

Skolelinux - Architecture

Petter Reinholdtsen

`pere@hungry.com`

Skolelinux - Architecture

by Petter Reinholdtsen

Published v0.1, 2002-12-07

Copyright © 2001, 2002, 2003, 2004 Petter Reinholdtsen

* *Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Public License, Version 2 or any later version published by the Free Software Foundation.*

Skolelinux is the Debian-edu project's Custom Debian Distribution (CDD) in development (<http://wiki.debian.net/index.cgi?CustomDebian>). What this means is that Skolelinux is a version of Debian whose out-of-the box environment gives you a completely configured school-network (In Norway, where Skolelinux was started, the main target group is schools serving the 6-16 years age bracket). This document describes the network architecture and services provided by a Skolelinux installation.

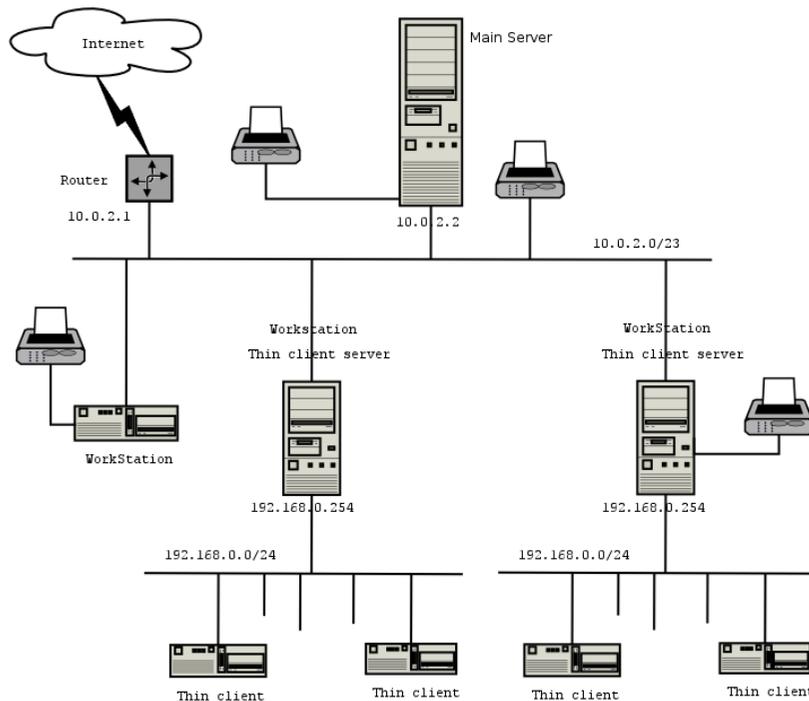
Revision History

Revision 0.1 2002-12-07 Revised by: pere

Table of Contents

- 1. Network 1
- 2. Services 2
 - 2.1. Thin client services 4
- 3. Administration 5
- 4. Installation 6
- A. File system access configuration 7
- B. Keywords 9

Chapter 1. Network



Network architecture

The figure is a sketch of the assumed network topology. The default setup of a Skolelinux assumes that there is one (and only one) main-server, while allowing the inclusion of both normal workstations and thin-client-servers (with associated thin-clients). The number of workstations can be as large or small as you want (starting from none to a lot). The same goes for the thin-client-servers, each of which is on a separate network so that the traffic between the thin-clients and the thin-client-server doesn't affect the rest of the network services.

The reason that there can only be one main-server in each school network is that the main-server provides DHCP, and there can be only one machine doing so in each network. It is possible to move the services of the main-server to other machines by setting up the service on another machine, and subsequently updating the DNS-configuration pointing the DNS alias for that service to the right computer.

In order to simplify the standard setup of Skolelinux the Internet connection runs over a separate router. It is possible to set up Debian with both a modem as an ISDN connection, however no attempt is made to make such a setup work out-of-the box for Skolelinux (the setup needed to adjust default situation to this should be documented separately).

Chapter 2. Services

With the exception of the control of the thin-clients, all services are initially set up on one central computer (the main-server). Because of performance reasons the thin-client-server should be a separate machine (though it is possible to install both the main-server and thin-client-server profiles on the same machine), All services are allocated a dedicated DNS-name and are offered exclusively over IPv4. The allocated DNS name makes it easy to move individual machines from the main-server to a different machine, by simply stopping the service on the main-server, and changing the DNS configuration to point to the new location of the service (which should be setup on that machine first off course).

To ensure security all connections where passwords are transmitted over the network are encrypted, so no passwords are sent over the network as plain text.

Below is a list of the services that are set up by default in a Skolelinux network, with the DNS name of each service given in square brackets. Where possible the DNS name corresponds to the service name in `/etc/services`, where this is not possible the common name of the service is used as the DNS name. All configuration files will, if possible, refer to the service by name (without the domain name) thus making it easy for schools to change either their domain (if they have an own DNS domain), or their ip-address.

- Centralized logging [syslog]
- DNS(Bind?)[domain]
- Automatic network configuration of machines(DHCP)[bootps]
- Clocksynchronization (NTP)[ntp]
- Home directories via network file system (SMB/NFS)[homes]
- Electronic postoffice(Limacute) [postoffice]
- Directoryservice(OpenLDAP)[ldap]
- Webserver (Apache/PHP/eZ)[www]
- SQL server (PostgreSQL)[database]
- Central backup (?) [backup]
- Web-cache / proxy (Squid)[webcache]
- Printing(CUPS) [ipp]
- Remote login (OpenSSH) [ssh]
- Automatic configuring [cfengine]
- Thin client servers (LTSP) [ltsp-server-#]
- Machine- and service surveillance with error reporting, + status and history on the web. Error reporting by mail.

Each user stores his personal files in his home folder which is made available by the server. Home folders are accessible from all machines, giving users access to the same files regardless of which machine they are using. The server is operating system agnostic in offering access using NFS for Unix Clients, SMB for Windows and Macintosh clients.

By default e-mail is set up for local delivery (i.e. within the school) only, though e-mail delivery to the wider Internet may be set up if the school has a fixed Internet-connection. Mailinglists are set up based on the user database, giving each class their own mailinglist. Clients are set up to deliver mail to the server (using “smarthost”), and users can access their personal mail through either POP3 or IMAP.

All services are accesible using the same username and password, thanks to the central user database for authentication and authorization.

To increase performance on frequently accessed sites a web proxy that caches files locally (Squid) is used. In conjunction with blocking web-traffic in the router this also enables control of Internet access on individual machines.

Network configuration on the clients is done automatigically using DHCP. Normal clients are allocated IP addresses in the private subnet 10.0.2.0/23, while thin clients are connected to the corresponding thin-client-server via the seperate subnet 192.168.0.0/24 (this to ensure that the network traffic of the thin clients doesn't interfere with the rest of the network services).

Centralized logging is set up so that all machines send their syslog messages to the server. The syslog service is set up so that it only accepts incoming messages from the local network.

By default the DNS server is set up with a domain for internal use only (*.intern), until a real (“external”) DNS domain can be set up. The DNS server is set up as caching DNS server so that all machines on the network can use it as the main DNS Server.

Pupils and teachers have the possibilty to publish websites. The webserver provides mechanisms for authenticating users, and for limiting access to individual pages and subdiretories to certain users and groups. Users will have the possibility to create dynamic webpages, as the webserver will be programmable on the server side.

Information on users and machines can be changed in one central location, and is made accessible to all computers on the network automatically. To achieve this a centralized directory server is set up. The directory will have information on users, user groups, machines, and groups of machines. To avoid user confusion there won't be any difference between file groups, mailing lists, and network groups. This implies that groups of machines which have to be network groups, have the same namespace as user groups and mailinglists.

Administration of services and users will by and large be via web, and follow established standards, functioning well in the webbrowsers which are part of Skolelinux. The delegation of certain tasks to

individual users or user groups will be made possible by the administration systems.

In order to avoid certain problems with NFS, and to make it simpler to debug problems, the clocks of the different machines need to be synchronized. To achieve this the Skolelinux server is set up as a local Network Time Protocol (NTP) server, and all workstations and clients are set up to synchronize their clock with the server. The server itself should synchronize its clock via NTP against machines on the Internet, thus ensuring the whole network has the correct time.

Printers are connected where convenient, either directly onto the network, or connected to a server, workstation or thin-client-server. Access to printers can be controlled for individual users according to the groups they belong to, this will be achieved by using quota and access control for printers.

2.1. Thin client services

A thin client setup enables an ordinary PC to function as an (X-)terminal. This means that that machine boots from a diskette or directly from the server using network-PROM without using the local client harddrive. The thin client setup used is that of the Linux Terminal Server Project (LTSP).

Thin clients are a good way to make use of older, weaker machines as they effectively run all programs on the LTSP-Server. This works as follows: The service uses DHCP and TFTP to connect to the network and boot from the network. Next, the file system is mounted via NFS from the LTSP-server, and finally X11 is started and connected to the same LTSP-server by XDMCP, thus ensuring that all programs are run on the LTSP-server.

The thin client server is set up to receive syslog from the thin clients, and forward these messages to the central syslog recipient.¹

Notes

1. Oops, the thin clients don't have unique names across LTSP servers. How can we identify which client is logged onto what on the central server?

Chapter 3. Administration

All the linux machines that are installed by means of the Skolelinux CD will be administrable from a central computer, most likely the server. It will be possible to login to all machines by ssh, and thereby have full access to the machines

We use cfengine to edit configuration files. These files are updated from the server to the clients. In order to change the client configuration, it suffices to edit the server configuration and let the automation distribute the changes.

All user information is kept in an SQL database. Updates of user accounts are made against this database. The information is exported to an LDAP directory which is used by the clients for user authentication.

Chapter 4. Installation

Installation is possible either from a CD or by a diskette from server.

The aim is to be able to install a server from CD, and install clients over the network by booting all other machines from the network. The installation has to work without access to the Internet.

The installation should not ask any questions, with the exception of desired language (e.g. Norwegian Bokmal, Nynorsk, Sami) and machine profile (server, workstation, thin client server). All other configuration will be set up automatically with reasonable values, to be changed from a centrally location by the system administrator subsequent to the installation.

Appendix A. File system access configuration

Each Skolelinux user account is assigned a section of the file system on the file server. This section (home directory) contains the user's configuration files, documents, email and web pages. Some of the files should be set to have read access for other users on the system, some should be readable by everyone on the internet, and some should not be accessible for reading by anyone but the user.

To ensure that all disks that are used for user directories or shared directories can be uniquely named across all the computers in the installation, they can be mounted as `/skole/host/directory/`. Initially, one directory is created on the file server, `/skole/tjener/home0/`, in which all the user accounts are created. More directories may then be created when needed, to accomodate particular user groups or particular patterns of usage.

To enable shared file access control using the file groups, each user must be assigned a primary group with no other members. The name of this private group should be identical to the username. ¹ This allows for all new files created by the user to be set with full access for the file's group. Together with set-gid bit on directories and inheritance of rights, this enables controlled file sharing between the members of a file group. Therefore, the users' umask should be 00X. ²

The initial access settings for newly created files is a matter of policy. They may either be set to give read access to everybody, which can later be removed by explicit user action, or they may be initially blocked, necessitating user action to make them accessible. The first approach encourages knowledge sharing, and makes the system more transparent, whereas the second method decreases the risk of unwanted spreading of sensitive information. The problem with the first solution is that it is not apparent to the users that the material they create will be accessible to all other users. This is detectable only upon inspection of other users' directories, where one can see that the files are readable. The problem with the second solution is that few people are likely to make their files accessible, even if they do not contain sensitive information and the content would be helpful to inquisitive users who want to learn how others have solved particular problems (typically configuration issues).

Suggestion: The files are initially set to be readable by all, but particular directories are created in which the content is initially blocked. This will simplify deciding whether the file should be made readable or not. Concretely, umask should be set to 002, and `~/` created with privileges 0775, `~/priv/` given 0750, and `~/pub/` given 0775. Files that should not be readable by others should be put in `~/priv/`, whereas public files will be put in `~/pub/`. Other files will initially be accessible, but may be blocked as needed.

ssh requires that the home directory can only be written to by the owner, thus the maximum access privilege for `~/` is 755.

- access to home directories (*~/.)? - home directories - shared directories?

Notes

1. *More info on private groups* (<http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/ref-guide/s1-users-groups-private-groups.html>) is available from Redhat.
2. If all users initially should be able to read newly created files, then $X=2$. If only the relevant group should be given initial read access then $X=7$.

Appendix B. Keywords

These are random notes concerning things which should be included in this document.

- Centralized user database with grouping and the ability to control which groups have access to which machines.
- Grouping of machines and ability to control access to network services for these groups (access blocking to Internet via squid)
- Should consider using a DNS name from RFC 2606.