MTAT.08.021
Systems Administration

L3: Backup, DNS, and Network Boot

Lecture
2011-03-03
Attention:
Test next week

- March 10: Small test (20min) on Intro, Raid, Volume Management, Backup, DNS, and Booting in Network
- Participation is compulsory
- If you cannot take part because of serious reason, you have to agree with me on an alternative in advance (24 hours at least)
- If no arrangement with me has been done and you do not show up to test, you fail the test
How to do a backup?

• Excercise (15min)
  - You want to backup your (unix-based) servers and user directories hosted on these (also consider, how to offer a service for users not having user directories mounted)
  - Develop a strategy describing the steps you will do!
  - Mention tools/programs you will use.
How to do a backup?

• Did you consider
  - multiple backups
  - full backups
  - incremental backups
  - archiving/compressing backups
  - indexing backups for fast restore
  - restore service for users
  - scheduling backups
  - backup clients on user machines
  - failure in storage and recovery
Lots of tools today

• backup
  - tar
  - rsync
  - partimage
  - cron

• network
  - name resolution (bind)
  - setup
    • dhcp
    • bootp
    • pxeboot
Select your tool

• Be aware, this might be a choice for life
• So make sure to make a good balance between usability and openness
• How are backups stored?
  – Files?
  – Proprietary format?
  – Standard archives?
• How is the access time?
Backup Software - features

- Volumes
- Data compression
- Remote backup
- File locking
- Incremental backups
- Schedules
- Encryption
- Transaction mechanism
- Disk/partition cloning
- Synchronization
Mentionable proprietary tools

- ARCServe (big enterprise solution, long history)
- IBM Tivoli storage (big enterprise solution)
- Time machine (file based backup, very intuitive graphical user interface – good for desktops)
- Norton Ghost (Image copy of systems – good for OS backup)
Free Sysadmins Toolbox

- tar (+ gzip, bzip2, lzma/xz)
- rsync
- dd
- partimage
- (cron)

- unison
- DirSync Pro (just to have also one with gui)
- bacalu (free enterprise backup software)
archiver

- tar, cpio, gzip, bzip2, xz/lzma, zip, p7zip
- archive file system in one other file or in one data stream
- tar, cpio – no compression but voluming
- gzip, bzip2, xz/lzma – only compress one file or data stream – xz/lzma highest compression (usually used in combination with tar)
- zip – not state of the art compression, allows fast access and no permission storage
- p7zip standalone, allows to archive directory, no permission support, high compression
tar

- like cpio based on tape devices
- no random access
- compression later with other tools or via pipe (or builtin in gnu tar)
- common archive format in unix
- used in debian and ubuntu archives
tar examples

• Archive directories /home and /etc with gzip compression
  - tar -czf /root/home_etc.tgz /home /etc

• List contents of tar archive
  - tar -tzf /root/home_etc.tgz > /root/index.txt
  - less /root/index.txt

• Unpack bzip2 tar archive and list files
  - tar -xjvf /root/test.tar.bzip2

• Use exclude pattern
  - tar -cf images.tar --exclude='*.avi' pics
rsync

- Swiss army knife of sysadmin
- Very versatile copying tool
- Allows also remote sources and or destinations, works via ssh
- Synchronization
- Incremental copies
- Copies only what is necessary
- Powerful filters

rsync examples

- `rsync -av /fromdir /todir`
- `rsync -avz /usbdisk/ user@somemachine:/backups/`
- `rsync -avz --del /usbdisk/ user@somemachine:/backups/`
- be aware, this copies the directory itself
  - `rsync -avz /usbdisk user@somemachine:/backups/`
- `rsync --backup --backup-dir=`\`date +%Y.%m.%d`` -a \
  /data/working/ /BACKUP/`
rsync filters

- --exclude "*.jpg" – exclude all filenames matching *.jpg
- --exclude "/foo" – exclude a file called foo in the transfer-root directory
- --exclude "foo/" – exclude any directory called foo
- --exclude "/foo/*/bar" – exclude any file called bar two levels below a directory called foo in the transfer-root directory
- --exclude "/foo/**/bar" – exclude any file called bar two or more levels below a directory called foo in the transfer-root directory
- --include "*/" --include "*.c" --exclude "*" – include all directories and C source files
- --include "foo/" --include "foo/bar.c" --exclude "*" – include only foo/bar.c (the foo/ directory must be explicitly included or it would be excluded by the "*" )
clone system with rsync

- Make copy with rsync -av, exclude /mnt, /media, /proc, /sys
- Create on destination excluded directories
- Mount with bind dev-directory to destination
- chroot to destination
- Mount proc and sys
- Install and update grub (or other bootloader)
- Unmount
- Reconnect and reboot (have rescue cd handy to eventually repeat steps after chroot)
**dd**

- easy tool to copy partitions or whole disks
  - `dd if=/dev/sda1 of=/mnt/newdisk/diskimage.img`
  - can be later mounted with `mount -o loop /mnt/newdisk/diskimage.img /mnt2`

- or create empty images/swapfiles
  - `dd if=/dev/zero of=myfile bs=1024k count=10`
  - (could be also `/dev/random` if input for encryption)

- pipe to the network
  - `dd if=/root/myfile|nc remotehost.ulno.net 3000`
  - receive with: `nc -l -p 3000|dd of=receive`
partimage

- Advanced tool to make images from partitions
- Comes with rescue-CD
- Very similar to Norton Ghost
(cron)

- Schedule commands at specific times
- Call as user: crontab -e
  - this will edit the respective crontab of a user
- Format:
  - minute hour “day of month” “month” “day of week” command
  - (0-59) (0-23) (1-31) (1-12 or Jan-Dec) (0-6 or Sun-Sat)
  - special strings: @reboot, @yearly, @annually, @monthly, @weekly, @daily, @midnight, @hourly
  - use comma for several, */n for “for each n’th”
- Is active after exiting editor
- Nice for scheduling tar or rsync commands
- If output produced → sent via email to executing user
  (avoid with piping to /dev/null, use 2>&1 to redirect stderr)
crontab example

* * * * * /sbin/ping \-c 1 192.168.0.1 > /dev/null

0 4 * * Mon find /home

0 0,12 1 */2 * /sbin/ping \-c 192.168.0.1; ls -la >> /var/log/cronrun

PATH=/usr/local/sbin:/usr/local/bin:/home/user1/bin

MAILTO=user1@nowhere.org,user2@somewhere.org

0 2 1-10 * * du \-h --max-depth=1 /

Some examples taken from http://www.pantz.org/software/cron/croninfo.html
Dir Sync Pro

- Powerful synchronization utility
- Java, runs standalone, also from usb-stick
- Userfriendly GUI
- Schedule engine
- Bi-directional
- Incremental backup
- Symbolic link support

Screenshot taken from and program available at http://www.dirsyncpro.org/

Ulrich Norbisrath (http://ulno.net)  
University of Tartu  
http://sysadmin.ulno.net  
2011-03-03
bacula

- Enterprise ready backup, restore, and verification tool suite
- Tape drive and autochanger support
- GUI and Web interface

Images and software available from http://www.bacula.org
Versioning as Backup?

- Very good for code and text but also for user data
- Development repositories
- Configuration files

- Previously rcs (co, ci, -l for lock)
- Today git, maybe mercurial → nice because they also can push data to remote repository
Cloud storage

• Increasingly popular to make backups to cloud or sync folders across multiple computers with a central cloud storage

• Examples
  - Desktop sync
    • dropbox
    • sugarsync
    • google docs
  - Amazon S3 ← very customizable, more for offering your own service
  - Any hosting solution
  - … many more – nearly each day …
How would you do this?

1) Synchronize two desktops, both behind a firewall?
2) Synchronize two systems connected via network - one of them with an open ssh port.
3) Have an incremental archive of your data?
4) Have different automatic backups for your system (full backups, weekly backups, depending on age).
5) Clone an old Linux system with a 160GB harddisk to new Linux system with 2x 2TB harddisk.

- Think for 15 minutes, compare with neighbor (5min), select best idea each → two total, I will select randomly people to present
Domain Name System (DNS)/Name resolution

- metaphor: phonebook for Internet
- Lookup: hostname → IP(s)
- Reverse Lookup: IP → hostname(s)
- DNS distributes responsibility by authoritative name servers for each domain.
- Authoritative name servers responsible for their particular domains, can assign other authoritative name servers for their subdomains
- -> distributed and fault tolerant
Example responsibility DNS

- ee domain – responsible eenet
- ut.ee – University of Tartu

- try out host -C <name>
  - ee: ns.tld.ee
  - ut.ee: ns.ut.ee
  - cs.ut.ee: ns.ut.ee
Domain Name Space

```
| NS RR ("resource record") | resource records associated with name |
| names the nameserver authoritative for delegated subzone | zone of authority, managed by a name server |
| "delegated subzone" | see also: RFC 1034 4.2: How the database is divided into zones. |

When a system administrator wants to let another administrator manage a part of a zone, the first administrator's nameserver delegates part of the zone to another nameserver.
```
Domain Name Syntax

• Example: sysadmin.ulno.net (label.label.label)
• Right most label: top-level domain (TDL)
• Hierarchy: right to left, to the left: subdivisions or subdomains → up to 127
• Each label up to 63 chars, full name max 253 chars
• Any character allowed, but A-Za-z0-9 and hyphen preferred
• Hostname: domain name with at least one IP
Authoritative name server

- Gives answers configured by an original source
- Either be a master server or a slave server
  - Master server: original (master) copies of all zone records
  - Slave server: automatic updating mechanism of DNS protocol for identical copy
- Every DNS zone ↔ set of authoritative name servers installed in NS records in the parent zone.
- New domain names: primary name server and at least one secondary name server
- Primary name servers often master name servers
- Secondary name server may be slave servers.
- Status indication by software flag called the Authoritative Answer (AA)
Recursive and caching name server

- Authoritative NS would be enough
- To improve efficiency: caching NS
- Store DNS query results for a period of time – time-to-live (TTL)
- They can resolve them selves recursively starting at root-servers
DNS resolver

- Client side – contact to NS
bind

- Dominant DNS Server on the Internet
- Also called named
- Berkeley Internet Name Domain (BIND) Server (1984)
- BIND 9 more secure (2000)
bind configuration

• Configure /etc/named.conf
• Zone file with Type A records (name to IP)
• Reverse zone file with PTR records (IP to name)
• Also specify TTL (take a small one in lab)
DHCP

- Dynamic Host Configuration Protocol
- Instead of configuring each computer individually – why not configure via the network from a central point → less work for sysadmins (good!)
- Assigns IP-address, netmask, router, dns server(s) automatically (time servers and more)
- Can be dynamic, based on prior configuration, MAC address, or hostname request
DHCP

- Extension of BOOTP (Bootstrap Protocol)
- Request-and-grant concept with leases (a time how long addresses are valid)

- Invoke client (examples)
  - dhclient eth0
  - dhclient3 ethusb
  - pump -i eth0
  - (must often be killed if unsuccessful)
DHCPd

- Dhcpd reference implementation from Internet Software Consortium (ISC)
- Configure dhcpd.conf
- Leases in /var/lib/dhcpd/dhcpd.leases (can be removed before restarting)
DHCPd example

```
[...]

ddns-update-style none;  Interesting for dynamic dns

subnet 192.168.1.0 netmask 255.255.255.0 {
  range 192.168.1.100 192.168.1.200;
  option subnet-mask 255.255.255.0;
  option broadcast-address 192.168.1.255;
  option domain-name-servers 123.123.123.10, 123.123.123.20;
  option routers 192.168.1.1;
}

host box1 {
  hardware ethernet 00:50:AB:AB:AB:AB;
  fixed-address 192.168.1.7;
}

host winbox1 {
  hardware ethernet 00:06:CD:CD:CD:CD;
  fixed-address 192.168.1.8;
}
```

Pool of 100 addresses

Special hosts

Taken from dhcpd howto

Ulrich Norbisrath (http://ulno.net)
University of Tartu

Systems Administration
http://sysadmin.ulno.net

2011-03-03
38/47
DHCP configuration extensions

• Multiple subnets
• Netbios server infos for Windows
• For network boot:
  host diskless
  {
    filename  "/dwboot/diskless.nb";
    hardware ethernet  00:04:5A:4F:8E:47;
    fixed-address  10.0.10.2;
  }
Network Boot

- Configure even more automatically!
  - Boot the whole system from the network

- Involved techniques:
  - bootp
  - tftp
  - pxeboot (pixie)

- Valuable site:
  - etherboot/gpxe (rom-o-matic) – generate firmware for your nw-card
BOOTP

- Bootstrap Protocol (pre-dates DHCP) – also for obtaining IP-address
- Usually used while booting – especially for diskless workstations
- Often was invoked from floppy disk
- Most dhcp servers offer the bootp-protocol subset
TFTPD

• Trivial File Transfer Protocol
• Much simpler than ftp (very limited in comparison)
• Used to transfer small amount of data in networks, like firmware, or initial bootcode
• No security → should only be used in local networks
• Part of PXE implementation
Preboot Execution Environment (PXE)

• Environment for booting computers through a network connection – independent of local storage
• Introduced by Intel and Systemsoft in 1999
• Steps, when booting
  - Firmware on client tries to locate a PXE redirection service (Proxy DHCP)
  - Ask appropriate boot server for file path of a network bootstrap program (NBP)
  - Download NBP using TFTP and execute
dnsmasq

- bind, dhcp, bootp easy!
- Very easy for homenetworks (included in lots of routers)
- Runs basically out of the box, uses /etc/hosts for name resolution
- Takes dhcpd parameters on command line
- Supports also BOOTP, tftpd, pxeboot
**dnsmasq example**

```
/usr/sbin/dnsmasq \
    --bind-interfaces \ 
    --interface=eth1 \ 
    --domain=ulno.net \ 
    --dhcp-range=102.168.1.50,192.168.1.100,12h \ 
    --dhcp-host=00:16:01:A4:E7:F0,192.168.1.41
```
Network boot demo

- Demo of boot via pxe in virtualbox network. One virtualbox server, one virtualbox client.
Questions and Discussion

• Any ideas, comments, or questions?