Obviously
296: this
297: # directory must be writable by the default user. Another user could of
298: course
299: # be specified, in which case all files would be owned by that user
300: instead.
301: ;[public]
302: ; path = /usr/somewhere/else/public
303: ; public = yes
304: ; only guest = yes
305: ; writable = yes
306: ; printable = no
307:
308: # The following two entries demonstrate how to share a directory so
309: that two
310: # users can place files there that will be owned by the specific users.
311: In this
312: # setup, the directory should be writable by both users and should have
313: the
314: # sticky bit set on it to prevent abuse. Obviously this could be
315: extended to
316: # as many users as required.
317: ;[myshare]
318: ; comment = Mary's and Fred's stuff
319: ; path = /usr/somewhere/shared
320: ; valid users = mary fred
321: ; public = no
322: ; writable = yes
323: ; printable = no
324: ; create mask = 0765
325:
326: #----- My Shares -----
327: #####
328: # All drives on the backup server are read only
329: #
330:
331: # This one is useful for people to share files
332: [tmp]
333: comment = Temporary file space
334: path = /encrypt_a/tmp
335: writeable = no
```

```
336:    public = yes
337:
338: # Log share
339: #
340: [log]
341:    comment = Log files of BSD03
342:    path = /var/log
343:    public = yes
344:    writeable = no
345:    browseable = no
346:
347: # Private-drives
348: #
349: [private]
350:    comment = Eigen Prive directory op de Server
351:    path = /encrypt_a/Private/%U
352:    public = no
353:    writeable = no
354:    browseable = no
355:
356: # Appz Drive
357: #
358: [appz]
359:    comment = Programma's, Games en dergelijke.
360:    path = /encrypt_a/Appz
361:    public = no
362:    writeable = no
363:    browseable = no
364:    force create mode = 0775
365:    force directory mode = 0775
366:
367: # Special Drive
368: #
369: [special]
370:    comment = Special Drives for: Ftp, Images, Sound, Apache
371:    path = /encrypt_a/Special
372:    public = no
373:    writeable = no
374:    browseable = no
375:    force create mode = 0775
376:    force directory mode = 0775
377:
378: # Media
379: #
380: [media]
381:    comment = Media Audio, Video, Multimedia
382:    path = /encrypt_a/Media
383:    public = no
384:    writeable = no
385:    browseable = no
386:    force create mode = 0775
387:    force directory mode = 0775
388:
389: # Ons
390: #
391: [ons]
392:    comment = Gezamenlijke schijf
393:    path = /encrypt_a/Ons
394:    public = no
395:    writeable = no
396:    browseable = no
```

```

    force create mode = 0770
    force directory mode = 0770

# Startup With batch files for connecting to BSD03
#
[startup]
    comment = Batch files to connect to the BSD03 FreeBSD Server
    path = /encrypt_a/Startup
    public = yes
    writeable = no
    browseable = yes

397: # Share to dump all the Ghost images from dos
398: [image]
399:     comment = Drive to dump all the Ghost image's to
400:     path = /encrypt_a/Images
401:     public = no
402:     writeable = no
403:     browseable = yes
404:     force create mode = 0775
405:     force directory mode = 0775
406: # Shares for the Media Center
407: #
408: [video]
409:     comment = Video Files for the media center
410:     path = /encrypt_a/Media/movies
411:     writeable = no
412:     browseable = yes
413:     force create mode = 0775
414:     force directory mode = 0775
415:
416: [TV]
417:     comment = Alle the tv programs we like to keep
418:     path = /encrypt_a/Media/TV
419:     writeable = no
420:     browsable = yes
421:     force create mode = 0775
422:     force directory mode = 0775
423:
424: [pictures]
425:     comment = All our pictures
426:     path = /encrypt_a/Media/pictures
    writeable = no
    browseable = yes
    force create mode = 0775
    force directory mode = 0775

[audio]
    comment = All our avaible audio
    path = /encrypt_a/Media/audio
    writeable = no
    browseable = yes
    force create mode = 0775
    force directory mode = 0775

```

**On the backup server I have set writable = no on every share, this is a backup server not a working server!**

67

### ***Enable Samba at startup***

Edit the /etc/rc.conf file and enable samba:

```
1: vi /etc/rc.conf
```

Add the following lines at the bottom of rc.conf

```
1: nmbd_enable="YES"
2: smbd_enable="YES"
```

68

### ***Edit welcome message***

```
1: vi ./etc/motd
```

Put the following lines into the editor:

```
1: FreeBSD 8.1-RELEASE (BSD02) - 2010 /Node:3 (Original file:/etc/motd.bak)
2:
3: Running:
4: - Apache2, Php 5, Mysql 5
5: - Pure-FTPD, SSH
6: - Samba 3
7: - NFS
8:
9: IP: 10.30.0.4 / Gateway: 10.30.0.100
10:
11: - FreeBSD Handbook: http://www.FreeBSD.org
12: - Use sysinstall to install additional Packages
```

Save the message with :w and exit.

69

### ***Restart Machine and check Samba***

```
1: shutdown -r NOW
```

70

### ***Add the user you want to access samba***

Before the windows clients can access the samba shares you have to add them as a samba user:

```
1: smbpasswd -a username
```

Enter the password twice and do this for every user you want to be able to access samba.

71

### ***Try to connect with a windows client***

Start the machine, enter the Passphrases to mount the encrypted partitions and wait until nmbd & smbd is loaded, then start a windows client and enter the name of your server in windows explorer.

```
1: \\bsd02\
```

You should be able to access the shares, read it, but you cannot write it

72

### ***Create a cronjob for rsync***

If you are going to use rsync, do it on a regular base! We add a cronjob.

Login as root

Type:

```
1: crontab -e
```

Add the following line:

```
1: 00 3 * * * /usr/local/bin/rsyncd.bat
```

Save the file and exit, you should see the line:

```
1: crontab: installing new crontab
```

then you know it is about to run!

73

### ***Enter a password for the root account***

A very important step for security, enter a password for the root account, Type:

```
1: passwd root
```

Enter the password twice.

74

## **Mail server installeren (Dovecot)**

1. We gaan Dovecot gebruiken als onze IMAP mail server om al die mailboxen met informatie over de server uit te lezen. Ga naar de juiste directory:  
`cd /usr/ports/mail/dovecot`
2. Installeer het pakket:  
`make install clean`
3. Zodra het pakket niet gevonden kan worden, ga je naar de directory:  
`cd /usr/ports/distfiles`
4. **(Optioneel)** En download je het bestand als volgt:  
`fetch http://www.dovecot.org/releases/1.0/rc/dovecot-1.0.rc7.tar.gz`

Ga weer terug naar de dovecot directory en start de procedure opnieuw.

5. Ga naar de juiste directory:  
`cd /usr/local/etc`
6. Open het bestand:  
`vi /usr/local/etc/dovecot.conf`
7. Zorg dat de volgende regels op de volgende wijze in het configuratie bestand staan:  
protocols = imap pop3 imaps pop3s  
disable\_plaintext\_auth = no
8. Voeg het volgende toe aan het /etc/rc.conf bestand:  
`dovecot_enable="YES"`

Start dovecot door het volgende commando:

```
/usr/local/etc/rc.d/dovecot start
```

Als er: *Starting dovecot* staat is het gelukt.

## Forward root emails to account rootmail

Zorg ervoor dat het account rootmail bestaat en maak deze anders aan.

Wijzig het bestand: `/etc/aliases` en ga op zoek naar:

```
# Well-known aliases -- these should be filled in!
```

En voeg het volgende onder deze lijn toe:

```
root: rootmail  
manager: rootmail  
dumper: rootmail
```

Sla dit bestand op en start: `newaliases`

If you have an error like:

missing map file `/etc/mail/aliases.db`

The above will fix this

Install Cacti for statistics:

Install net-snmp

```
# cd /usr/ports/net-mgmt/net-snmp; make install clean
```

Add a line to /etc/rc.conf

```
snmpd_enable="YES"
```

Edit snmpd.conf

```
cd /usr/local/share/snmp
cp snmpd.conf.example snmpd.conf
vi snmpd.conf
```

And edit the things you want to change.

Cacti is a powerful network graphing utility that front ends Rrdtool. This process will compile Rrdtool and all of its dependencies for you.

```
# cd /usr/ports/net-mgmt/cacti; make install clean
```

**Enable MySQL in /etc/rc.conf**

```
1: mysql_enable="YES"
2: mysqllimits_enable="NO"
```

## ***Install Apache***

```
1: cd /usr/ports/www/apache22
2: make install clean
```

Choose the standard options

14

## ***Start apache at startup***

```
1: vi /etc/rc.conf
```

Add the following lines at the bottom:

```
1: # Apache 2
2: apache22_enable="YES"
```

### 9. Installeer php5 extensons

```
cd /usr/ports/lang/php5-extensions
make config
```

Selecteer: *MySQL, PCRE, Posix, Session*, typ dan:  
**make install clean**

Go to the public\_html directory: `/usr/local/www/apache22/data` and make a symbolic link to cacti:

```
Ln -s /usr/local/share/cacti/ cacti
```

At this to `/etc/contrab`:

```
* /5 * * * * cacti /usr/local/bin/php /usr/local/share/cacti/poller.php > /dev/null 2>&1
```

Setup The Database

```
#mysqladmin --user=root create cacti
```

Set the passwd for the cacti user.

```
#passwd cacti
```

Changing local password for cacti

New Password: [cactipasswd]

Retype New Password: [cactipasswd]

Edit the `/usr/local/share/cacti/include/config.php` file for the proper database permissions:

```
$database_type = "mysql";  
$database_default = "cacti";  
$database_hostname = "localhost";  
$database_username = "cacti";  
$database_password = "cactipasswd";  
$database_port = "3306";
```

Set the Cacti database's permissions.

```
#echo "GRANT ALL ON cacti.* TO cacti@localhost IDENTIFIED BY 'cactipasswd'; FLUSH PRIVILEGES;" | mysql
```

Import the default tables.

```
#mysql cacti < /usr/local/share/cacti/cacti.sql
```

Use the `php.ini` file.

```
Cd /usr/local/etc
```

```
cp php.ini-production php.ini
```

Edit the file and set the correct timezone :

```
Vi php.ini
```

Search for: /dat.timezone

And enter behind it: "Europe/Berlin"

Restart the machine

Go to the ip address of your machine and follow the instructions of cacti.

For the first time there will be asked for username and password, this will be:

Admin

admin

Then you will be forced to change the cacti password, I have used the same one as I entered with the installation routine.