Build your own cloud

using ganeti (kvm, drbd for the win!)

Dobrica Pavlinušić
Luka Blašković

CUC2014
CARNetova korisnička konferencija

What can you do with ganeti?

- create instances, move them, convert them, live migrate, backup/restore
- remove nodes, add them, promote them to master candidates, change master node
- simulate disk failures, node failures, recover instances and nodes
- use advanced tools for balancing, instance placement, planning and automated recovery.
- but first let’s setup our test cluster...

Hardware setup

- this time we use 3 physical machines:
  - ganeticuc0.ffzg.hr
  - ganeticuc1.ffzg.hr
  - ganeticuc2.ffzg.hr
- all of them different (cpu count, ram)
- two nodes with 2x 2TB disks and one with 2x 1TB disks in jbod
- different nic count

Initial node config

- minimal Debian Wheezy setup
- booted form usb keychain into the compressed ram (http://goo.gl/luY3d6)
- ganeti 2.11 installed and tweaked with saltstack (http://goo.gl/RIwcmc)
- all nics in bonding mode with 2 bridged vlans
  - br80 (172.16.2.0/24) - main network
  - br1001 (172.16.3.0/24) - drbd network (mtu 9000)

Initial node config

- VG ***cucvg*** created with two PVs:

```
/dev/sda   cucvg lvm2 a--  1.82t   1.82t
/dev/sdb   cucvg lvm2 a--  1.82t   1.82t
```

- Custom 3.10 kernel based on well tested RHEL sources with latest drbd, zram and nic driver pathces (http://goo.gl/CcV9VO)

DNS setup

- first we must have working DNS, in production and here we use standard /etc/hosts files:

```
172.16.2.1      gateway.cuc
172.16.2.2      dhcp.cuc
172.16.2.10     master.cuc
172.16.2.53     node0.cuc node0
172.16.2.54     node1.cuc node1
172.16.2.55     node2.cuc node2
172.16.3.53     node0.drbd.cuc
172.16.3.54     node1.drbd.cuc
172.16.3.55     node2.drbd.cuc
```

Node hostname

- In details: http://docs.ganeti.org/ganeti/current/html/install.html#hostname-issues

```
# cat /etc/hostname
node2.cuc
# hostname $(cat /etc/hostname)
# hostname --fqdn # should not segfault :)
node2.cuc
```

*.debs

- already installed by saltstack http://goo.gl/UGCuIb
- please use wheezy-backports
- manual install would look like:

```
apt-get install -t wheezy-backports ganeti
```

Finally cluster initialize

- bootstrap the cluster on node0

```
gnt-cluster init \
--secondary-ip 172.16.3.53 \
--vg-name cucvg \
--no-etc-hosts \
--master-netdev br80 \
--enabled-hypervisors kvm \ --primary-ip-version
4 \
--enabled-disk-templates plain,drbd master.cuc
```

Let’s tune a little bit before fun starts

- ***gnt-cluster verify*** # is your friend
  ```
  ERROR: node node0.cuc: hypervisor
  kvm parameter verify failure (source
  cluster): Parameter 'kernel_path'
  fails validation: not found or not a
  file (current value: '/boot/vmlinuz-
  3-kvmU')
  ERROR: node node0.cuc: missing
  bridges: xen-br0
  ```
- Yes we have problems!

Let’s tune a little bit before fun starts

- `gnt-cluster info` # will show you cluster settings
- `/boot/vmlinuz-3-kvmU` is symlink to host kernel
- we use `/boot/vmlinuz-3.2-kvmU` (http://goo.gl/1e305E)
- `gnt-cluster modify -H kvm: kernel_path=/boot/vmlinuz-3.2-kvmU, kernel_args=ro, initrd_path=/boot/initrd.img-3.2-kvmU`

Let’s tune a little bit before fun starts

- `gnt-cluster modify -N link=br80 #` will fix second problem with missing default nic
- Because we use VG for other things:

```
gnt-cluster modify --reserved-lvs='cucvg/exports,cucvg/cache'
```

Add a node(s)

```
# gnt-node list # current state
Node      DTotal DFree MTotal MNode MFree Pinst Sinst
node0.cuc   3.6T  3.6T   7.8G  215M  5.6G     0     0
# gnt-node add \
   --secondary-ip=172.16.3.54 \
   node1.cuc
# gnt-node add \
   --secondary-ip=172.16.3.55 \
   node2.cuc
```

Add a node(s)

```
# gnt-node list
Node      DTotal DFree MTotal MNode MFree Pinst Sinst
node0.cuc   3.6T  3.6T   7.8G  221M  5.6G     0     0
node1.cuc   1.8T  1.8T  11.7G  147M  9.5G     0     0
node2.cuc   3.6T  3.6T  15.7G  147M 13.4G     0     0
```

More tuning

- Policy bounds for our test cluster:

```
# gnt-cluster modify --ipolicy-bounds-
specs min:disk-size=128,cpu-count=1,
disk-count=1,memory-size=128,
nic-count=1,spindle-use=1
/max:cpu-count=2,disk-count=2,
disk-size=3072,memory-size=512,
nic-count=2,spindle-use=2
```

Moar tuning

```
# gnt-cluster modify -D drbd:
metavg=cucvg,resync-rate=4194304

# gnt-cluster modify -H kvm:
root_path=/dev/vda
```

Next: Instance manipulation


CUC workshop
And now let’s do practical workshop!

Login

`ssh root@ganeticuc0.ffzg.hr` # if behind fw try port 443 or 80

Password:

****

This presentation:

http://bit.ly/cuc2014-ganeti

Instance creation

```
# gnt-instance add --iallocator=hail --disk-
template=plain --disk=0:size=1G -o
debootstrap+default instance0.cuc
```

/usr/share/ganeti/os/debootstrap/

/etc/ganeti/instance-debootstrap/variants.list

```
# cat /etc/ganeti/instance-
debootstrap/hooks/hostname
#!/bin/sh
echo ${instance%%.*} > $TMPDIR/etc/hostname
```

Jobs, List instances

```
root@node0:~# gnt-job list
 ID Status  Summary
952 running INSTANCE_CREATE(instance0.cuc)

root@node0:~# gnt-job watch 952
Output from job 952 follows
----


# gnt-instance list -o name,status,oper_vcpus,
oper_ram,disk_usage,pnode,snodes
Instance      Status  VCPUs Memory DiskUsage Primary_node Secondary_Nodes
instance0.cuc running     1   128M      1.0G node0.cuc
```

Serial console

```
root@node0:~# gnt-instance console instance0
```

press enter here

```
Debian GNU/Linux 7 instance0 ttyS0

instance0 login: root
Last login: Tue Nov 18 19:00:50 CET 2014 on ttyS0
Linux instance0 3.2.0-4-amd64 #1 SMP Debian 3.2.63-2+deb7u1 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@instance0:~#
```

ctrl+] to exit

```
root@instance0:~# Connection to node1.cuc closed.
root@node0:~#
```

Instance move

```
# gnt-instance move --node node2 instance0
Instance instance2 will be moved. This requires a shutdown of the
instance. Continue?
y/[n]/?: y
Tue Nov 18 19:24:41 2014  - INFO: Shutting down instance instance0.cuc on source
node node0.cuc
Tue Nov 18 19:24:46 2014 Exporting disk/0 from node0.cuc to node2.cuc
Tue Nov 18 19:24:50 2014 disk/0 is now listening, starting export
Tue Nov 18 19:24:52 2014 disk/0 is receiving data on node2.cuc
Tue Nov 18 19:24:52 2014 disk/0 is sending data on node0.cuc
Tue Nov 18 19:24:58 2014 disk/0 sent 173M, 25.0 MiB/s, 16%, ETA 34s
Tue Nov 18 19:25:55 2014 disk/0 finished sending data
Tue Nov 18 19:26:00 2014 disk/0 finished receiving data
Tue Nov 18 19:26:00 2014  - INFO: Removing the disks on the original node
Tue Nov 18 19:26:00 2014  - INFO: Starting instance instance0.cuc on node node2.
cuc
```

Convert instance to DRBD

```
# gnt-instance shutdown instance0
```

```
# gnt-instance modify --disk-template=drbd --no-wait-for-sync --node node0 instance0
```
iallocator instead of --node in >=2.12

```
# gnt-instance start instance0
```

```
# gnt-instance list -o name,status,oper_vcpus,oper_ram,disk_usage,pnode,snodes instance2
Instance      Status  VCPUs Memory DiskUsage Primary_node Secondary_Nodes
instance2.cuc running     1   128M      2.1G node2.cuc    node0.cuc
```

Live migrate

```
# gnt-instance list -o name,pnode,snodes
instance0
Instance      Primary_node Secondary_Nodes
instance0.cuc node0.cuc    node2.cuc

# gnt-instance migrate instance0

# gnt-instance list -o name,pnode,snodes
instance0
Instance      Primary_node Secondary_Nodes
instance0.cuc node2.cuc    node0.cuc
```

Increase instance disk size with reboot

```
root@node0:~# gnt-instance grow-disk instance0 0 1G
Tue Nov 18 19:55:38 2014 Growing disk 0 of instance 'instance0.cuc' by 1.0G to 2.0
G
Tue Nov 18 19:55:39 2014  - INFO: Waiting for instance instance0.cuc to sync disks
Tue Nov 18 19:55:40 2014  - INFO: - device disk/0:  0.80% done, 3m 57s remaining
(estimated)
Tue Nov 18 19:56:40 2014  - INFO: Instance instance0.cuc's disks are in sync
```

```
root@node0:~# gnt-instance reboot instance0
gnt-instance reboot instance0
root@node0:~# gnt-instance console instance0
```

```
root@instance0:~# resize2fs /dev/vda
root@instance0:~# df -h /
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda        2.0G  447M  1.5G  24% /
```

Increase instance disk size without reboot

```
root@node0:~# gnt-instance grow-disk instance0 0 1G
root@node2:~# ps aux | grep name\ instance0
..qemu-system-x86_64 .. -device virtio-blk-pci,drive=hotdisk-143b8467-pci-4,id=hotdisk-143b8467-pci-4,bus=pci.0,addr=0x4 -drive
root@node2:~# echo block_resize hotdisk-143b8467-pci-4 2G | socat - unix:/var/run/ganeti/kvm-hypervisor/ctrl/instance0.cuc.monitor

root@node0:~# gnt-instance console instance0

root@instance0:~# resize2fs /dev/vda
root@instance0:~# df -h /
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda        2.0G  447M  1.5G  24% /
```

Backup instances

```
# gnt-backup export --node node0 --noshutdown instance0
Tue Nov 18 19:50:37 2014 Creating a snapshot of disk/0 on node node2.cuc
Tue Nov 18 19:50:37 2014 Exporting snapshot/0 from node2.cuc to node0.cuc
Tue Nov 18 19:50:41 2014 snapshot/0 is now listening, starting export
Tue Nov 18 19:50:43 2014 snapshot/0 is receiving data on node0.cuc
Tue Nov 18 19:50:43 2014 snapshot/0 is sending data on node2.cuc
Tue Nov 18 19:50:48 2014 snapshot/0 sent 73M, 12.0 MiB/s
Tue Nov 18 19:51:30 2014 snapshot/0 finished receiving data
Tue Nov 18 19:51:30 2014 snapshot/0 finished sending data
Tue Nov 18 19:51:30 2014 Removing snapshot of disk/0 on node node2.cuc
Tue Nov 18 19:51:30 2014 Finalizing export on node0.cuc
Tue Nov 18 19:51:31 2014 Removing old exports for instance instance0.cuc
```

```
root@node0:~# ls -al /var/lib/ganeti/export/instance0.cuc/
total 438024
drwx------ 2 root root      4096 Nov 18 19:53 .
drwxr-xr-x 3 root root      4096 Nov 18 19:53 ..
-rw------- 1 root root 448522240 Nov 18 19:53 2c7eadc2-2bff-4e6b-8949-
b4012b368835.disk0_data.snap
-rw------- 1 root root      1809 Nov 18 19:53 config.ini
```

Balance your cluster

- dry-run, see what will be changed

```
# hbal --luxi --mond=yes
```

- execute changes

```
# hbal --luxi --mond=yes --exec
```

this will take some time!

Space planning on my cluster

```
root@node0:~# hspace --luxi
The cluster has 3 nodes and the following resources:
  MEM 36036, DSK 9535464, CPU 20, VCPU 80.
There are 1 initial instances on the cluster.
Tiered (initial size) instance spec is:
  MEM 512, DSK 3072, CPU 2, using disk template 'drbd'.
Tiered allocation results:
  -  29 instances of spec MEM 512, DSK 3072, CPU 2
  -   1 instances of spec MEM 512, DSK 3072, CPU 1
  - most likely failure reason: FailMem
  - initial cluster score: 2.02904035
  -   final cluster score: 12.11524279
  - memory usage efficiency: 42.98%
  -   disk usage efficiency:  3.89%
  -   vcpu usage efficiency: 100.00%
Standard (fixed-size) instance spec is:
  MEM 128, DSK 1024, CPU 1, using disk template 'drbd'.
Normal (fixed-size) allocation results:
  -  47 instances allocated
  - most likely failure reason: FailDisk
  - initial cluster score: 2.02904035
  -   final cluster score: 8.98818671
  - memory usage efficiency: 17.05%
  -   disk usage efficiency:  2.97%
  -   vcpu usage efficiency: 85.00%
```

Planned maintenance (upgrade!)

- Evacuating instances

```
root@node0:~# gnt-node modify --drained=yes node1
root@node0:~# hbal -L -X
root@node0:~# gnt-node modify --offline=yes node1


root@node1:~# ipmitool lan print 1
dpavlin@black:~$ ipmitool -H 10.80.2.154 -U root -P calvin power
off
```

- Using node again

```
dpavlin@black:~$ ipmitool -H 10.80.2.154 -U root -P calvin power
on
root@node0:~# gnt-node modify --offline=no node1
root@node0:~# hbal -L -X
```

Failover master

- kill master node

```
root@node0:~# gnt-cluster getmaster
node0.cuc
```

```
root@black:~# ipmitool -H 10.80.2.153 -U root -P calvin power
off
```

- failover master to new node

```
root@node2:~# gnt-cluster getmaster
node0.cuc
root@node2:~# gnt-cluster master-ping || gnt-cluster master-
failover
```

```
root@node2:~# gnt-cluster getmaster
node2.cuc
```

- power on old node and re-add it to cluster

```
root@black:~# ipmitool -H 10.80.2.153 -U root -P calvin power on
```

```
root@node2:~# gnt-node add --readd --secondary-ip=172.16.3.53
node0.cuc
```

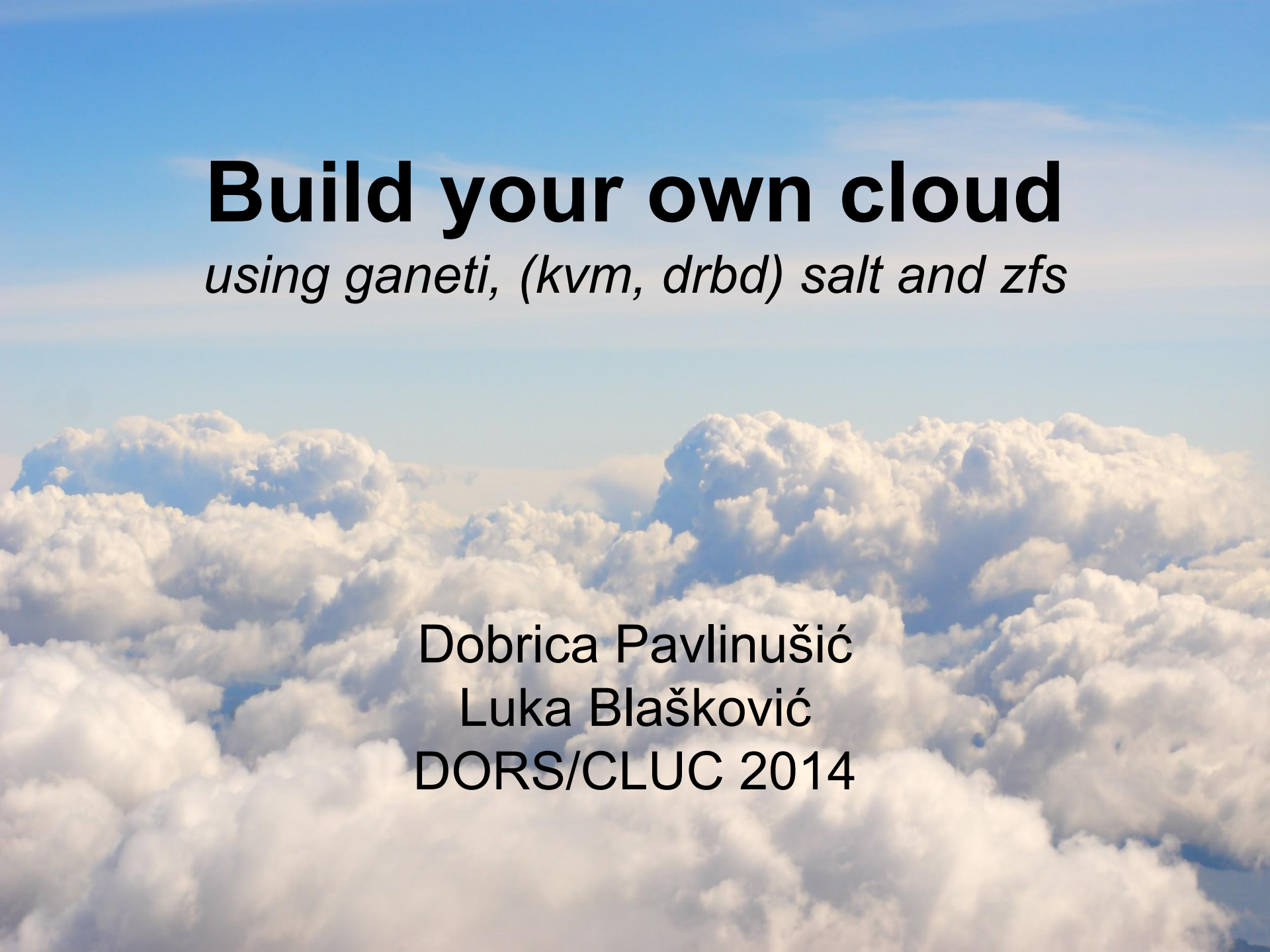
Repair or re-create instances

```
root@node0:~# gnt-cluster add-tags
ganeti:watcher:autorepair:fix-storage

man harep

root@node0:~# harep --luxi
```

Test cluster will stay online to the end of the conference! Do ganeti not botnets ! :)

The End?

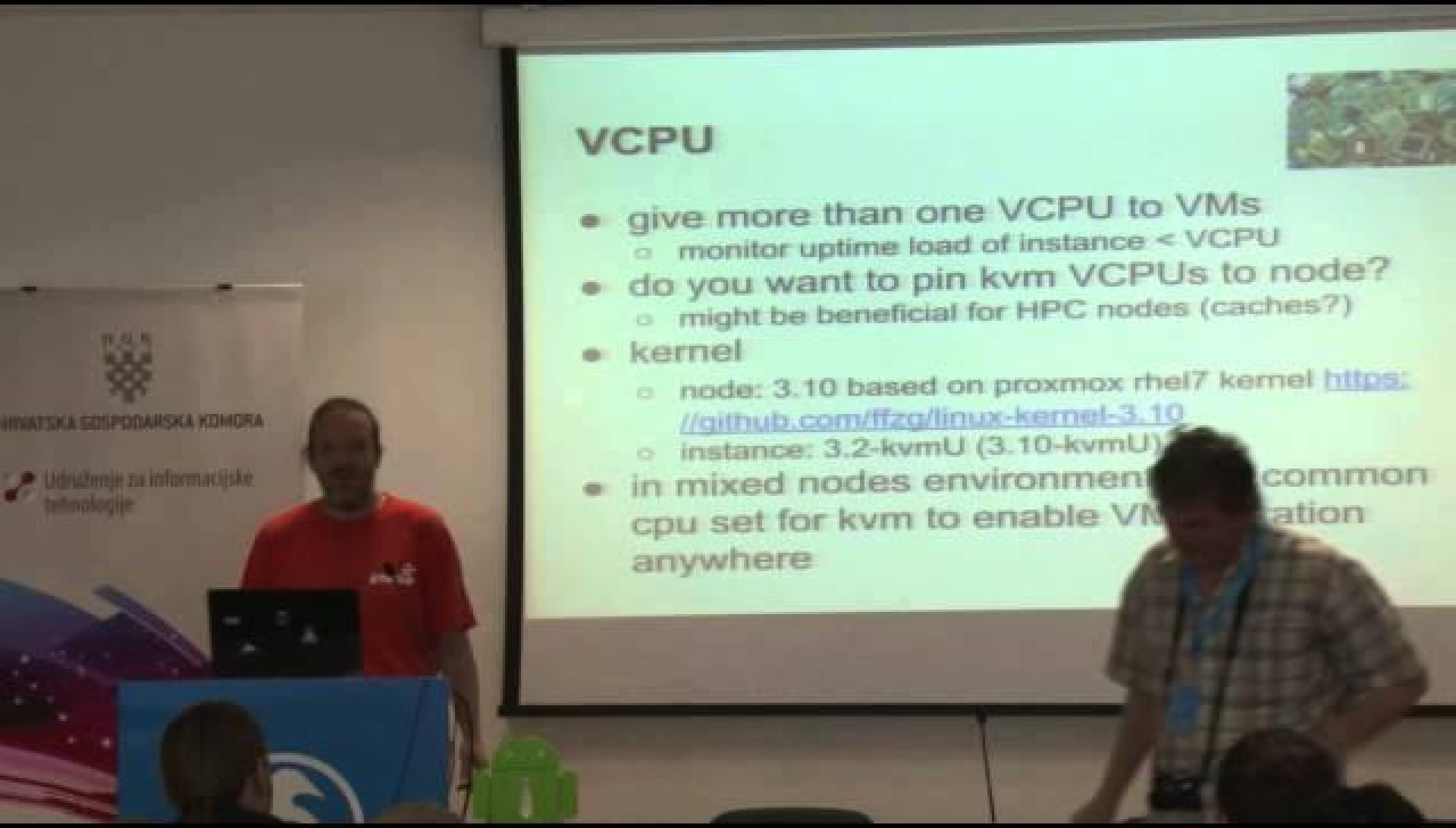
No, just beginning…
Examine following slides and videos for (much) more information….



Build your own cloud

using ganeti, (kvm, drbd) salt and zfs

Dobrica Pavlinušić
Luka Blašković
DORS/CLUC 2014


VCPU
give more than one VCPU to VMs
monitor uptime load of instance < VCPU
do you want to pin kvm VCPUs to node?
might be beneficial for HPC nodes (caches?)
kernel
node: 3.10 based on proxmox rhel7 kernel https://github.com/ffzg/linux-kernel-3.10
instance: 3.2-kvmU (3.10-kvmU)
in mixed nodes environment
common
cpu set for kvm to enable VM
ation
anywhere
HRVATSKA GOSPODARSKA KOMORA
Udruženje za informacijske tehnologije

What are we going to talk about?

- Which cloud IaaS or PaaS
- FFZG legacy infrastructure overview
- Ganeti - Open Source cloud solution
- SaltStack - deploy ganeti nodes
- ZFS - storage server (extstorage, nfs)
- our migration to cloud
- http://youtu.be/hiuHAPeRYsw

Cloud: is it IaaS or PaaS ?

Infrastructure as a service

reliable, persistent VMs
legacy consolidation

VMWare
Amazon EC2 (persistent?)
oVirt (libvirt)
Ganeti
OpenStack

Platform as a service

deploy applications using custom config

heroku
Google App Engine
Azure
Docker (kubernetes, DEIS)

Motivation for building a cloud

- 10+ aging Debian GNU/Linux machines installed in last 15 years on three locations
- upgraded memory
- upgraded disks (SAS and SATA)
- better resource usage
- **high availability**
  - resilient to failure of machines
  - maintenance during working hours
- VMs are not cattle, they are pets
- Every VM configured like real snowflake

SaltStack

- http://www.saltstack.com/
- automation for installation of ganeti nodes
- ZeroMQ and declarative rules
- deployment of new node under an hour
  https://github.com/ffzg/ganeti-salt

Ganeti integrates known tools

- kvm (or xen) virtualization
- drbd (w/ LVM) for disk replication (no SAN!)
- kvm+drbd = HA with live migration

Terminology:

- node - physical hardware
- instance - virtual machine
- cluster - combination of above components

gnt-* command-line interface for sysadmins

Ganeti hints

What you wanted to know about cloud but were too afraid to ask it….

ganeti nodes and instances

```
root@vmh02:~# gnt-node list
Node                  DTotal   DFree MTotal MNode MFree Pinst Sinst
arh-hw.gnt.ffzg.hr         ?       ?   7.8G  173M  1.3G     0     0
blade05.gnt.ffzg.hr 123.7G    1.4G   7.8G  5.0G  2.5G     8     2
box02.gnt.ffzg.hr     1.3T 1005.6G  15.7G 10.0G  6.7G    14     0
lib10.gnt.ffzg.hr     3.6T    2.5T  19.6G 12.1G 10.6G     8     7
lib15.gnt.ffzg.hr   543.7G  279.5G  15.7G  8.4G 10.6G     3     2
lib20.gnt.ffzg.hr   822.6G  516.4G  15.7G 10.7G  4.2G     3     3
vmh01.gnt.ffzg.hr   917.0G  583.3G  11.7G  7.6G  4.6G     8     8
vmh02.gnt.ffzg.hr   917.0G  569.7G  15.7G 10.0G  6.5G     8     7
vmh03.gnt.ffzg.hr   917.0G  592.9G  15.7G  8.9G  9.5G     8     7
vmh11.gnt.ffzg.hr   264.9G   38.6G   7.8G  5.2G  1.7G     8     7
vmh12.gnt.ffzg.hr   917.0G  566.6G  15.7G  9.7G  7.7G     5    10
```

```
root@vmh02:~# gnt-instance list --no-headers -o status,hv/kernel_path | sort |
uniq -c
      2 ADMIN_down
      4 ADMIN_down /boot/vmlinuz-3.2-kvmU
     34 running
     33 running    /boot/vmlinuz-3.2-kvmU
```

```
root@vmh02:~# kvm -version
QEMU emulator version 1.7.0 (Debian 1.7.0+dfsg-2~bpo70+2), Copyright (c) 2003-2008
```

disks

- two LVs as disks for instance (root, swap)
- boot via grub or from host kernel
- liberal use of nfs (from zfs pool) to provide shares to VMs (backups, archives…)
- gnt-instance modify -t drbd
- gnt-backup assumes 1 partition per disk
  - create LV snapshot (without shutdown)
  - transfer dump of file system to some node
  - remove snapshot
- plan to modify into incremental backup
  - lv snapshot => rsync => zfs snap

PERC SAS/SATA controllers

PERC 4 - bios JBOD mode (SCSI vs RAID)

PERC 5 - no JBOD mode

PERC 6 - LSI IT firmware for JBOD mode (newer IR have JBOD)

SMBus issue on Intel Chipsets with tape fix

http://www.overclock.net/t/359025/perc-5-i-raid-card-tips-and-benchmarks

VCPU

- give more than one VCPU to VMs
  - monitor uptime load of instance < VCPU
- do you want to pin kvm VCPUs to node?
  - might be beneficial for HPC nodes (caches?)
- kernel
  - node: 3.10 based on proxmox rhel7 kernel https://github.com/ffzg/linux-kernel-3.10
  - instance: 3.2-kvmU (3.10-kvmU)
- in mixed nodes environment, use common cpu set for kvm to enable VM migration anywhere

reboot

- **It will happen, sooner than you think**
- **don’t run manually started services!**
- acpi-support-base for clean shutdown
- gnt-instance reboot [instance]
  - power-cycle as opposed to reboot within instance (ganeti >=2.11 kvmd)
  - required to reload kvm config, hwclock, etc

network

- bonded 1G bridges per vlan
- jumbo frames for drbd traffic (9k mtu)
- disable host nic hardware offloads
- don’t let bridge traffic pass through fw chains
- pay with sysctl setting, switch congestion control algorithm
- Use our virtio-mq patch (ganeti >=2.12, linux kernel >=3.8)

tap challenges

```
qemu-system-x86_64: -netdev type=tap,id=hotnic-a74f9700-pci-6,fd=8,vhost=on: Device 'tap' could not be initialized
```

```
gnt-instance modify -H vhost_net=false pxelator
```

- mysterious unreported bug when vhost_net=True (network offloading from qemu to separate kernel thread)
- we will fix this, don’t worry :)

groups

- limit instance drbd replication and migration
  - e.g. same top-of-rack switch

```
root@vmh02:~# gnt-instance list --no-headers -o
status,pnode.group,snodes.group | sort | uniq -c
      6 ADMIN_down test
      6 running    default
     48 running    default default
      5 running    lib     lib
      8 running    test
```

console

- serial console
  - console=ttyS0
  - gnt-instance console [instance]
- VNC for graphic console
  - vnc on local address
  - NoVNC web console
  - https://code.grnet.gr/projects/ganetimgr/
  - https://code.osuosl.org/projects/ganeti-webmgr/

NoVNC web console

https://code.grnet.gr/projects/ganetimgr/

time

- ntp and/or ntpdate inside vms harmful
- ntp should be on node
- make sure that UTC=yes is same on vm/host

htools

A collection of tools to provide auxiliary functionality to Ganeti.

- hail: gnt-instance add -I hail instance # Where to put an instance ?
- hbal: hbal -G default -L --mond=yes # cluster balancing
- hspace: hspace -L # How many more instances can I add to my cluster ?
- harep: harep -L # repair/recreate instances

Exclusion tags

instance tags ensure that same service won't end up on same physical node

```
# man hbal
# gnt-cluster add-tags htools:iextags:service
# gnt-instance add-tags kibana service:
elasticsearch
```

Reason trails

- Jobs can be expanded to other jobs
- Keep track why job was run, inherited to all child jobs
- ganeti 2.13+ has rate-limit buckets

```
gnt-group evacuate --reason “rate-limit:5:
maintenance 123” groupA
```

Planned maintenance

- Evacuating instances

```
gnt-node modify --drained=yes node.example.com
hbal -L -X
gnt-node modify --offline=yes node.example.com
```

- Using node again

```
gnt-node modify --offline=no node.example.com
hbal -L -X
```

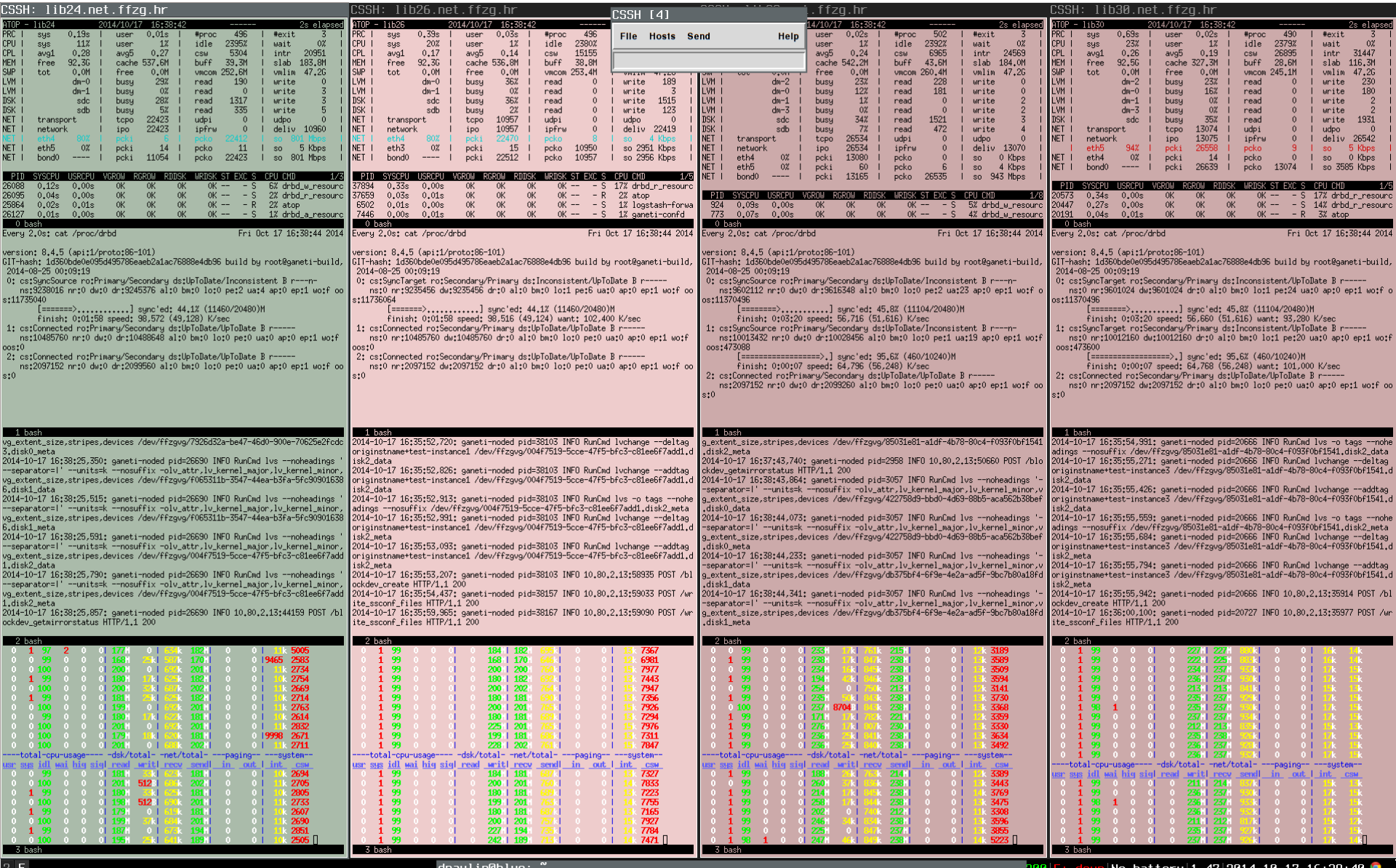
burn-in

So, will my cloud work in production?

Will I hit some arbitrary limit or bug in stack?

```
/usr/lib/ganeti/tools/burnin --disk-size=10G,1G,
5G -p --no-ip-check --no-name-check -n lib24,
lib26,lib28,lib30 --early-release --maxmem-
size=4G --minmem-size=1G --vcpu-count=4 --disk-
growth=11G,2G,6G -o debootstrap+default test-
instance{1..100}
```

burn-in - 4 nodes, 100 instances



stuck jobs

Stuck jobs are (sadly) ganeti's reality, version 2.12 + create jobs as separate processes so it's easy to shot them in the head with *kill -9.* With lower versions you can do something like this:

stop ganeti on master node

gnt-job list # find job id

ID=55321

mv /var/lib/ganeti/queue/job-${ID} /var/lib/ganeti/queue/archive/${ID:0:2}/

Migration of LXC into Ganeti VMs

Your (LXC) snowflakes can melt in process

- create LV for root fs
- rsync files over (defragment, ext4 upgrade)
- VMs disk size = used + 10%
- use host 3.2 kernel to run machines
- install modules and acpi support
- modify disk configuration to drbd

http://sysadmin-cookbook.rot13.org/#ganeti_migrate_lxc

Our experience

- We are not creating similar instances
- Performance impact compared to LXC
- Memory usage of VM hit-or-miss game
- Memory upgrade during working hours (evacuate, power off, upgrade, hbal)
- Firmware upgrades become reality
- First time to backup some machines (!)
- Works for us™ (anecdotally, more stable than commercial Xen or VMWare)
- tune your cluster and it will work well for you

https://code.google.com/p/ganeti/wiki/PerformanceTuning

Ganeti is good cloud core

- ganetimgr - KISS web interface https://code.grnet.gr/projects/ganetimgr/
- Synnefo - AWS like compute, network, storage https://www.synnefo.org/
  - OpenStack API (not code!)
  - Archipelago - distributed storage management
    - Ceph - distributed disk store


Questions?
And now let’s do practical workshop!

Technologies

- Linux and standard utils (iproute2, bridge-utils, ssh)
- socat
- KVM/Xen/LXC
- DRBD, LVM, SAN, Ceph, Gluster (=>2.11)
- Python (plus a few modules)
- Haskell

Ganeti on ganeti

- 6 virtual nodes
- nested virtualization not working (no KVM)
- separate volume group
- so plan is to setup XEN-PVM (paravirtualized), sorry no KVM this time :(

Bootstrap virtual “nodes”

```
gnt-instance add -t plain \
-n node{0..5}
-B maxmem=3.7G,minmem=1G,vcpus=4 \
-o debootstrap+salted \
--disk 0:size=20g,vg=dorsvg \
--disk 1:size=2g,vg=dorsvg \
--disk 2:size=300g,vg=dorsvg \
--net 0:mode=bridged,link=br1001 \
--net 1:mode=bridged,link=br0080 \
--no-name-check --no-ip-check \
dors-ganeti{0..5}.dhcp.ffzg.hr # metavg= for drbd
```

debootstrap+salted

- debootstrap default variant with saltstack bootstrap script:

https://raw.githubusercontent.com/lblasc/dorscluc2014-ganeti/master/salted_variant.sh

Initial salting

- nodes (minions) are automagically connected to master (know as “h”)

```
lblask@h:~$ sudo salt-key -L
Accepted Keys:
Unaccepted Keys:
dors-ganeti01.dhcp.ffzg.hr
dors-ganeti02.dhcp.ffzg.hr
dors-ganeti03.dhcp.ffzg.hr
dors-ganeti12.dhcp.ffzg.hr
dors-ganeti20.dhcp.ffzg.hr
dors-ganeti21.dhcp.ffzg.hr
```

Initial salting

lblask@h:~$ sudo salt-key -A

The following keys are going to be accepted:
Unaccepted Keys:
dors-ganeti01.dhcp.ffzg.hr
dors-ganeti02.dhcp.ffzg.hr
dors-ganeti03.dhcp.ffzg.hr
dors-ganeti12.dhcp.ffzg.hr
dors-ganeti20.dhcp.ffzg.hr
dors-ganeti21.dhcp.ffzg.hr

Proceed? [n/Y] y

Initial salting

- used states: https://github.com/lblasc/dorscluc2014-ganeti
- check boring stuff (apt_sources, dhcp hostname, locales, timezone, ssh)
- install xen kernel and tools
- leave hard work to workshoppers

Initial salting

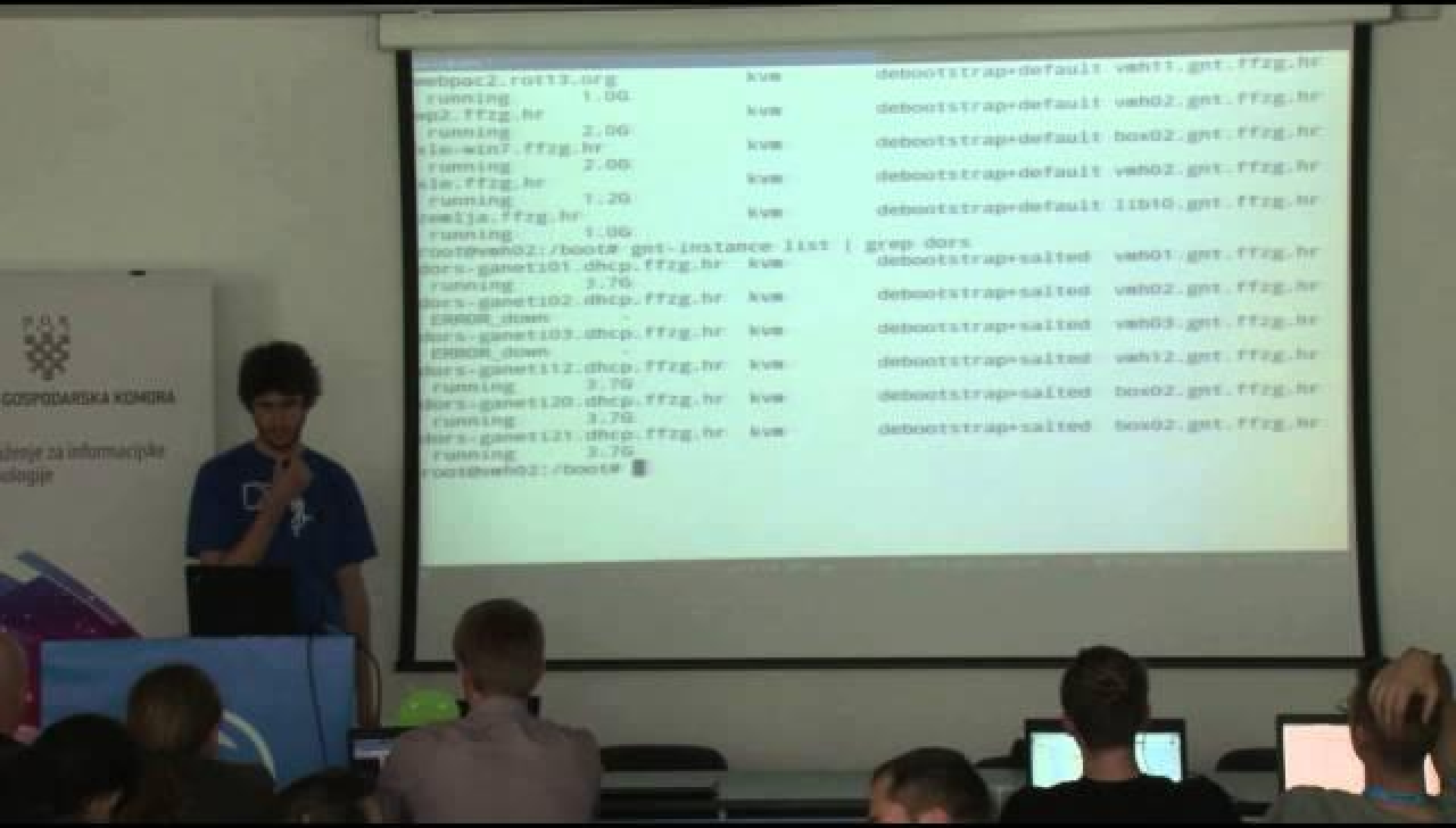
- modify instances to boot from own kernel

```
for x in \
$(gnt-instance list|grep dors|awk '{print $1}'| xargs); \
do gnt-instance modify --submit \
-H initrd_path=,kernel_path=,disk_type=scsi,
nic_type=e1000 $x \
; done
```

Initial salting

- reboot instances

```
for x in \
$(gnt-instance list|grep dors|awk '{print $1}'| xargs); \
do gnt-instance reboot --submit $x \
; done
```


webpac2.rot13.org kvm debootstrap+default vmh11.gnt.ffzg.hr
running 1.0G
wp2.ffzg.hr kvm debootstrap+default vmh02.gnt.ffzg.hr
running 2.0G
ala-win7.ffzg.hr kvm debootstrap+default box02.gnt.ffzg.hr
running 2.0G
ala.ffzg.hr kvm debootstrap+default vmh02.gnt.ffzg.hr
running 1.2G
nawija.ffzg.hr kvm debootstrap+default lib10.gnt.ffzg.hr
running 1.0G
root@vmh02:/boot# gnt-instance list | grep dors
dors-ganeti01.dhcp.ffzg.hr kvm debootstrap+salted vmh01.gnt.ffzg.hr
running 3.7G
dors-ganeti02.dhcp.ffzg.hr kvm debootstrap+salted vmh02.gnt.ffzg.hr
ERROR_down -
dors-ganeti03.dhcp.ffzg.hr kvm debootstrap+salted vmh03.gnt.ffzg.hr
ERROR_down -
dors-ganeti12.dhcp.ffzg.hr kvm debootstrap+salted vmh12.gnt.ffzg.hr
running 3.7G
dors-ganeti20.dhcp.ffzg.hr kvm debootstrap+salted box02.gnt.ffzg.hr
running 3.7G
dors-ganeti21.dhcp.ffzg.hr kvm debootstrap+salted box02.gnt.ffzg.hr
running 3.7G
root@vmh02:/boot#
GOSPODARSKA KOMORA

Go go http://bit.ly/dc14-ganeti

- open: https://github.com/lblasc/dorscluc2014-ganeti#dorscluc-ganeti-workshop

- will be using latest Ganeti from wheezy-backports (2.10)

- http://docs.ganeti.org/ganeti/2.10/html/install.html#ganeti-installation-tutorial

SSH to machine

ssh root@***hostname***.dhcp.ffzg.hr

- password
- change password :D

Hostname

- ganeti needs fqdn in hostname:
- run:

echo “***hostname***.dors.cluc” > /etc/hostname

hostname ***hostname***.dors.cluc

/etc/hosts

- should have valid hosts file:
- run:

echo “172.16.1.**XXX** ***hostname***.dors.cluc ***hostname***” >> /etc/hosts

echo “172.16.1.**1** cluster.dors.cluc” >> /etc/hosts

checkpoint

hostname -f # should work

XEN specific settings

- go to: http://docs.ganeti.org/ganeti/2.10/html/install.html#xen-settings

Limit amount of memory dedicated to hypervisor, add to /etc/default/grub:

GRUB_CMDLINE_XEN_DEFAULT="dom0_mem=512M"

Selecting the instance kernel

```
$ cd /boot
$ ln -s vmlinuz-3.2.0-4-amd64 vmlinuz-3-xenU
$ ln -s initrd.img-3.2.0-4-amd64 initrd-3-xenU
```

DRBD setup

echo “drbd minor_count=128 usermode_helper=/bin/true” >> /etc/modules

apt-get install drbd8-utils

Network setup

```
auto xen-br0
iface xen-br0 inet static
   address YOUR_IP_ADDRESS
   netmask YOUR_NETMASK
   bridge_ports eth1
   bridge_stp off
   bridge_fd 0
   up ip link set addr $(cat /sys/class/net/eth1/address) dev
$IFACE
```

Network setup

apt-get install bridge-utils

ifup xen-br0

LVM setup

apt-get install lvm2

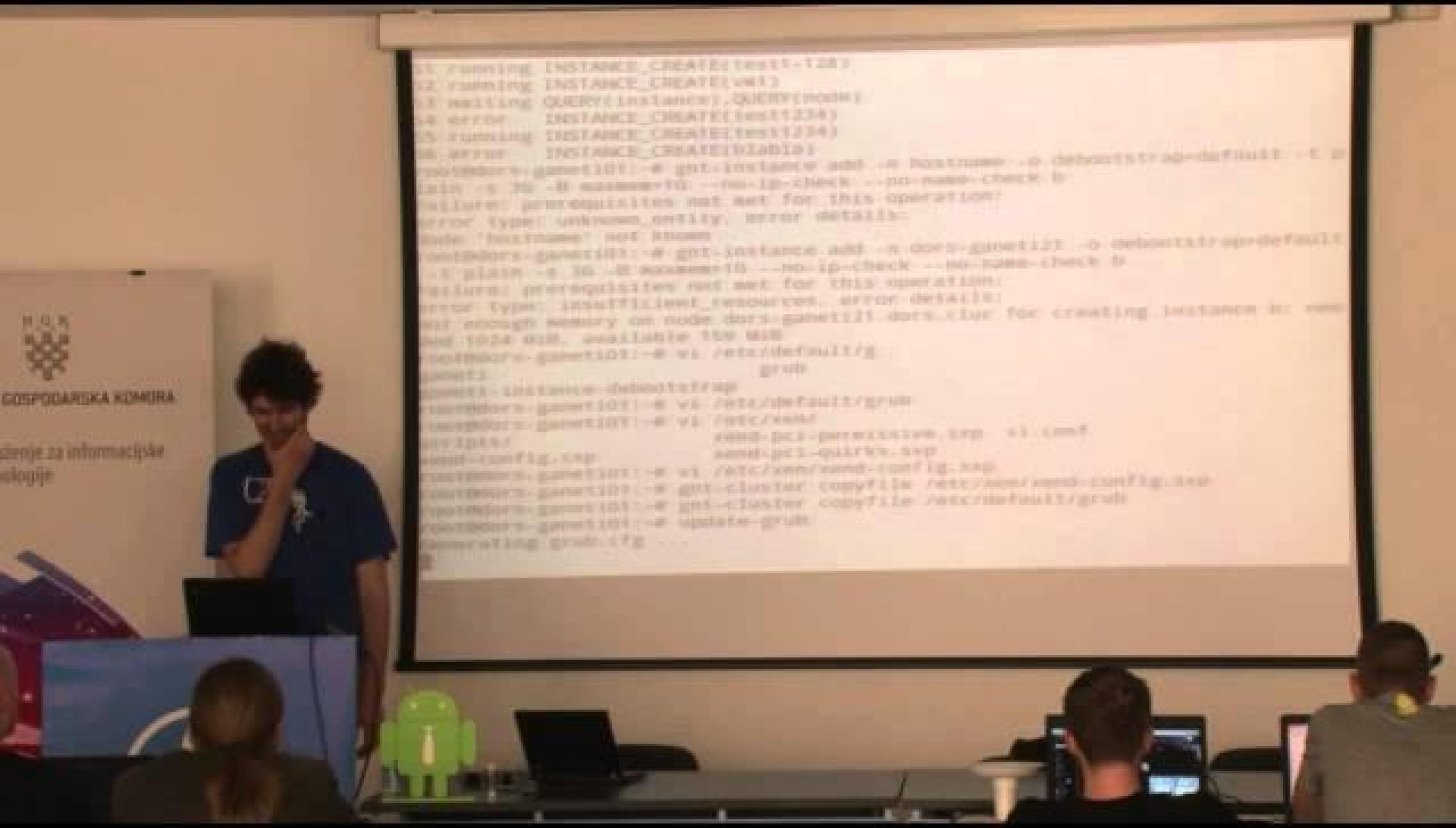
pvcreate /dev/sdc

vgcreate xenvg /dev/sdc

Install ganeti & instance-debootstrap

apt-get install -t wheezy-backports ganeti

apt-get install -t wheezy-backports ganeti-instance-debootstrap


GOSPODARSKA KOMORA

Initialize cluster

gnt-cluster init --vg-name xenvg --no-etc-hosts --master-netdev xen-br0 --enabled-hypervisors xen-pvm --primary-ip-version 4 cluster.dors.cluc

Initialize cluster

```
# set default memory and vcpu count
gnt-cluster modify -B vcpus=2,memory=512M
```

Add a second node

gnt-node add --master-capable=yes dors-ganeti20.dors.cluc

Create the instance

gnt-instance add -n ***hostname*** -o debootstrap+default -t plain -s 3G --no-ip-check --no-name-check myfirstinstance

Lets play

gnt-instance *

gnt-node *

hbal # load balance your cluster

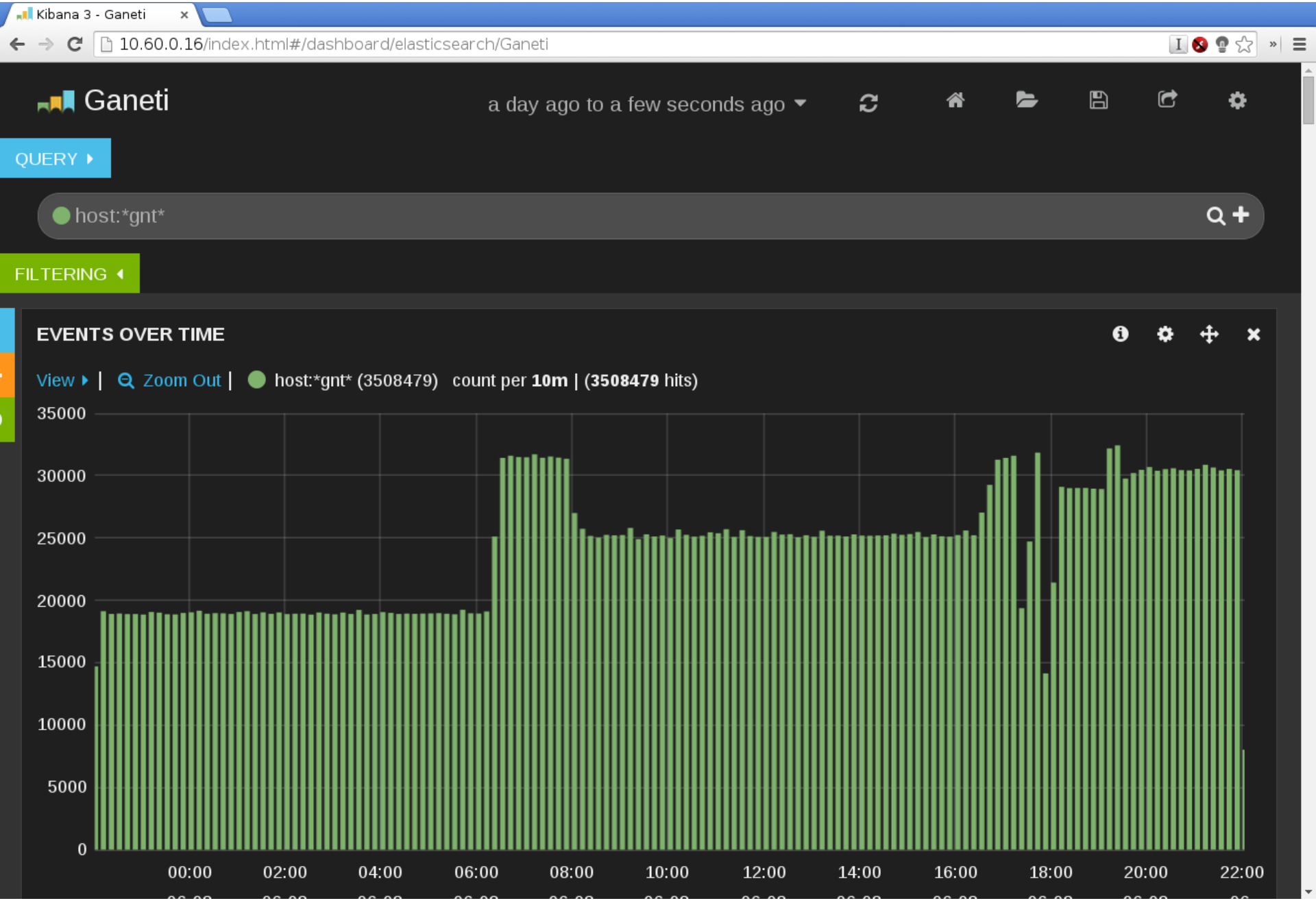
hspace # capacity planning on your cluster

-I hail # instance allocation tool

…..

Kibana, LogStash and ElasticSearch

```
dpavlin@kibana:/etc/cron.hourly$ cat kibana-drop-index
#!/bin/sh -xe

min_free=`expr 2048 \* 1024` # k

free() {
        df -kP /var/lib/elasticsearch/ | tail -1 | awk '{ print
$4 }'
}

while [ $(free) -lt $min_free ] ; do

curl http://localhost:9200/_cat/indices | sort -k 2 | grep
logstash- | head -1 | awk '{ print $2 }' | xargs -i curl -
XDELETE 'http://localhost:9200/{}'

done
```


Kibana 3 - Ganeti
10.60.0.16/index.html#/dashboard/elasticsearch/Ganeti
Ganeti
a day ago to a few seconds ago
QUERY
host:*gnt*
FILTERING
EVENTS OVER TIME
View | Zoom Out | host:*gnt* (3508479) count per 10m | (3508479 hits)
35000
30000
25000
20000
15000
10000
5000
0
00:00
02:00
04:00
06:00
08:00
10:00
12:00
14:00
16:00
18:00
20:00
22:00

Thx!