

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

Lots of info is stolen from <http://fai-project.org> , so all credits go to those people.  
I just "restructured" some stuff, :-)

## 0. INTRO

### a. Scope/abstract

In my company, we are looking for a provisioning system to deploy 3 OS'ses on virtual servers (phase I). Later, we want to deploy physical servers too (phase II). FAI seems very promising. Unfortunately, we use Microsoft Hyper-V as virtualization platform to run virtual Linux machines.

### b. Objective/objective/purpose/goal

The goal of this blog post, is to document all the steps necessary, to implement and use FAI. I want to have a FAI installation server (virtual)

*NAME: fai*

*IP: 192.168.118.30*

that can automatically deploy virtual FAI client machines. In this example, I use:

*NAME: demohost*

*IP: 192.168.118.31*

with the following OS's:

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- *Ubuntu 8.04 LTS 32-bit and 64-bit*
- (- *Ubuntu 10.04 LTS 32-bit and 64-bit*)
- *CentOS 5.5 32-bit and 64-bit*

as guests on Microsoft Hyper-V ! (brrrr, I now :p )

( and on the following physical hardware:

- *Dell PowerEdge 1850/1950/2850/2950*
- *Dell R410/R610/R710 )*

Customers will be able to select a couple of parameters:

- distro: ubuntu804 or centos55
- memory: e.g. 2096 (in megabytes)
- packages: apache, mysql, ...

### **c. Ways of (re)building/restore configurations**

Here's a short overview on the (dis)advantages of some solutions:

#### **Disk images.**

-> make template from "clean" installation

++ low learning cost

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

++ simple and fast (cp, dd, rsync, CloneZilla, Symantec Ghost / g4u (Ghost for Unix), ...)  
++ some can work in broadcast mode  
++ ...

-- inflexible: the smallest change requires rebuilding the image  
-- still manual work needed to get an installed system  
-- image hell = different config = another image: you need 1 image per HW/SW configuration  
-- storage cost linear to number of different configurations (especially if you a keep history of changes)

### Home grown shell scripts.

++ manual work replaced by shell scripts  
++ much lower storage cost  
++ higher flexibility  
++ tailor made  
++ ...

-- full-blown development project (you'll realize after a while)  
-- you solve every problem on your own, instead of reusing work of others  
-- ...

### Available auto-installers.

++ choice: diverse client and server distro's possible

- Anaconda / Kickstart: Red Hat, Fedora, CentOS
- AutoYast: Suse
- Nlite/Unattend: Windows in many flavours
- Jumpstart: Solaris
- Cobbler/Koan: Red Hat, Fedora, CentOS
- OpenQRM
- FAI: Debian, Fedora/Red Hat/CentOS, SuSE, Windows experimental, Solaris possible

++ multiple installation types/updates  
++ bare metal -> fully working  
++ good integration with config management tools (cfengine, puppet, chef, custom scripts/hooks/files/configs...)  
++ ...

-- kickstart files: One Kickstart configuration file per hardware/purpose combination  
-- after installation all configuration still manual in-place  
-- rely on back-ups in case of failure

### d. I choose FAI. What is FAI ?

#### **My definition.**

"A non-interactive system to install, customize and manage Linux systems and software configurations on physical computers as well as virtual machines and chroot environments, from small networks to large scale infrastructures (rendering farm, classrooms) and clusters with several thousands of systems"

a.k.a

- Fully Automated Install
- Mass Unattended Linux Installation
- Automatic Systems Installation and Change Management
- Auto-Installer
- Non-Interactive Setup Tool
- Remotely Triggered Re-installation
- Automated Linux Management Infrastructure

#### **Motivation & History.**

- started by Thomas Lange at university of Cologne in 1999
- had to install cluster with 1 server and 16 computers, knew Solaris Jumpstart, started something for Debian
- manual repeating task = boring = error prone = unreproducible (humans make errors) = unefficient (time) = unscalable (clusters) = no long term thinking = no central infrastructure management <-> automation
- 10 active devs, small but nice community
- success stories:
  - City of Munich, several hundreds, (14.000 hosts planned)
  - Linux Information Systems AG, >1000 hosts
  - Albert Einstein Institute, 800 hosts
  - Lycos Europe, 3000 hosts
  - Host Europe, 250 hosts
  - Thomas Krenn AG, 500 per month
  - MIT Computer science research lab, 200 hosts

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- Electricit  de France (EDF), France, 200 hosts
- France Telecom, TRANSPAC, France, 300 hosts
- Danmarks Meteorologiske Institut, 85 hosts
- OPIT Solutions AG, Swizerland, 80 hosts
- University of New Orleans, USA, 72 node Beowulf cluster
- Brown University, Dep. of Computer Science, USA, 300+ hosts
- Ewetel, ISP and telco, 65 hosts at 3 locations
- High Performance Computing Center North, HPC2N, 2 clusters of top500.org, 192 dual Opteron, 120 dual Athlon
- Netcologne, Pironet, Linuxhotel, ZEDAT FU-Berlin, DESY, mc-wetter.de, ALTANA Pharma AG, Networking4all BV, Move Next BV, Belgacom NV/SA, easynet GmbH, Minick AG, Technische Universit t M nchen
- The Centibots project (100), Lucidor Cluster (90), Genome research cluster (168), Merlin Cluster (180), IITac cluster (400+), Computer science labs (400+), LiMUX in Munich (400 to be 14000), multiple top 500 performance clusters and also SOHO with 2 PC's, ...
- My company ? :-)

## FAI <-> Puppet.

- FAI <-> puppet can distribute "secrets" (has data encryption functionality)
- FAI has simpler file copying structure <-> could be "emulated" with some code in Puppet
- FAI partitions and does base-install <-> Puppet only does "configuration"
- FAI has multiple specific logs <-> Puppet logs to syslog
- FAI: config files + scripts <-> Puppet: everything is a script
- FAI: (scripted) ssh into amanged host, additional tool <-> Puppet: advanced client/server config push
- FAI: a bit more mature - no youth problems <-> Puppet
- FAI <-> Puppet has some more advanced distribution abstractions
- FAI is still lacking a nice logo and website :-> <-> Puppet

## BUT

- both cool tools that you should use
- can work together well and easy

## Outlook.

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- support stuff for more distributions: fai-distributions
- GOSA as LDAP and FAI GUI
- automatic tests of the installed systems
- better configspace layout
- ...

### Components/Requirements.

#### 1. Install server

- Services:
  - DHCP (Host/IP/MAC as usual)
  - TFTP
  - FAI
  - (DNS)
  
- (local mirror)
- configuration space = "config space"
  
- defines
  - the hard disk layout
  - file system type/mount points/mount options
  - software packages
  - keyboard layouts, time zones, NIS, Xorg configuration, remote file systems, user accounts, printers, ...
- ...
  
- simple text files and scripts that reflect the requirements from your installation plan
- location: /srv/fai/config
- layout:

```
fai:/srv/fai/config# tree -d -L 1
```

```
.
```

|-- basefiles (minimal base images for non-Debian distributions and special uses, named by classes e.g. UBUNTU804\_64.tar.gz)

|-- class (see further)

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

|-- debconf (presets for package install scripts, only for dpkg installs, analogue to Debian installer)  
|-- disk\_config (detailed control over partitions and mounting, including LVM/SW RAID, hardsize or ranges)  
|-- files (structure of a filesystem, starting with /, class based copy/unpack, for usage with fcopy/ftar, copy single files explicitly, or recursive from /  
|-- hooks (execution before the according task)  
|-- package\_config (defines packages to be installed, files named by class names, supports apt-get, aptitude, yum, taskinst, yast, ...)  
|-- scripts (scripts to be executed after package installation: bash, perl, cfagent, expect, others..., naming scheme defines order of execution)  
`-- tests

- NFS root (which the clients mount)

## 2. Install client

### Why I choose FAI ? Advantages.

- lots of powerful features:
- very fast rollout, deployment, provisioning and to some extend configuration management of systems (according to #packages 3 to 30 minutes)
- physical or virtual (Xen, VMWARE, ...) or chroots
- diff distro's (Debian, Ubuntu, Mandriva, Suse, Solaris, Fedora, Windows experimental ?)
- diff install types:
  - network (PXE) and BOOTP
  - CD/floppy/USB stick
  - directly calling "dirinstall" for chroots (can be easy integrated into xen-tools, ganeti, ...)
  - fai-cd / fai-usb (environments without network)
  - with grml-live: Live-CD generation
- diff architectures: i386, AMD64, PowerPC, Alpha, SPARC and IA64
- diff bootloaders: LILO and GRUB
- diff filesystems: EXT2/3/4, ReiserFS, XFS
- GUI for FAI using GOsa
- diff scripting languages: shell, perl, expect, cfengine support
- optional full remote control via SSH during install + 2 additional virtual terminals available for debugging
  - LVM, software RAID, ...
  - softupdates possible (since 2005): update running system (daily maintenance)

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- (automatic) restart with production configuration
- integration with config management tool (puppet, chef, cfengine)
- automatic hardware detection
- ...
- ...
  
- simple architecture: everything is a "shell script" (simple, proper readable and writable configuration format (no XML), version controllable (svn, cvs, git...))
- open development model & community support by "seasoned installers"
- automatic documentation in central config repo (the "config space")
- no worries about easily...
  
- getting a config cloned (reproduce an EXACT same system)
- setting up a new machine
- restoring after a crash (advanced disaster recovery system)
- changing configs on multiple systems
  
- very flexible and easy to extend:
  - customization via scripts, hooks and configs (e.g. partition layout) to suit your needs: "swiss army knife for a experienced sysadmin"
  - class based approach: multiple definitions:
    - a class defines system properties and actions to be taken
    - a class is in fact an "attribute"
    - a class abstracts HW and SW configuration (bundles scripts, configs/settings, custom set of packages)
    - a class contains: class- and variable definitions/assignments
    - a class can be assigned:
      - statically: based on hostnames (in its simplest way)
      - dynamically: based on specific HW, IP, MAC, disk size, ... (any script can be used to assign a class)
  
- clients can be assigned multiple classes, combined at will, or assigned no class at all
- considerations/pitfalls:
  - KISS: don't define too many classes !
  - keep software classes task-oriented
  - you can make your own script to determine most classes
  - Some "built-in" classes the software determines automatically
  -



## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- examples:
- physical: based on common location e.g.
  - RC\_GEN2 (Room Class GEN2)
  - RC\_GEN4
- architecture: a separate kernel with hardware specific drivers for each computer type is needed e.g.
  - CC\_HP (Computer Class HP computers)
  - CC\_DELL
  - CC\_DELL\_1X50
  - CC\_DELL\_2X50
  
  - CC\_VMWARE
  - CC\_HYPERV
- software:
  - the parts that are common throughout all your systems, all the core software packages/configs/files/scripts you want to install on every host, go to e.g.
    - BC\_LINUX (Basic Class for ALL FAI clients)
    - BC\_KDE
    - BC\_GNOME
    - BC\_X11
  
  - prepare a client for a particular service
    - SC\_LDAP (Service Class for authentication to your OpenLDAP directory service)
    - SC\_LAMP (Service Class for Linux/Apache/MySQL/PHP/Perl/Python/Postfix/ProFTP/P...)
  
    - SC\_DNSCACHE\_SERVER
  
  - bundle set of apps that define a program/project

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- PC\_HBTOOL (Program Class HB tool)
- PC\_BOINC

- role
- others:
- RESTRICTED\_SHELL\_ACCESS

### Disadvantages.

- very high learning curve: difficult for beginners, trust me, I now :-)
- heavier burden on the network during simultaneous installation (NFS)

### Considerations.

-> motto: "plan your installation, and FAI installs your plan", so I have to plan carefully !  
Structured approach required: Installation and administration is a process, not a product !!

- PLANNING THE FAI SERVER INSTALLATION:
- install and document the FAI installserver installation  
-> OK (this website)
- decide the matching install type (net/cd/dirinstall)

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- > OK (netinstall)
- install and configure PXE boot / TFTP
  - > OK (see 1. Installation of FAI server)
- install and configure NFS server
  - > OK (see 1. Installation of FAI server)
- install and configure DHCP: how will we get the to-deploy-hosts into /etc/dhcp3/dhcpd.conf ?
  - > OK (see 1. Installation of FAI server)
  
- 
- Ubuntu/CentOS mirrors:
  - how do we setup mirrors of Internet software repositories (debmirror, ...) ? full ? partial ?
    - > apt-mirror, full
  - will we mirror security update repositories?
    - > yes
  - will we use package cache repo's (apt-proxy) ?
    - > not in phase 1
  - updates:
    - will we deploy updates ? )> NO
    - how and when should which patches be applied? -> NO
    - testing processes: how do we know the effect of a patch/update? -> NO
    - automatic (regular scheduled, timebased) or manually started? -> NO
  
- PLANNING FAI CLIENTS INSTALLATION:
  - Classes
  - Which computer groups ? Which hardware configuration ? Which software classes ?

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- 
- Beowulf cluster or some desktop machines ?
  - > only virtual servers
- special kernel needed ?
  - > NO
- MS Hyper-V:
  - Legacy/Emulated drivers
  - can do PXE boot
  - supported by default, but speed problem: max. 10/100 Mbps, no 1000 Mbps + slower as everything has to pass Hyper-V before reaching the Hypervisor
  - NO need for integration components
  
- Synthetic drivers
- can NOT PXE boot !
- talk straight to the Hypervisor instead of going through Hyper-V, so faster and 1000 Mbps available
  - you DO need integration components
  - the Microsoft integration components are in fact 4 modules: hv\_vmbus, hv\_storvsc, hv\_blksvc, hv\_netsvc (or something like that: TO CHECK)
  - Microsoft officially supports Red Hat and Novell Suse
  - those integration components are available in the Linux kernel version  $\geq 2.6.30$
  
- CentOS 5.3 / 5.4 have a kernel  $< 2.6.30$ , but source code is available to build these modules (MS support for Red Hat...).
- CentOS 5.5 ... ???
- Ubuntu 8.04 has kernel 2.6.24 so  $< 2.6.30$  but NO source code available (no MS support)
  
- Ubuntu 10.04 has kernel 2.6.32, so the integration components are already in the kernel
  
  
  
  
  
  
  
  
  
  
- physical hardware:
  - uniform hardware ?
    - > YES: these models: Dell PowerEdge 1850/1950/2850/2950 and Dell R410/R610/R710
  - will it stay uniform ?
    - > NO: in a few years, the models will change
  
- varia:

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- user accounts ?
  - > YES: the root user (+pass) and a company user (+pass)
- additional file and config adjustments ?
  - > YES: /etc/apt/sources.list (Ubuntu), ..., ...
- printers ? -> NO
- a mail system ? -> NO
- cron jobs ? -> NO
- graphic cards -> NO
- dual boot ? -> NO
- NIS ? -> NO
- NTP ?
  - > YES: client hosts will connect to a central timeserver
- timezone:
  - > Europe/Brussels
- keyboard layout:
  - > be-latin1
- exporting and mounting directories via NFS ?
  - > not after installation
- ...
- ...

## - CONFIGURATION OF COMPUTER GROUPS/SOFTWARE CLASSES

- hard disk partition scheme
  - > easy:
    - /boot, 100 MB
    - swap, 2 times the amount of RAM with a maximum of 4 GB swap
    - /, use the remaining free space
- package selection
- LAN topology ? network and environment ? network configuration
  - > for the moment, very simple: 192.168.118.0/24
- create shell scripts

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- PLANNING CONFIGURATION OF SOFTWARE CLASSES
- which apps ? what daemons ?

### e. Varia:

- I learned that I don't need a separate root NFS to deploy multiple distro's (source: IRC, Mrfai)
  - > but on the other hand, [THIS](#) learns me that you have to create a separate NFS root if you want to deploy 32-bit systems from a 64-bit fai server... ???
- I do not have a DNS server as for testing purpose, I will keep it simple and use /etc/hosts for now
  - Our devs will SSH into the fai server, and run a script with these values (and others) as parameters:

```
dev@fai:~# fai:~# ./fai_deploy.sh "hostname mac ip password_root reguser  
password_reguser distro memory arch packages"
```

We need hostname, mac and ip for DHCP configuration. Root password, regular user and it's password speak for themselves. Distro, arch, packages is used to know to which class(es) the client system belongs. Memory is used to know how big the swap must be.

- ...

## 1. INSTALLATION of FAI-SERVER

**a. first of all, Ubuntu's 8.04 repo package for fai-quickstart is way too outdated:**

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

```
root@a-hardy-server:~# apt-cache show fai-quickstart | grep Version
Version: 3.2.4+svn4837-0ubuntu2
```

and adding " [deb http://fai-project.org/download lenny koeln](http://fai-project.org/download/lenny/koeln) " to /etc/apt/sources.list, "apt-get update" and simulate an install via "apt-get install fai-quickstart -s" results in:

```
root@a-hardy-server:~# apt-get install fai-quickstart -s
Reading package lists... Done
Building dependency tree
Reading state information... Done
Some packages could not be installed. This may mean that you have
requested an impossible situation or if you are using the unstable
distribution that some required packages have not yet been created
or been moved out of Incoming.
```

Since you only requested a single operation it is extremely likely that the package is simply not installable and a bug report against that package should be filed.

The following information may help to resolve the situation:

```
The following packages have unmet dependencies:
fai-quickstart: Depends: syslinux-common but it is not installable
E: Broken packages
```

so that's why I installed my virtual fai server as a Debian 5.0 aka Lenny, so I can use the most recent FAI 3.5.5 version.

### **b. put in /etc/hosts**

```
127.0.0.1    localhost
192.168.118.30 fai.mydomain.com fai
192.168.118.31 demohost.mydomain.com demohost
```

**c. add " [deb http://fai-project.org/download lenny koeln](http://fai-project.org/download/lenny/koeln) " to /etc/apt/sources.list**

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

### d. add his key;

```
fai:~# gpg -a --recv-keys AB9B66FD
gpg: requesting key AB9B66FD from hkp server keys.gnupg.net
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key AB9B66FD: public key "Thomas Lange <lange@debian.org>" imported
gpg: no ultimately trusted keys found
gpg: Total number processed: 1
gpg:    imported: 1
```

### e. add to chain:

```
fai:~# gpg -a --export AB9B66FD | apt-key add -
OK
```

### f. apt-get update

### g. apt-get install fai-quickstart

-> choose OK at the DHCP dialog box (non-authoritative stuff)

==> installation successfull <==

## 2. CONFIGURATION:

### DHCP:

#### a. you will see during install that DHCP failed to start:

```
Starting DHCP server: dhcpd3check syslog for diagnostics. failed!
failed!
invoke-rc.d: initscript dhcp3-server, action "start" failed.
```



## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

-> put "eth0" in /etc/default/dhcp3-server

-> backup the original DHCP config first + put a subnet declaration in /etc/default/dhcp3-server together with the MAC-address/IP of the demohost:

```
fai:~# more /etc/dhcp3/dhcpd.conf
option domain-name "mydomain.tld";
option domain-name-servers X.X.X.X;
option routers 192.168.118.254;

subnet 192.168.118.0 netmask 255.255.255.0 {
}

allow booting;
next-server 192.168.118.30;
filename "pxelinux.0";

include "/etc/dhcp3/demohost.dhcp";
```

and

```
fai:~# more /etc/dhcp3/demohost.dhcp
host demohost {
hardware ethernet 00:1D:D8:B7:1D:11;
fixed-address 192.168.118.31;
option host-name "demohost";
}
```

-> now DHCP will start:

```
fai:~# /etc/init.d/dhcp3-server start
Starting DHCP server: dhcpd3.: OK
```

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

### FAI:

#### **b. edit config file: /etc/fai/fai.conf**

-> I know if undefined, the default value will be the same, but I want to be sure and explicitly define:

```
FAI_CONFIG_SRC=nfs://192.168.118.30/srv/fai/config
```

#### **c. edit config file: /etc/fai/make-fai-nfsroot.conf**

-> we don't have a DNS-server, so:

```
NFSROOT_ETC_HOSTS="192.168.118.30 fai"
```

-> we won't make a local Debian mirror for now, just for creating the +300MB NFSROOT, so:

```
FAI_DEBOOTSTRAP="lenny ☐☐☐ http://cdn.debian.net/debian"
```

#### **d. create the NFS-root: fai-setup -v**

- downloads some +300 MB of package data via http from the repo defined at /etc/fai/fai.conf
- /srv/fai/nfsroot/live/filesystem.dir will contain NFS root
- exports the dirs required for the client installation on NFS
- drops a matching kernel image into /srv/tftp

-> check the logfile: /var/log/fai/make-fai-nfsroot.log

-> my /etc/exports looks like this:

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

```
/srv/fai/config 192.168.118.30/24(async,ro,no_subtree_check)
/srv/fai/nfsroot 192.168.118.30/24(async,ro,no_subtree_check,no_root_squash)
```

### TFTP:

**e. during installation, you should notice "tftpd-hpa disabled".  
Make sure /etc/default/tftpd-hpa looks like this:**

```
#Defaults for tftpd-hpa
RUN_DAEMON="yes"
OPTIONS="-l -s /srv/tftp/fai/"
```

-> now it will start:

```
fai:~# /etc/init.d/tftpd-hpa start
Starting HPA's tftpd: in.tftpd.
```

### **f. prepare the tftp configuration file:**

```
fai:~# fai-chboot -IFv demohost
Booting kernel vmlinuz-2.6.26-2-amd64
append initrd=initrd.img-2.6.26-2-amd64 ip=dhcp
FAI_FLAGS=verbose,sshd,createvt
```

```
demohost has 192.168.118.31 in hex C0A8761F
Writing file /srv/tftp/fai/pxelinux.cfg/C0A8761F for demohost
```

### 3. INITIAL TESTDRIVE

PXE boot the demohost, and you will see it works !

#### How does it work ? Installation steps (tasks).

-> the whole installation phase can be monitored using **faimond**.

--- TO IMPROVE FURTHER ---

- client boots via PXE
- DHCP
- client -> broadcast: DHCPDISCOVER-extended: "Yo, I'm looking for a DHCP/PXE server"

- Protocol: UDP
- Source IP: 0.0.0.0
- Source port: 68
- Dest. address: 255.255.255.255
- Dest. port: 67
- DHCP option 53: DHCP Discover
- DHCP option 60: PXEClient:Arch:xxxxxUNDI:yyzzz

-

- server -> unicast: DHCPOFFER-extended: "Hey, I could serve you if you want"

- Procol: UDP
- Source IP: 192.168.118.30
- Source port: 67
- Dest. address: 255.255.255.255 (?)
- Dest. port: 68
- DHCP option 53: DHCP Offer
- IP: 192.168.118.31 ("You must use this IP address")
- DHCP option 1: 255.255.255.0 ("You must use this subnet mask")

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- DHCP option 3: 192.168.118.254 ("You must use this gateway")
  - DHCP option 51: 1 day ("This is your lease time")
  - DHCP option 54: 192.168.118.30 ("This is my IP as DHCP server")
  - DHCP option 60: "PXEClient"
  - DHCP-extensions options 43:
  - PXE\_DISCOVERY\_CONTROL: "How do you want to contact the bootserver?"
  - PXE\_BOOT\_SERVERS: "This is a list of available boot servers"
- 
- client -> broadcast: DHCPREQUEST-extended ("Cool, I'm looking for 'this type' of PXE server. Can you help me too ?")
- 
- Protocol: UDP
  - Source IP: 192.168.118.31
  - Source port: 68
  - Dest. IP: 255.255.255.255
  - Dest. port: 67
  - MAC: 00:1D:D8:B7:1D:11
  - DHCP option 53: DHCP request ("I'm looking for a DHCP/PXE server")
  - DHCP option 60: PXEClient:Arch:xxxxxUNDI:yyzzz
  - DHCP-extensions options 43:
  - PXE\_BOOT\_ITEM: "I'm looking for this type of PXE server)
- 
- 
- server -> **unicast**: DHCPACK-extended ("We're in business! I'm you're PXE server)
- 
- Protocol: UDP
  - Source IP: 192.168.118.30
  - Source port: 67
  - Dest. IP: **192.168.118.31**
  - Dest. port: 68
  - DHCP option 60: PXEClient
  - DHCP-extensions options 43:
  - MTFTP IP ADDR: 192.168.118.30 (IP of PXE server)
  - MTFTP Client UDP: 4011/UDP (???) (Portnr of PXE-service)

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- PXE\_BOOT\_ITEM: pxelinux.0 (name + path of the mini OS = NBP = Network Bootstrap Program)

- 
- TFTP
- client -> server: TFTP request for **pxelinux.0**

```
fai:~# ls -lh /srv/tftp/fai/
total 11M
-rw-r--r-- 1 fai fai 9.0M 2010-05-17 18:40 initrd.img-2.6.26-2-amd64
-rwxr-xr-x 1 fai fai 16K 2010-05-17 18:40 pxelinux.0
drwxr-xr-x 2 fai fai 4.0K 2010-05-17 19:07 pxelinux.cfg
-rw-r--r-- 1 fai fai 1.7M 2010-03-10 01:34 vmlinuz-2.6.26-2-amd64
```

```
fai:~# ls -lh /srv/tftp/fai/pxelinux.cfg/
total 4.0K
-rw-r--r-- 1 root root 292 2010-05-18 13:50 C0A8761F
```

- server -> client: pxelinux.0
- client: saves pxelinux.0 in RAM and executes it
- client -> server: request for PXE configuration file **pxelinux.cfg** using his MAC
- server -> client:
- Trying to load: pxelinux.cfg/564ddf52-be24-09bb-33bb-c1724d2d9b11 -> fails
- Trying to load: pxelinux.cfg/00:1D:D8:B7:1D:11 -> fails
- Trying to load: **pxelinux.cfg/C0A8761F** -> works, as this exists:

```
fai:/srv/tftp/fai# cat /srv/tftp/fai/pxelinux.cfg/C0A8761F
# generated by fai-chboot for host demohost with IP 192.168.118.31
default fai-generated
```

```
label fai-generated
kernel vmlinuz-2.6.26-2-amd64
append initrd=initrd.img-2.6.26-2-amd64 ip=dhcp root=/dev/nfs nfsroot=/srv/fai/nfsroot
boot=live FAI_FLAGS=verbose,sshd,createvt FAI_ACTION=install
```

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- client -> server: downloads Linux kernel and initial RAMdisk:
- Loading **vmlinuz-2.6.26-2-amd64**.....
- Loading **initrd.img-2.6.26-2-amd64**.....

```
fai:~# ls -lh /srv/tftp/fai/
```

```
total 11M
```

```
-rw-r--r-- 1 fai fai 9.0M 2010-05-17 18:40 initrd.img-2.6.26-2-amd64
```

```
-rwxr-xr-x 1 fai fai 16K 2010-05-17 18:40 pxelinux.0
```

```
drwxr-xr-x 2 fai fai 4.0K 2010-05-17 19:07 pxelinux.cfg
```

```
-rw-r--r-- 1 fai fai 1.7M 2010-03-10 01:34 vmlinuz-2.6.26-2-amd64
```

- client loads Linux kernel
- Linux kernel loads the initial RAMdisk
- Linux kernel does some hardware detection
- Linux kernel tries to figure out where the root file system is located
- network ? gets it from PXE configuration: *root=/dev/nfs* and *nfsroot=/srv/fai/nfsroot*
- CD/USB stick ? kernel probes removable devices (can be compressed filesystem e.g. squashfs)

- using package **live-initramfs** (only needed inside the *nfsroot* and adds some *initramfs* hooks)
  - client mounts its "root file system" via NFS from the install server READ-ONLY
  - root filesystem is made WRITABLE by mounting a RAM disk via *aufs* (another *unionfs*) on top of it (so it's possible for programs or daemons to write to files inside a read only mounted file system)
- > at this stage we have a fully functional network rescue system without using local harddisks

- start and setup FAI
- main FAI install script **/usr/bin/fai** begins (NO */etc/init.d/\** !!!)
- additional parameters are received from the DHCP daemon (??? what/which ones ???)
- configuration space is made available via - if choosen - NFS (by default)
- 2 additional virtual terminals are created

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- SSH daemon for remote access is started- if choosen
  
- TASK defclass:
  - DEFINE CLASSES: script `fai-class` is used to define classes by executing alphabetically  
`$FAI/class/` `[0-9]`  
`][0-9]*`
  
- each script echoes to "standard output" (output is ignored), but ...
- ... is interpreted as a "class name"
  
- every `.source` is sourced and can define new classes by adding them to `$newclasses`
  
- LOAD KERNEL MODULES: script `20-hwdetect.source` loads kernel modules on demand
  
- DEFINE VARIABLES: every file matching `*.VAR` with a prefix which matches a defined class is executed to define variables (e.g. `FAI-ACTION` if not set via `fai-chboot`, ...)
  
- TASK partition:
  - exactly 1 config is selected from `$FAI/disk_config` using classes
  - local disk(s) get partitioned using the newer tool **setup-storage**
  - disk layouts can be preserved (e.g. dual boot Windows, ...)
  - data on certain partitions can be preserved
  
  - partitions get filesystem and type (ext2/3/4, ...)
  - during installation, all local file systems are mounted relative to `/target` (`/target/home` will become `/home` in the new installed system)
  
- TASK extrbase:
  - a minimal base system is unpacked



## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- Debian
- or the one you packed yourself (e.g. UBUNTU804\_64.tar.gz)
  
- TASK instsoft:
  - command **install\_packages** installs all packages using apt-get, aptitude, ... without manual intervention and resolving dependencies.
  - (classes are also used when selecting the configuration files in `$FAI/config_package`)
  
- TASK debconf: read and apply Debconf preseedings
- ???
  
- TASK configure: site specific customization
  - arbitrary scripts which match a class name in `$FAI/scripts` are executed (which adjust the system configuration)
  - if `$FAI/scripts/classname` is a directory, all scripts that match `[0-9][0-9]*` in this directory are executed
  
- Automated tests
  - command **fai-do-scripts** will execute some tests if available: check for errors on the installation or of the softupdate - if choosen
  - it's messages are logged to `$LOGDIR/test.log`
  - a test can also define a new class for executing another tests during next boot via the variable `$ADDCLASSES`
  
- TASK savelog:
  - the log files are written to `/var/log/fai/$HOSTNAME/install/`
  - the logfiles are pushed via (scp by default but rsh, ftp possible) onto the account on the install server if `$LOG`  
`USER`  
is defined in  
`/etc/fai/fai.conf`
  - > another log saving destination is possible
  - two symlinks will be created to indicate the last directory written to

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- client boots OS from local disk
- at last the system is automatically rebooted if "reboot" was added to `$FAI_FLAGS` using its second boot device: the local hard disk (normally)
  - > to skip booting from network card, you can use the command **fai-chboot** to enable localboot.

So far for the Debian "demohost". Let's make Ubuntu work, shall we ? :p

--- TO IMPROVE FURTHER ---

## 4. NOW MAKE UBUNTU 8.04 LTS 64-bit WORK

-> checkout [http://wiki.fai-project.org/index.php/FAI\\_multi-distribution](http://wiki.fai-project.org/index.php/FAI_multi-distribution)

-> I learn from the manual: "You can install all sorts of Linux distributions from a single Debian nfsroot. Therefore you have to create a base.tgz of the distribution you like to install and place it into the basefiles directory. Then name it UBUNTU804\_64.tar.gz for example."

./BASEFILES

**a. apparently, I have to create the "basefiles" directory, as it does not exist by default:**

```
fai:~# mkdir /srv/fai/config/basefiles/
```

### **b. I first have to "build a minimal base image" (source: WIKI), either:**

- using make-fai-base-tgz (but I'm going for the do-it-myself option)
- do it myself (I want to better understand how to build a chroot system)
  - > "You first need to build a minimal Ubuntu system with debootstrap, using an Ubuntu package. Debian's debootstrap does not work for Ubuntu. "  
"Then make an archive with it, say <UBUNTU804\_64.tar.gz>, that goes to <configdir>/basefiles"  
"The easy way is to have an Ubuntu host to run debootstrap on. Otherwise you just get the Ubuntu debootstrap .deb and install it somewhere on a Debian host" (source: IRC, Nic0)
- So my choice is to install a clean Ubuntu 8.04 64-bit server in a virtual machine:

```
root@temp-hardy-debootstrap:~# lsb_release -a
root@temp-hardy-debootstrap:~# lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description: Ubuntu 8.04.3 LTS
Release: 8.04
Codename: hardy
```
- it is the 64-bit version:

```
root@temp-hardy-debootstrap:~# uname -a
Linux hardy64.hostbasket.com 2.6.24-26-server #1 SMP Tue Dec 1 18:26:43 UTC 2009
x86_64 GNU/Linux
```
- First, install "debootstrap" which is a utility that can download and unpack a basic Ubuntu system:

```
root@temp-hardy-debootstrap:~# apt-get update; apt-get install debootstrap
```
- Now create a temporary directory to hold the minimal Ubuntu system:

```
root@temp-hardy-debootstrap:~# mkdir /tmp/ubuntu804_64
```
- Run "debootstrap":

```
root@temp-hardy-debootstrap:~# debootstrap --arch amd64 hardy /tmp/ubuntu804_64/
http://ubuntu.mirrors.skynet.be/ubuntu/
I: Retrieving Release
I: Retrieving Packages
I: Validating Packages
I: Resolving dependencies of required packages...
```

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

```
I: Resolving dependencies of base packages...
I: Checking component main on http://ubuntu.mirrors.skynet.be/ubuntu...
I: Retrieving adduser
I: Validating adduser
...
...
I: Configuring libc6...
I: Configuring initramfs-tools...
I: Base system installed successfully.
```

- You will have something like this:

```
root@temp-hardy-debootstrap:~# tree -dL 1 /tmp/ubuntu804_64/
/tmp/ubuntu804_64/
|-- bin
|-- boot
|-- dev
|-- etc
|-- home
|-- initrd
|-- lib
|-- lib64 -> /lib
|-- media
|-- mnt
|-- opt
|-- proc
|-- root
|-- sbin
|-- srv
|-- sys
|-- tmp
|-- usr
`-- var
```

- now create an archive of it:

```
root@temp-hardy-debootstrap:~# tar -zcf /tmp/UBUNTU804_64.tar.gz -C
/tmp/ubuntu804_64 .
root@temp-hardy-debootstrap:~#
```

- doublecheck if everything went fine:

```
root@temp-hardy-debootstrap:~# tar -tf /tmp/UBUNTU804_64.tar.gz | more
./
./media/
```

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

```
./root/  
./root/.bashrc  
./root/.profile  
...
```

### c. "the archive you build with it must go in FAI config" (source: IRC, Nic0)

- put the archive in the basefiles directory:

```
root@temp-debootstrap:~# scp /tmp/UBUNTU804_64.tar.gz  
root@192.168.118.30:/srv/fai/config/basefiles/  
root@192.168.118.30's password:  
UBUNTU804_64.tar.gz 100% 88MB 14.6MB/s 00:06
```

```
fai:/srv/fai/config/basefiles# ls -lh  
total 88M  
-rw-r--r-- 1 root root 88M 2010-05-05 12:28 UBUNTU804_64.tar.gz
```

```
./CLASS
```

### d. "I would choose a name like <UBUNTU804\_64.tar.gz> This will then be used as a FAI class name" (source: IRC, Mrfai) "add the install host to the class UBUNTU804\_64 (source: WIKI)"

- I make a small adjustment to the default installed file /srv/fai/config/class/50-host-classes. I just assign the 'FAIBASE' and 'UBUNTU804\_64' class to host 'demohost':

```
#!/bin/bash  
  
# assign classes hosts  
  
# use a list of classes for our demo machine  
case $HOSTNAME in  
demohost)  
echo "FAIBASE UBUNTU804_64" ;;  
esac  
  
(ifclass I386 || ifclass AMD64) && echo GRUB
```

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

*exit 0*

- This is how the FAIBASE class looks like by default. I won't change anything here:

```
fai:/srv/fai/config/class# cat FAIBASE.var
# default values for installation. You can override them in your *.var files

# allow installation of packages from unsigned repositories
FAI_ALLOW_UNSIGNED=1

CONSOLEFONT=
KEYMAP=us-latin1

# Set UTC=yes if your system clock is set to UTC (GMT), and UTC=no if not.
UTC=yes
TIMEZONE=Europe/Berlin

# root password for the new installed linux system; md5 and crypt are possible
# pw is "fai"
ROOTPW='$1$kBnWcO.E$djxB128U7dMkrltJHPf6d1'

# MODULESLIST contains modules that will be loaded by the new system,
# not during installation these modules will be written to /etc/modules
# If you need a module during installation, add it to $kernelmodules
# in 20-hwdetect.source. But discover should do most of this job
MODULESLIST="usbkbd ehci-hcd ohci-hcd uhci-hcd usbhid psmouse"

# erros in tasks greater than this value will cause the installation to stop
STOP_ON_ERROR=700

# use the new partitioning tool
USE_SETUP_STORAGE=1
```

- I now create this UBUNTU804\_64 class, where I will simply override the keymap and timezone, just for testing purpose :

```
fai:/srv/fai/config/class# cat UBUNTU804_64.var
KEYMAP=be-latin1
TIMEZONE=Europe/Brussels
```

./DEBCONF

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

e. This subdir is used for the class-independent configuration of the Debconf system parameters. Debian uses Debconf to store global settings such as the default editor or the default webbrowser.

-> not used right now

./DISK\_CONFIG

f. Storage media are configured here

./HOOKS

**g. "Write a hook prepareapt.UBUNTU804\_64 to replace the stuff specific for the distribution of the fai server with something suitable to get your specific distribution ready to install packages. The actual prepareapt task must be skipped. (WIKI)**

Extra:

-> "You'll also need a hook prepareapt.UBUNTU804\_64 to replace the standard task prepareapt. Otherwise the task will copy the Debian sources.list from the nfsroot into your installation" (source: IRC, Nic0)

-> "Copy the function task\_prepareapt from /usr/lib/fai/subroutines." (source: IRC, Nic0)

*And remove at least the 'cp -r \$FAI\_ETC\_DIR/apt/\*' line*

*+ add the following line at the end to skip the task: skiptask prepareapt*

*You can also remove line 18*

*"You also need to add #!/bin/bash and remove the function stuff to make it a script ;-)" (IRC, Nic0)*

- So I first copy the function into a hook, so it looks like this:

*I'm still working on this hook now...*

...

...

## Installing FAI

Written by Jurgen Lamsens

Wednesday, 05 May 2010 10:34 - Last Updated Friday, 01 October 2010 13:41

---

- don't forget to make it executable !

```
fai:/srv/fai/config/hooks# chmod +x prepareapt.UBUNTU804_64
```

...

...

### **./PACKAGE\_CONFIG**

h. This subdir defines the packages required by the target system for each class, usually with the "PACKAGES aptitude package name" command

- I create a simple configuration, where I install the 'SSH server' and the 'tree' package, just for testing purpose:

```
fai:/srv/fai/config/package_config# cat UBUNTU804_64
```

```
PACKAGES aptitude
```

```
openssh-server
```

```
tree
```

i. ...

--- TO BE CONTINUED ---