

# OpenFest

Sofia, November 2011

Снишаване на риска от загуба на  
видеозаписване и данни чрез виртуализация  
на записващия сървър под FreeBSD

Златко Асенов

[zlatko.asenov@gmail.com](mailto:zlatko.asenov@gmail.com)

# Рискове при видеонаблюдението

- **Спиране на видеозаписването**
- **Загуба на данни – видеозаписи, конфигурации.**

# Обща коренова причина

- Загуба на електрическо захранване.
  - Решение – UPS.

# Коренови причини за отделните рискове

- Отпадане на видеокамера:
  - изгаряне на камера, спиране на подаване на видеосигнал, препречване на наблюдавания терен.
  - Риск: спиране на видеозаписването.
- Отпадане на мрежа – изгаряне на мрежови устройства, промяна на мрежова конфигурация.
  - Риск: спиране на видеозаписването.

# Коренови причини за отделните рискове

- Отпадане на видеозаписващ сървър
  - Рискове:
    - а) Спиране на видеозаписването
    - б) Загуба на данни – видеозаписи, конфигурации.

# Снишаване на риска от отпадане на видеозаписващ сървър

- Намиране на кореновите причини
- Постигане на целите:
  - свеждане на downtime на видеозаписването и загуба на данни до нула.
  - възможно най-бързо възстановяване на отпаднал сървър.
- Намиране решение.
- Изпълнение.
- Проверка и анализ.

# Коренови причини за отпадане на записващ сървър

- Изгаряне на хардуерен модул, водещо до downtime, както следва:
  - мрежова карта – до 2 часа.
  - друго (HDD, MB, etc.) – от 1 до 5 дни.
- Загуба на работоспособност на операционната система, водеща до downtime от 1 до 5 дни.
- Загуба на работоспособност на софтуера за видеозаписване, водеща до downtime от 1 до 5 дни.

# Намиране на решение за

- Дублиране на видеозаписването
  - Премахва риска от отпадане на видеозаписването.
- Виртуализация на видеозаписването
  - Премахва риска от отпадане на видеозаписването
- Взаимен backup на сървърите – всеки сървър съдържа настройките и копия на виртуалните машини на останалите.

# Описание на текущото състояние

- Видеозаписващи сървъри – 2 бр.
- ОС – Microsoft Windows
- Дублиране на видеозаписването – частично
- RAID – няма
- Копие на настройки - да

# Какво конкретно искаме?

- Зареждане на host OS от произволен твърд диск.
- Надежден софтуерен RAID.
- Възможност за прехвърляне на системата на друг хардуер без преинсталация.
- Бързо възстановяване при цялостно отпадане на хардуера.
- Минимално харчене на ресурси.

# Тествани системи

- Debian + VMWare – не работи поради разлика във версиите на компилаторите. Ядрото и инструментите се компилират с различни версии на gcc.
- Debian + VirtualBox – трудности при ъпдейт на ядрото. Debian може да се ъпдейтва автоматично. VirtualBox спира работа при разминаване на версията на ядро, сорс и версия на компилирания модул за работа с ядрото. Рискът е, че не знаем кога точно ще спре всичко едновременно.

# Тествани системи

- FreeBSD + QEMU – Изключително ниска производителност на виртуалната машина.
- FreeBSD + VirtualBox – Задоволителна производителност. Guest машината работи с 80% от производителността си в реална инсталация.

# Работещо решение:

## HOST

- FreeBSD 8.2 Stable
- gpt
- root on ZFS RAIDZ2
- zfsboot
- VirtualBox

## GUEST

- MS Windows XP + ACTI NVR
- MS Windows XP + GV 8.4 NVR

# HOST SETUP

- Сваляме snapshot на 8.2-STABLE

<ftp://ftp.de.freebsd.org/pub/FreeBSD/snapshots/201105/FreeBSD-8.2-STABLE-201105-amd64-memst>

- Ползваме ето този скрипт на **Philipp Wuensche**

<http://anonsvn.h3q.com/s/gpt-zfsroot.sh>

- Настройваме параметрите в скрипта, така че да отговарят на нашия хардуер.
- Рестартираме.

# HOST SETUP

- Ето как изглежда системата след инсталация:

```
%gpart show
```

```
=> 34 976770988 ad0 GPT (465G)
```

```
34 128 1 freebsd-boot (64k)
```

```
162 8388608 2 freebsd-swap (4.0G)
```

```
8388770 62914560 3 freebsd-zfs (30G)
```

```
=> 34 976773101 ad1 GPT (465G)
```

```
34 128 1 freebsd-boot (64k)
```

```
162 8388608 2 freebsd-swap (4.0G)
```

```
8388770 62914560 3 freebsd-zfs (30G)
```

# HOST SETUP

```
=> 34 976770988 ad2 GPT (465G)
    34 128 1 freebsd-boot (64k)
    162 8388608 2 freebsd-swap (4.0G)
    8388770 62914560 3 freebsd-zfs (30G)
```

```
=> 34 976773101 ad3 GPT (465G)
    34 128 1 freebsd-boot (64k)
    162 8388608 2 freebsd-swap (4.0G)
    8388770 62914560 3 freebsd-zfs (30G)
```

# HOST SETUP

%zpool status

pool: bigbro0

state: ONLINE

scan: scrub repaired 0 in 0h30m with 0 errors on Tue Oct 25 03:35:16 2011

config:

NAME	STATE	READ	WRITE	CKSUM
bigbro0	ONLINE	0	0	0
raidz2-0	ONLINE	0	0	0
gpt/system-disk0	ONLINE	0	0	0
gpt/system-disk1	ONLINE	0	0	0
gpt/system-disk2	ONLINE	0	0	0
gpt/system-disk3	ONLINE	0	0	0

errors: No known data errors

%

# HOST SETUP

- Добавяме останалите дискови дялове в /dev/gpt чрез

```
# gpart add -s 431G -t freebsd-ufs -l disk00 ad0
```

Резултат :

```
bigbro0# gpart show ad0
```

```
=>   34 976770988 ad0 GPT (465G)
      34    128   1 freebsd-boot (64k)
      162 8388608   2 freebsd-swap (4.0G)
      8388770 62914560  3 freebsd-zfs (30G)
      71303330 905467692  4 freebsd-ufs (431G)
```

```
bigbro0#
```

# HOST SETUP

- Base system upgrade

```
# mkdir /etc/cvsup
```

```
# cp /usr/share/examples/cvsup/stable-supfile /etc/cvsup/
```

Задаваме \*default host=cvsup.de.FreeBSD.org в /etc/cvsup/stable-supfile

```
# csup /etc/cvsup/stable-supfile
```

```
# cd /usr/src && make -j24 buildworld && make j24 buildkernel && make installkernel &&  
mergemaster -p && make installworld && mergemaster
```

```
# reboot
```

# HOST SETUP

- ZPOOL UPGRADE

```
# zpool upgrade -V 28 bigbro0
```

За всеки диск се ъпдейтва MBR:

```
# gpart bootcode -b /boot/pmbr -p /boot/gptzfsboot -i 1 ad0
```

- Инсталиране на портовете и VirtualBox

```
# portsnap fetch extract
```

```
# cd /usr/ports/emulators/virtualbox-ose && make config install clean
```

Премахнете опциите QT4, DEBUG, PULSEAUDIO, X11, CDE, NLS и изберете GUESTADDITIONS, DBUS, VNC, WEBSERVICE

Добавете в /etc/rc.conf

```
vboxnet_enable="YES"
```

```
vboxheadless_user="bigbro0"
```

и vboxdrv\_load="YES" в /boot/loader.conf

# HOST SETUP

- **Активиране на pf**

**В `/etc/rc.conf` добавяме**

- `pf_enable="YES"`
- `pf_rules="/etc/pf.conf"`
- `pflog_enable="YES"`
- `pflog_logfile="/var/log/pflog"`

**Примерна конфигурация на `/etc/pf.conf`**

- `block in all`
- `pass out all keep state`
- `tcp_ssh = "{ ssh }"`
- `trusted = "{192.168.1.0/24, 10.10.0.10}"`
- `pass in inet proto tcp from { $trusted } to port $tcp_ssh`
- `pass in inet proto icmp from { $trusted }`
- `pass in inet proto tcp from 127.0.0.1 to 127.0.0.1`

# HOST SETUP

# reboot

в: Защо reboot?

о: Защото всичко трябва да се тества :)

- Добавяне на полезни програми:

```
#pkg_add -r ifstat pftop nmap fping zfs-stats  
tmux
```

- Изтегляне и преправяне за наши нужди portupdater script, писан от DutchDaemon:

<http://forums.freebsd.org/showpost.php?p=39092&postcount=37>

•

# HOST SETUP

- Активиране на sshd в /etc/rc.conf  
ssh\_enable="YES"
- Задаване на IP, netmask, gateway, hostname  
ifconfig\_re0="inet 192.168.1.100 netmask 255.255.255.0"  
defaultrouter="192.168.1.1"  
hostname="bigbro0"
- Позволяване на директен достъп до определените дялове чрез добавяне в /etc/rc.conf на следния ред:  
chmod 660 /dev/gpt/disk0[0-3]

# HOST SETUP

- Автоматично синхронизиране на времето чрез `ntpdate_enable="YES"` в `/etc/rc.conf` и добавяне на:

```
# ntpdate  
0 * * * * root ntpdate 212.50.0.15
```

в `/etc/crontab`. Откомментируйте или добавете желан ntp сървър в `/etc/ntp.conf`

- Добавяме потребителя в съответните групи

```
# pw groupmod vboxusers -m bigbro0  
# pw groupmod operators -m bigbro0
```

# GUEST SETUP

- Инсталиране на виртуалната ОС на машина с работеща графична среда, за да се спести време. Добавяме SATA Controller.
- Настройваме мрежа и отдалечен достъп.
- Export на виртуалната ОС.
- Прехвърляне с sftp, filezilla, etc. изнесения файл bigbro0.ova
- Внасяне на bigbro0.ova

```
% cd
```

```
% VBoxManage import WindowsXp.ova
```

# GUEST SETUP

- Добавяне дисковите дялове за директен достъп

**ГРЕШНО:** VBoxManage internalcommands createrawvmdk -filename /usr/home/bigbro0/VirtualBox\ VMs/bb0/data\_bb0.vmdk -rawdisk /dev/ad3 -partitions 4

- **ВЯРНО:** VBoxManage internalcommands createrawvmdk -filename /usr/home/bigbro0/VirtualBox\ VMs/bb0/data\_bb0.vmdk **-rawdisk /dev/gpt/disk00**

\* в упътването пише да се прави по горния начин, но когато използваме gpt с disk label в /dev/gpt/disk00 например, трябва да посочим етикета вместо дяла. В противен случай MS Windows вижда диска, но не може да работи с него.

- Стартиране на VM и форматиране добавените дискове. Изключваме VM.

# Последни настройки

- Добавяне в `/etc/rc.local`
- `su -c - bigbro1 '/usr/home/bigbro1/bb0.sh'`
- Създаване на изпълняем скрипт `bb0.sh` със следния запис в него:

```
#!/bin/sh
```

```
VBoxHeadless --startvm bb0 &
```

- Променяне на група `bigbro0` към `operators` на следните директории и файлове:

Директория `.VirtualBox`

Директория `VirtualBox Vms`

Файл `bb0.sh`

# Резултат

- Елиминиран риск от загуба на видеозаписи при отпадане на записващ сървър.
- Снишен риск от голям downtime. Вече са нужни най-много 10 мин. за възстановяване от snapshot на работила VM.
- Междуконпютърна преносимост. Достатъчно е архитектурата да е amd64.
- По-пълно използване на капацитета на компютрите.

# Предимства и недостатъци на виртуализация на видеозаписване

- Недостатъци:
  - Понижена производителност на VM в сравнение с реална инсталация
  - Зависимост от операционната система на хоста и от виртуализиращия софтуер
- Предимства
  - По-голям брой записвани камери с безплатен лиценз
  - Всички други ползи на виртуализацията

# OpenFest

Sofia, November 2011

Питанки?

Златко Асенов

[zlatko.asenov@gmail.com](mailto:zlatko.asenov@gmail.com)