



VAASAN AMMATTIKORKEAKOULU  
VASA YRKESHÖGSKOLA  
UNIVERSITY OF APPLIED SCIENCES

Jean Christoffer Forsman

# OPEN SOURCE SOM HELHETSLÖSNING I SKOLMILJÖ

Enheten för företagsekonomi och turism

2010

## FÖRORD

Mitt lärdomsprov handlar om hur man med hjälp av avgiftsfria open source program kan bygga upp en komplett IT-miljö för skolor. I detta lärdomsprov förklarar jag hur jag gått till väga för att introducera programlösningar baserade på öppen källkod i Oravais och Vörå-Maxmo skolor.

Lärdomsprovet har skrivits under tiden oktober 2008 till oktober 2010. Jag har skrivit detta lärdomsprov som en dokumentation av det arbete som gjorts, och jag hoppas att det ska kunna fungera som en guide till hur kommuner ska kunna sänka sina IT utgifter på lång sikt samtidigt som man höjer kvaliteten på programmen och förlänger livslängden på befintliga datasalar. Handledare för detta arbete har varit Kenneth Norrgård.

Vasa den 1 november 2010

---

Jean Forsman

## VASA YRKESHÖGSKOLA

## Utbildningsprogrammet för informationsbehandling

## ABSTRAKT

Författare	Jean Forsman
Lärdomsprovets titel	Open Source som helhetslösning i skolmiljö
År	2010
Språk	Svenska
Sidantal	84 + 4 bilagor
Handledare	Kenneth Norrgård

---

Ändamålet med detta arbete har varit att utreda vad som krävs för att skapa en fungerande IT miljö med hjälp av program baserade på öppen källkod.

I arbetet beskrivs samtliga program och deras tillämpningsområden grundligt.

Det fysiska arbetet har delats upp i tre faser. Den första fasen går ut på att kartlägga behov och avgöra hur lämpliga Linux-operativsystemet är för skolelever. Den andra fasen är ett exempel på hur man kan konvertera en datasal till Linux med syfte att öka effektiviteten. Den tredje fasen den viktigaste punkten i lärdomsprovet. Den visar hur program med öppen källkod kan användas för att bygga en fungerande helhetslösning för skolektorn. Alla program som använts har varit avgiftsfria och registrerade under någon form av GPL licens.

VASA UNIVERSITY OF APPLIED SCIENCES  
Utbildningsprogrammet för informationsbehandling

ABSTRACT

Author	Jean Forsman
Topic	Open Source as a complete solution in school environments
Year	2010
Language	Swedish
Pages	84 + 4 appendices
Name of Supervisor	Kenneth Norrgård

---

The purpose of this thesis has been to determine what is required to create a fully functional IT environment using only open source software.

All the programs and their applications are thoroughly described.

The physical work has been divided into three phases. The first phase is to identify needs and determine how suitable the Linux operating system is for school children. The second phase is an example of how to convert a computer lab to Linux with the aim of increasing efficiency. The third phase is the most important part of this thesis. It shows how open source software can be used to build a fully functional working environment for the school sector. All software used has been provided free of charge and are registered under GPL license.

---

Keywords: Open source, IT-architecture, User interface, Licenses

## INNEHÅLL

## ABSTRAKT

## ABSTRACT

## IT-ORDLISTA

1.	INLEDNING	13
1.1.	Syfte	13
1.2.	Genomförande	15
2.	ÖPPEN KÄLLKOD SOM KONCEPT	17
2.1.	Open Source	17
2.2.	Green IT Konceptet	21
2.3.	Licenser	24
2.4.	Moodle	29
2.5.	Zimbra	31
2.6.	Föräldramötet	34
2.7.	Openmeetings	36
2.8.	LDAP, lightweight directory access protocol	37
2.9.	Definition av Linuxdistribution	39
2.10.	Distributioner som använts i projektet	40
3.	FAS 1: TESTNING OCH UTVÄRDERING	50
3.1.	Acer minilaptops	50
3.2.	Skol-Linux och Mini-laptops i Nykarleby	51
4.	FAS 2: KONVERTERING	53
4.1.	Konvertering av datasal i Oravais	53
4.2.	Datorer och problemställningar	53
4.3.	Installationer i Oravais	54
4.3.1.	Printning	55

	6
4.3.2. Fil- och Serverhantering	55
4.3.3. Problemlösning för kommunikation med Windows servern	56
4.3.4. Distributionsinnehåll	58
4.3.5. Skolning	58
4.4. Feedback samt fler datorer	59
5. FAS 3: IMPLEMENTERING OCH ÖVERGÅNG	61
5.1. Vörå-Maxmo och Eduvom projektet	61
5.2. Serverinstallation	62
5.2.1. Vad skiljer en virtuell server från en vanlig?	63
5.2.2. Virtuell server	64
5.2.3. LDAP i Vörå	68
5.2.4. Datorer utan LDAP	68
5.3. Thin & Fat Client lösning	69
5.3.1. LTSP workshop i Vörå	70
5.4. Brandvägg i Vörå	72
5.5. Moodle och Föräldramötet	73
5.6. Openmeetings	74
6. AVSLUTNING	80
KÄLLFÖRTECKNING	81
BILAGOR	85

## IT-ORDLISTA

Den här IT-ordlistan har sammanställts utgående från artiklar på Wikipedia. Dess syfte är fungera som stöd för läsaren, eftersom min text innehåller en del terminologi och uttryck som kanske inte är bekanta för personer som inte sysslar med IT.

### Bootning & Bootsekvens

Bootning eller Initial Program Load är en term för processen som startar igång ett operativsystem på en dator när strömmen slås på. Med bootsekvens avses de operationer som datorn genomför efter strömpåslag som leder till att datorn är driftklar.

### BIOS

Basic Input/Output System, är det mest grundläggande programmet som körs i en PC-dator när den startas. Dess främsta syfte är att starta datorns periferenheter, som skärm, tangentbord och det inbyggda skivminnet (hårddisken).

### CUPS

Common UNIX Printing System, ett portabelt ramverk för utskrifter och printerhantering för Unix och Unixliknande operativsystem.

CUPS använder Internet Printing Protocol för att hantera skrivarköer och jobb. Andra tekniker för skrivarhantering som Line Printer Daemon, Server Message Block och JetDirect stöds också i varierande grad.

CUPS är fri programvara och är licensierad under GNU GPL.

### Debian

Debian GNU/Linux är en av de största och äldsta linuxdistributionerna, skapad 1993. Debian betonar frihet och kvalitet, det senare med följden att programvaran i de officiella versionerna är äldre än i många andra distributioner. Debian stöder ett dussintal olika datorarkitekturer och förutom Linux också vissa andra operativsystemkärnor.

### Ekiga

Ekiga (f.d. GNOME Meeting) är ett program avsett för IP-telefoni och videokonferenser.

## Linux

Linux är egentligen kärnan i fria operativsystem såsom Debian och Fedora, men dessa operativsystem kallas ofta bara Linux.

## E-post, E-mail

En av de ursprungliga typerna av meddelandebefordran över Internet. Det kännetecknas av högre tillförlitlighet än andra typer av elektroniska meddelanden. E-brev läses med ett e-postprogram eller webbposttjänster som exempelvis Yahoo, Hotmail och Gmail.

E-post levereras ofta genom servrar med hjälp av nätverksprotokollet SMTP. Vanligtvis lagras brevet på en server som låter mottagaren komma åt brevet med hjälp av protokollen IMAP, POP3 eller med hjälp av ett webbgränssnitt. Inom stora organisationer är det vanligt med e-postsystem som använder slutna protokoll.

## GNU

GNU-projektet grundades 1984 med målet att utveckla ett komplett Unix-liknande operativsystem bestående helt av fri programvara: operativsystemet GNU. GNU är en rekursiv akronym av engelskans GNU's Not Unix, på svenska ungefär GNU är inte Unix.

## HTML

Är en förkortning för Hyper Text Markup Language och är ett märkspråk och webbstandard för strukturering av text, hypertext, media och inbyggda objekt på exempelvis webbsidor och i e-postmeddelanden.

## Hårddisk

En anordning för lagring av information som används i datorer

## Mac

En persondator från det amerikanska datorföretaget Apple Inc. Namnet har hämtats från den kanadensiska äppelsorten McIntosh. Jef Raskin, en av



nyckelpersonerna i det ursprungliga Macintosh-projektet.

#### Microsoft Office

Microsoft Office är ett kontorspaket, det vill säga en samling programvaruprodukter för diverse kontorsarbeten. Microsoft Office finns i olika programversioner med olika innehåll.

#### Open Office

OpenOffice.org är en samling kontorsprogram som härstammar från programvaruföretaget Sun Microsystems kontorsprogram StarOffice, ursprungligen utvecklat av Star Division. Det består av de delar av StarOffice vars källkod Sun har släppt fri.

#### Open Kvarken projektet

Open Kvarken är ett Open source projekt som arbetar för att utvidga användningen av öppen källkod i Kvarken-regionen. Projektet kommer att etablera permanenta kompetenscenter i Vasa och Umeå.

#### Open Source, öppen källkod

Datorprogram där källkoden är tillgänglig att använda, läsa, modifiera och vidare distribuera för den som vill. Detta gör att användaren kan försäkra sig om att programmet gör vad det ska, eller anpassa det till sina behov. Sådana modifikationer erbjuds vanligen tillbaka till den ursprungliga upphovsmannen, som kan välja att göra dem till en del av den officiella versionen.

#### PDF

PDF, Portable Document Format, är ett digitalt dokumentformat utvecklat av Adobe Systems och introducerat 1993. Filerna visas på skärm i samma form som de har som utskrivna, så långt skärmens upplösning tillåter.

#### Powerpoint

Microsoft Powerpoint är ett presentationsprogram från Microsoft och ingår i programsviten Microsoft Office. Programmet är mycket vanligt och närmast dominerande för presentationer på företag. Motsvarigheten till

Powerpoint i OpenOffice heter Impress. Man talar dock ofta generellt om Powerpoint-presentationer, oavsett vilket filformat eller i vilket program presentationen gjorts.

### Proprietär programvara

Proprietär programvara är programvara som har restriktioner (vanligtvis satta av ägaren) vad gäller att använda, modifiera eller kopiera den och alltså saknar de grundläggande friheter som finns hos fri programvara; exempelvis får man inte tillgång till källkoden, man får inte ändra i programmet, och man får inte ge bort kopior. Stängd källkod används synonymt med proprietär programvara, för program där källkoden inte är offentlig.

### RAM-minne

Random Access Memory eller RAM är ett minne där man kan nå varje minnescell direkt och utan att behöva läsa igenom andra delar av minnet till skillnad från minnen som läses sekventiellt. RAM används ofta som benämning på olika typer av arbetsminne av RWM-typ.

### Server

En värddator (engelska: server) är i datasammanhang ett system som betjänar andra system, klienter, ofta över ett nätverk. Beroende på sammanhang kan ordet syfta på den fysiska datorn eller programvaran den kör. Vanliga tjänster som använder värddatorer utnyttjas när man använder Internet, som E-post, World Wide Web och DNS. Exempelvis webbservrar är ofta i sin tur klienter till en databas-server.

### Skrivbord

Skrivbord är en vardagsterm för det man egentligen kallar grafiskt användargränssnitt. Ett grafiskt användargränssnitt (engelska: Graphical user interface, GUI) är en metod för att underlätta interaktion mellan människa och dator. Detta typiskt genom bilder och grafiska element som utgör metaforer eller analogier till objekt i verkligheten.

### Thin/Fat – client

Thin client eller Tunna klienter innebär att en datoranvändare arbetar direkt mot ett centralt nätverk. Lokalt har den enskilda användaren tangentbord, mus och bildskärm kopplat till en minimal centralenhet, utan hårddisk och operativsystem. Centralenheten kräver inte lika snabb CPU och inte lika mycket arbetsminne som en vanlig persondator som kör tillämpningsprogramen lokalt. Prestandan beror istället på nätverkets och serverns kapacitet eftersom det är där som det egentliga operativsystemet och alla programmen körs.

En fet klient eller rik klient är en dator i ett klient-server nätverk som normalt ger rik funktionalitet oberoende av den centrala servern. Namnet syftar till att det är en kontrast från tunna klienter, eftersom en fet klient använder sin egen hårdvara för att köra program. En fet klient kräver dock fortfarande minst periodiska anslutning till ett nätverk eller en central server.

#### Webbkonferensverktyg

Webbkonferens möjliggör möten på distans via Internet. Webbkonferenser fungerar så att flera kan prata samtidigt med varandra (via datorn, telefon eller en kombination). Deltagarna ser alla samma sak på sin dataskärm, varandras ansikte via webbkameror, en presentation, ett program eller en Internetsida.

#### Windows 95, NT, XP, Vista, 7

Microsoft Windows är ett operativsystem med mycket bred spridning på persondatorer. Det är skapat av Microsoft och är företagets flaggskepp, tillsammans med Office-produkterna. Namnet Windows (fönster) syftar just på systemets grafiska användargränssnitt med fönsterhantering.

#### WYSIWYG

What You See Is What You Get, engelska för "vad du ser (på bildskärmen) är vad du får (utskrivet)", var en slogan för ordbehandlare och layoutprogram som syftade på att programmet på skärmen hela tiden avbildar textens utseende "exakt" som resultatet blir på papper.

WYSIWYG-uttrycket används nuförtiden även i en rad andra sammanhang, i synnerhet om program för att skapa webbsidor, då till skillnad från program som i första hand visar själva HTML-koden och eventuellt en enklare förhandsvisning.

## 1. INLEDNING

Detta projekt har varit väldigt omfattande, och det skulle ha varit väldigt svårt att genomföra utan stöd från mina kollegor Mikael Torp och Daniel Lindblad. Jag vill tacka Teemu Haapoja och Johnny Ström för den hjälp jag fått med bland annat serverkonfigurationer. Jag vill även tacka Rainer Lytz och OpenKvarken projektet för möjligheten att förverkliga detta projekt. Om det inte varit för OpenKvarken projektet så skulle detta lärdomsprov inte ha kunnat förverkligas. Glädjande är även att det ute i kommunerna finns ett intresse för övergång till öppna programlösningar och öppna standarder, och jag är tacksam för det varma mottagande jag fått från Vörå kommun och IT-ansvarige Kenneth Nickull.

Jag vill även tacka Angus MacGyver för att han alltid funnits till hands under min uppväxt och för att han visat att allt är möjligt om viljan och kunnandet finns. Slutligen vill jag tacka Juliet Simms för att hon funnits vid min sida under många långa sömnlösa nätter.

### 1.1. Syfte

Idén till detta projekt väcktes egentligen för ca ett år sedan när jag utförde min arbetspraktik vid Vasa yrkeshögskola sommaren 2008. Jag såg alla datorer som stod i korridorerna, några av dem hade redan fått ett nytt hem hos någon studerande, men för många andra såg återvinningsstationen ut att bli deras slutdestination.

Detta var på sätt och vis förståeligt, många av datorerna kunde ha kallats skräp. Det hela kändes ändå som ett onödigt slöseri, det måste ju finnas någon som kan ha användning av dessa gamla arbetshästar. Det var ungefär två månader in i projektets gång som min chef Rainer Lytz berättade att vi skulle besöka Vörå och berätta om Openkvarken projektet.

Det var i det här skedet som jag kom att tänka på hur dåliga datorer vi hade i mina grundskolor, och anslagen för kommunala skolor har knappast ökat de senaste 15 åren. Jag vet även vilka licenskostnader som uppbärs för bland annat Microsoft

licenser, vilket i sig är en förklaring till varför det knappt görs några nya datorinköp till skolorna, trots att det varje år budgeteras pengar för ändamålet. Jag började fundera kring vilka behov man hade på grundskolenivå, och kom ganska snabbt fram till att alla program som användes fanns att få som licensfria. Varför betala för något man kan få gratis?

Jag bollade idén med mina kollegor, och de var försiktigt positiva till det hela. Det fanns en viss oro gällande underhåll och det faktum att det rörde sig om gamla datorer. Vem skulle reparera dem ifall något går sönder? Mitt projekt fick slutligen godkännande med ett axelryckande ”det är väl värt ett försök”.

Det var viktigt att i ett tidigt skede komma överens om att Vasa yrkeshögskola och Openkvarken avsäger sig allt ägande och ansvar för datorerna i samma stund som de levererats.

Grundtanken var alltså att samla in avlagda datorer från Vasa yrkeshögskola och installera Linux operativsystem samt licensfria programvaror och med hjälp av dessa datorer bygga upp EN datasal med ett tjugotal datorer i Vörå. Något av en testmiljö för att se hur komplicerat det är att bygga upp en datasal med Linux programvara.

Under vårt möte i Vörå togs idén positivt emot, men jag var osäker huruvida jag verkligen lyckats sälja min vision. Jag måste medge att jag blev smått förvånad när Vörå-Maxmos IT-ansvarige Kenneth Nickull ett par veckor senare meddelade att de ville ha 80 datorer. Jag var givetvis glad över att de var så positivt inställda till min idé, men var skulle jag få tag i 80 datorer? Jag hade ju hoppats på 20.

Lyckligtvis hade min kollega Teemu Haapoja goda kontakter till skolan IT-avdelning, och det var tack vare honom som alla datorer fann sin väg till vårt arbetsrum. Allt efter som projektet fortgick uppstod nya idéer om vad man skulle kunna göra för att utveckla datasalarna och spara pengar.

Min tanke var att erbjuda en helhetslösning, med Moodle som inlärningsplattform, Zimbra som e-mail klient, Openmeetings som ett webbkonferensverktyg och en datasal med licensfria program. Detta skulle i slutändan resultera i en testmiljö

och om framgångsrikt, ett exempel på hur man kan bygga en skolmiljö med endast open source program.

## **1.2. Genomförande**

Jag valde att dela upp själva arbetet i tre stycken faser. Jag gjorde detta för att jag hoppades på att det skulle avslöja några av problemen i ett tidigt skede, och på så sätt ge mig mer tid att angripa problemet. Det första kapitlet behandlar teori och fakta kring de olika programvarorna som använts, och ska förhoppningsvis ge en bild av hur de ser ut och fungerar.

### Fas 1. Test och utvärdering

Målet med denna fas var att fastställa hur mottagliga elever och lärare var för ett annat operativsystem än t.ex. Windows XP.

Från egen erfarenhet visste jag att Linux-skrivbordsmiljön har utvecklats mycket de senaste åren, och jag visste att det är en fullgod arbetsmiljö. Jag kunde dock inte använda mitt eget omdöme i denna fråga eftersom jag själv arbetat med och testat olika program på daglig basis de senaste 15 åren, och jag insåg att mina krav för vad som är användarvänligt kunde skilja sig markant från vardagsanvändaren.

Det var även ett intressant test eftersom det rörde sig om elever i årskurs 4. Min teori var att eftersom de var så unga så hade de inga inrotade vanor om hur ett operativsystem ska se ut och vad alla knappar ska göra, och elever skulle därför ha lättare att anpassa sig till den nya användarmiljön.

### Fas 2. Konvertering

Jag fick ett e-mail från Centrumskolan i Oravais. De förklarade att de hade en föråldrad datasal med datorer som låste sig när de körde Windows NT och Windows 95, och ville därför försöka köra Linux för att se om det skulle fungera bättre.

De hade även nyare datorer som de ville behålla Windows på, samtidigt som de

hade en ny server som körde Windows server 2007.

Detta gav möjlighet till en intressant försöksmiljö där man dels körde Linux datorer, dels Windows datorer och sparade allt på en Windows server.

Jag ville även pröva på den form av konvertering eftersom jag misstänker att den kommer att bli allt mer vanlig i våra skolor, dvs. att man behåller server-systemet men låter gamla datorer övergå till Linux istället för att slänga bort dem.

### Fas 3. Implementering och övergång

Detta avsnitt behandlar endast det arbete som gjorts i Vörå-Maxmo.

Visionen var att bygga upp en hel arbetsmiljö för Vörå-Maxmos skolor och endast använda avgiftsfria program med öppen källkod.

Vi gjorde en mängd olika implementeringar, bland annat förnyad brandvägg och två läroplattformar till lågstadie, högstadie och gymnasie.

Mitt drömscenario skulle ha varit att man gjort sig av med alla licenser och helt och hållet övergått till t.ex. Linux, men till följd av ett tidigare licensavtal kunde man inte överge Windows helt och hållet förrän år 2011, när licensavtalet går ut.

Jag levererade två olika lösningar för att introducera Linux i skolorna. Jag har valt att kalla dem för Green IT-konceptet och Fat client-konceptet. Det första baserar sig på att återanvända avlagda datorer genom att installera Linux på dem, det andra är en migrationsstrategi för att vänja användare vid Linux samtidigt som man behåller det nuvarande operativsystemet (Windows)



## 2. ÖPPEN KÄLLKOD SOM KONCEPT

### 2.1. Open Source

Begreppet Open Source översätts lättast till öppen källkod. Konceptet är användaren av ett program har rätt att se koden i programmet, göra förändringar i koden samt kunna vidare distribuera programmet som helhet. Dessa program är skyddade under GPL eller GNU-licenser.

Begreppet Open Source är ännu inte allmänt känt, och Open Source produkter misstas ofta för så kallade gratisprogram. Dessa gratisprogram kallas proprietär programvara, och trots att de kanske finns tillgängliga för gratis nedladdning och användning så har användaren inga rättigheter att se koden och göra förändringar. Man har med andra ord inte möjlighet att se om programmet i fråga döljer någon form av spionprogram, eller om det gör andra saker än vad som uppges. Ett bra exempel på detta är Skype. Många företag skulle kunna spara enorma summor genom att övergå till IP-telefoni via t.ex. Skype, men eftersom man inte vet om samtalen via Skype avlyssnas eller sparas så vågar man inte använda det. Nu hävdar jag ingalunda att Skype spionerar på sina användare, men när det handlar om företagshemligheter så är en liten risk alltid för stor. Med anledning av detta så har vi inom Openkvarken projektet arbetat för att vidareutveckla en motsvarighet till Skype, nämligen Ekiga. I skrivande stund finns det problem med att få Ekiga att gå genom brandväggar och därför har utvecklingen hämmats. Genom Openkvarken och OCOSS-2 projektet har vi lyckats köpa upp koden till en tunnelleringsfunktion från Sesca i Vasa. Denna tunnelleringsfunktion kommer att göra så att all trafik från Ekiga går via port 80, d.v.s. den port som används för att surfa på nätet. Denna port är alltid öppen, och kommer att möjliggöra användning av Ekiga från alla datorer som har en internetanslutning.

Upphovsmännen bakom namnet Open Source är Eric Raymond och Bruce Perens som senare grundade Open Source Initiative. De ansåg att begreppet ”fri programvara” alldeles för lätt kopplades ihop med FSF (Free software foundation). Definitionen av öppen källkod konstruerades senare efter ”Debian

Free Software Guidelines”. Dessa är en uppsättning riktlinjer som Debian-projektet använder för att bestämma om en programvara kan anses lyda under en fri programvarulicens, vilket i sin tur är avgörande för om en programvara kan inkluderas i Debian. Den första, och kanske viktigaste punkten är fri återdistribution, även om den används kommersiellt och tillsammans med annan programvara. Ingen royalty eller avgift får krävas.

Man kan jämföra öppen källkod med att man anlitar en byggfirma för att bygga ett hus, och när huset är färdigt får kunden en kopia av ritningen så att han själv kan förändra huset. Alternativet är att arkitekten eller byggfirman behåller alla ritningar, och alla ändringar måste således utföras av samma byggföretag.

Ett program som är tillgängligt under Open Source kan förändras och skräddarsys enligt kundens behov och önskemål. Man kan välja vilka funktioner man önskar och programvaran kan byggas på i ett senare skede om behov uppstår genom t.ex. omstruktureringar i organisationen.

Tack vare att källkoden finns tillgänglig är man inte bunden till en specifik leverantör, och den som gör förändringarna i programmet behöver nödvändigtvis inte vara samma person eller företag som gjorde den ursprungliga programlösningen. Man kan således lätt byta leverantör utan att äventyra den infrastruktur man redan byggt upp.

Det här är kanske den punkt på vilken Open Source skiljer sig mest från proprietära programvarulösningar. När man väljer en proprietär programlösning är man ofta bunden till en eller flera leverantörer, och samkörningar mellan olika system är problematiska och i de många fall omöjliga. Genom att välja program med öppen källkod kan man alltid skräddarsy sin arbetsmiljö efter behov.

Spridning är nyckeln. Ju fler användare ett program har, desto snabbare hittar man eventuella buggar och problem. Genom att ge tillgång till källkoden öppnar man även upp möjligheterna till att en community bildas.

Någon bra översättning på ordet ”community” till svenska är svår att hitta. Alltför

ofta associeras "community" idag fortfarande med tekniska funktioner som chatt och forum. Även om detta kan utgöra grunden för en community på Internet, är en community mycket mer än så.

En community omfattar dess medlemmar i form av människor, en gemensam identitet och intresse. Fokus ligger på gruppen och interaktionen, i detta fall ett utbyte av information och idéer kring programvaran. Om man använder en öppen programvara och önskar en speciell funktion i programmet så är communityn en bra plats att vända sig. Oftast finns det en prioritetslista för olika funktioner, och beroende på hur användbar funktionen är för den breda skaran kan man få vänta länge innan man hittar någon som är beredd att utveckla funktionen. Detta kan dock oftast kringås på ett flertal sätt. Ett sätt är att donera pengar till communityn. För en symbolisk summa i stil med 200€ kan ens förslag få ökad prioritet, och klättra upp på prioritetslistan. Ett annat sätt är att anställa en programmerare från communityn. Det senare förslaget är mycket dyrare och är ämnat för större programförändringar. Priset kan ändå vara upp till 10 gånger billigare än vad det kostar att anlita ett programmeringsföretag.

Man brukar säga att bra källkod aldrig dör. Ett bra program får lätt efterföljare som i sin tur hjälper till att utveckla programvaran. Alla inom communityn är nödvändigtvis inte programmerare, men kan oftast bidra genom att till exempel påpeka brister i ett program. Det är en gemenskap med ett enda mål, att skapa så bra programvara som bara är möjligt.

Alla kan använda öppen källkod och framför allt, Linux. För den erfarna användaren kan förändringen ske över en natt. Det hela kan jämföras med en upptäcktsfärd eller kanske en resa in i det okända. Den nya generationens Linux-distributioner har fått ett grafiskt gränssnitt som liknar Windows till den grad att de passar alla användare, och en användarvänlighet som gör att t.o.m. barn kan använda programvaran.

En övergång till t.ex. Linux-distributionen Ubuntu öppnar upp dörren till tusentals kostnadsfria program, alla till användarens förfogande, att användas eller förkastas. Att hela tiden testa nya program som blir tillgängliga kan dock snabbt

bli ett heltidsarbete, och därför har t.ex. Ubuntu-distributionen en komplett uppsättning program färdigt installerade.

Även distributionerna i sig kan bli ett dilemma. Under juli 2009 släpptes inte mindre än 57 nya Linux distributioner. Eftersom det är snudd på omöjligt att hinna pröva alla innan nya versioner släpps är det en bra tumregel att testa sig fram tills man hittar något som motsvarar ens krav och önskemål, och sedan hålla fast vid det. Självt började jag med Ubuntu, men har under det senaste året övergått till Linux 7 Mint KDE.

Program med öppen källkod utvecklas generellt sett mycket snabbare än program som underhålls av ett specifikt företag, och ett program som man i ett skede dömde ut som undermåligt kan 6 månader senare ha utvecklats till den nivå att den kan ersätta proprietär programvara. Källkoden i denna nya och förbättrade versionen kan i sin tur komma andra program till gagn.

Det bästa exemplet för när öppen källkod kommer bäst till användning är enligt mig den offentliga sektorn, eftersom som alla kommuner och städer har samma behov när det gäller IT-administration. Om man valde lösningar med öppen källkod så skulle man kunna dela med sig av programvaran och spara ofantliga summor pengar. Tyvärr väljer dock alltför många kommuner och städer att inte se längre än den egna organisationen, och betalar fantasisummor till utländska företag för att de ska uppfinna hjulet gång på gång. Vad de i slutändan får är en dyr proprietär paketlösning som inte motsvarar deras behov fullt ut, och som de inte har möjlighet att förändra eftersom källkoden är låst.

Som tur är finns det ljus i mörkret. Uleåborg stad har valt att ge ut sitt ärendehanteringsprogram Omaoulu.fi som open source, och källkoden med tillhörande dokumentation finns att ladda ned från OSOR.EU. Om man behöver support kan man kontakta företag Ixonos som hjälpt till att utveckla programvaran. Vad detta i slutändan innebär är att man istället för att köpa ett färdigt paket med proprietär kod för 250 000 euro kan använda ett befintligt open source program och betala ett företag skräddarsy det enligt ens önskemål. Kostnaden för att skräddarsy ett open source program landar oftast på runt 40 000 euro. Allt

vad som krävs är att man låter andra se de förändringar man själv gjort i programmet så att de i sin tur slipper uppfinna hjulet igen.

## 2.2. Green IT Konceptet

Inom högskolevärlden ersätts varje år en uppsjö av datorer, trots att de är fullt funktionerande. Deras enda brist är att de inte kan köra de senaste Windows programmen, och till följd av detta motsvarar de inte längre de kriterier som satts upp för undervisningsdatorer.

Detta är ju fullt förståeligt. På högskolenivå så måste man ha tillgång till alla de program och system som ingår i utbildningsprogrammet, och ingen vill ju sitta i datasalen som tar en halv föreläsning att starta upp.

Skolan har en egen intern minnesåtervinning för själva hårdvaran, där avskrivna datorer används som reservdelar. Men med dagens teknologiutveckling är ett 3 år gammalt RAM-minne ingalunda kompatibelt med en ny dator, och är därför tämligen värdelös som reservdel.

Det var då jag fick tanken till vad jag valt att kalla Green IT-konceptet.

Tänk om alla datorer från alla högskolor kunde återanvändas i de närliggande kommunerna. De utbildningar med datorer som grundskolorna ger innefattar oftast bara grundläggarna kunskaper inom ord och bildhantering, på högre stadier kanske grundläggande HTML.

Program som till exempel OpenOffice och Fronter har eliminerat behovet av att skaffa dyra proprietära program såsom Microsoft Office, men tänk om man kunde ta det hela ett steg längre?

Tänk om man kunde bygga upp en hel arbetsmiljö med 100% kostnadsfria program, kanske till och med endast använda program med öppen källkod? Office 2007 har visserligen många fina animerade funktioner och snygga rundade ikoner, men är det verkligen nödvändigt att spendera så stora summor pengar för att få den där irriterande lilla animationen i högra hörnet som ger dumma förslag till vad

man gör fel? I mina ögon vore det bättre att spendera licenspengarna på att till exempel anställa en lärare eller behålla ett skolkök.

Att hitta gratis programvara som är identisk med de dyra mjukvarujättarnas program är i många fall inte möjligt. Open Source utvecklas med stormsteg och programmen utvecklas ständigt mot det bättre, men ännu är vi inte där. Mitt mål var dock aldrig att replikera den senaste Microsoft-utgåvan. Vad jag eftersträvade var att bygga upp en arbetsmiljö som fyllde alla de kriterier som satts i läroplanen för grundskolornas utbildningar, fri från licenser och som skulle fungera på datorer som dömts ut som otjänliga i kombination med Windows programvara.

Teemu Haapoja ansvarade för att samla ihop avlagda datorer från skolans olika avdelningar. Datorerna rengjordes och kontrollerades. En del av datorerna saknade t.ex. RAM-minne, andra hade en söndrig hårddisk. Uppskattningsvis 95% av datorerna kunde sättas i funktionsdugligt skick. Denna siffra motsvarade ungefär mina förväntningar eftersom det ändå rörde sig om ca 5 år gamla datorer. Överlag var datorerna i bättre skick än jag trodde att de skulle vara, beaktande att datorerna varit i bruk i medeltal 12 timmar per dag.

Det första hindret var att biosen på datorerna var lösenordsskyddad och lösenordet fanns inte att finna. Den verkliga sanningen var väl att samma lösenord användes på de kvarvarande datorerna i skolan, och IT-supporten såg mig förmodligen som en säkerhetsrisk. Jag har ingen kommentar till huruvida det var befodrat. Vad detta ledde till var att vi blev tvugna att nollställa alla biosar för att kunna ändra bootsekvens. Detta gör man väldigt enkelt genom att lösgöra ett litet batteri från datorns moderkort, och sedan sätta tillbaka det. Detta batteri har som uppgift att hålla alla BIOS-inställningar vid liv även när datorn inte strömförsedd.

Efter att detta gjorts bootades alla datorer upp med Clonezilla.

Clonezilla är ett program med öppen källkod som gör det möjligt att spara ned

innehållet på en dator till en portabel hårddisk eller ett USB-minne. Detta ska dock inte förväxlas med en säkerhetskopia. En säkerhetskopia sparar innehållet på datorn, men inga uppgifter om själva datorn. Med en kloning så fångar man datorn som den var i det ögonblicket, operativsystem och samtliga uppgifter om den fysiska datorns egenskaper. Detta är ett mycket användbart verktyg för systemadministratörer eftersom de kan göra en basinstallation på en viss datormodell, spara en klon av datorn, och ifall en dator av den modellen kraschar kan man med hjälp av Clonezilla återställa datorn inom några minuter.

Ett problem som uppstod var att en del av datorerna inte ville boota upp via Clonezilla-skivan. Detta var förmodligen en kombination av gamla slitna dvd-stationer och det faktum att datorerna har Windows installerat på hårddisken. Efter att datorerna klonats till Linux-datorer blev de mycket mer benägna att boota upp med Clonezilla. Detta åtgärdades genom att hårddiskarna på de datorer som inte ville boota upp lösgjordes och sattes in i en dator som var mera villig att boota upp via clonezilla. Detta medförde givetvis lite extra arbete, men det var ändå den snabbaste lösningen på problemet.

Eftersom alla datorer var av samma modell och med samma hårddisk, fanns möjligheten att göra en kopia av en befintlig Linux-installation, en så kallad klon.

Jag började med att installera Edubuntu på en av datorerna, skräddarsydde innehållet och gjorde samtliga konfigurationer som behövdes för att logga in mot LDAP servern. Efteråt gjorde jag en klon med hjälp av programmet Clonezilla [bilaga Clonezilla], och sparade ned alltihop till en portabel hårddisk.

Fördelen med detta är att man kan återställa en klon med Clonezilla på runt 5 minuter ifall att olyckan är framme. Att göra en installation från grunden, göra alla behövliga förändringar samt konfiguration skulle ha tagit minst 12 timmar per dator. Jag beskriver kloningsprocessen närmare i bilagan gällande Clonezilla.

## 2.3. Licenser

Det finns ett antal olika licenser som behandlar rättigheterna för öppen källkod.

GNU General Public License, förkortat GNU GPL eller endast GPL.

GPL licensen skrevs av Richard Stallman för GNU projektet. Stallmans mål var att producera en licens som kan användas för alla projekt, vilket gör det möjligt för många projekt att dela kod.

GPL är det populäraste och mest kända exemplet på den typ av starka copyleft-licens som kräver härledda verk kommer att finnas tillgängliga på samma sätt som det ursprungliga.

Enligt [www.blackducksoftware.com](http://www.blackducksoftware.com) använde i September 2009 hela 64,77% av alla open source projekt någon form av GNU GPL licens. De efterföljande var Artistic License (Perl) med 8.79% samt BSD License 2.0 på 6.30%. Detta innebär att GPL är den överlägset mest använda.

Filosofin bakom GPL är "Free as in freedom". Det brukar även klarifieras genom uttrycket "Free as in freedom, not price". Syftet är alltså att ge mottagaren av ett program möjlighet att göra förändringar och vidareutveckla programmet, inte att skapa en kommunistisk eutopi där allt är gratis. Den starka copyleft-licensen garanterar att programvaran förblir fri även efter att förändringar gjorts i programmet. Det är till exempel inte lagligt för ett företag att använda öppen källkod licensierad under GPL, sälja programmet som sitt eget och sedan vägra ge ut programkoden. För att förhindra att detta sker har man tagit fram ett program som heter Black duck. Black duck har tillgång till all kod som är licensierad som öppen källkod, och genom att köra ett program genom Black duck kan man se om programmet i fråga innehåller öppen källkod. Detta är ett jättebra verktyg för att se om programkoden man köper verkligen är unik, eller om det bara handlar om öppen källkod som någon försöker sälja som sin egen. Att köra ett program genom Black duck betyder inte att man måste avslöja sin egen kod som finns i programmet. Det är därför ett mysterium för mig varför många stora proprietära programvarutillverkare vägrar låta sina program köras genom Black duck.



GNU GPL Version 1 släpptes januari 1989.

Den första versionen som kallades GPLv1 skapades för att förebygga de dåvarande restriktionerna som satts av mjukvarudistributörer. Det första problemet var att distributörerna endast offentliggjorde binära filer. En binär fil är vad du vanligtvis laddar ned när du laddar ned ett program via internet. Dessa filer är körbara, men kan inte läsas eller modifieras av användaren.

Det andra problemet var att distributörer kombinerade en programvara med en annan, vilket ledde till att licensrestriktioner för båda programvarorna kombinerades, vilket i sin tur ledde till en oacceptabel kombination av restriktioner för slutanvändaren. För att förhindra detta har GPLv1 sagt att modifierade versioner måste distribueras i sin helhet enligt villkoren i GPLv1. [1]

Programvara som distribueras enligt villkoren i GPLv1 kan kombineras med program under mer tillåtande termer, eftersom detta inte förändrar de villkor under vilka det hela skulle kunna delas ut.

Programvara som distribueras enligt GPLv1 kan däremot inte kombineras med programvara som distribueras enligt en mer restriktiv licens, eftersom detta skulle strida mot kravet på att det hela skall distribueras enligt villkoren i GPLv1.[2]

GPL version 2 togs i bruk 1991.

Under de kommande 15 åren höjdes kritiska röster från en del medlemmar i FOSS (Free and Open Source Software) gällande GPLv1. De ansåg att en del företag fann kryphål i GPL licensen som gjorde att den kunde användas på sätt som stred mot den ursprungliga filosofin. GPLv2 medförde att om ett företag hamnade i en sådan position att de endast kunde distribuera programmet i binär form så fick de inte distribuera programmet överhuvudtaget. I samband med GPLv2 släpptes även LGPL.[3]

### GPL version 3

Tredje versionen blev offentlig den 29 juni 2007.

Den nya licensen anger speciellt att programvaran inte skall anses vara en effektiv teknisk åtgärd som enligt olika upphovsrättslagar inte får brytas (t.ex. DMCA och DRM). Problemet som uppstått var att programvara licenserad under GPL licens använts i bland annat digi-boxar och mobiltelefoner. GPLv3 medförde att när programvara levereras i samband med hårdvara så ska hårdvaran åtföljas av instruktioner om hur en modifierad version kan installeras. Installation av modifierade versioner får vara omöjlig, men om programvaran kan uppdateras så skall konsumenten också kunna installera en av modifierad version om så önskas. GPLv3 har också en skild formulering som gör det möjligt att kombinera den med programvara licenserad enligt Afero-licensen.

Den största förändringen är att GPLv3 är lättare att kombinera med andra licenser för fri mjukvara.[4]

GNU Lesser General Public License (LGPL) är en uppdaterad och friare version av GPL, ursprungligen avsedd för viss programvara i bibliotekssystem.[5]

GNU Free Documentation License var ursprungligen avsedd för användning med dokumentation till GNU-mjukvara, men har också antagits för andra ändamål t.ex. Wikipediaprojektet. [6]

GNU Affero General Public License (GNU AGPL) är en licens med fokus på nätverk och serverprogramvara. GNU AGPL liknar GNU General Public License, förutom att den dessutom omfattar användning av programvaran i ett datornätverk. Detta kräver i sin tur att källkoden görs tillgänglig i sin helhet för alla nätanvändare i AGPL arbete. Free Software Foundation rekommenderar denna licens för all programvara som vanligen körs i nätverk.[7]

APGL skapades av FSF och är riktad mot tjänster såsom Flickr och Hotmail. Om

någon ändrar källkod licenserad med AGPL och bara publicerar det som en webbtjänst så måste de fortfarande publicera källkoden

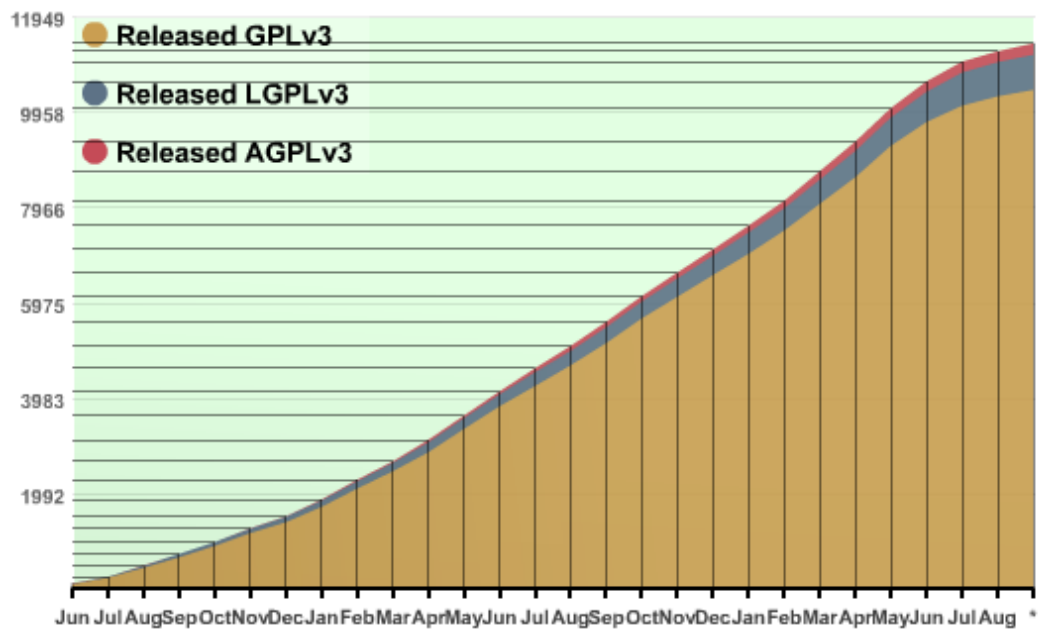


Bild 1: Bilden visar den ökade användning av GLPv3 enligt Black Duck Open Source Resource Center den 21 september 2009. [8]

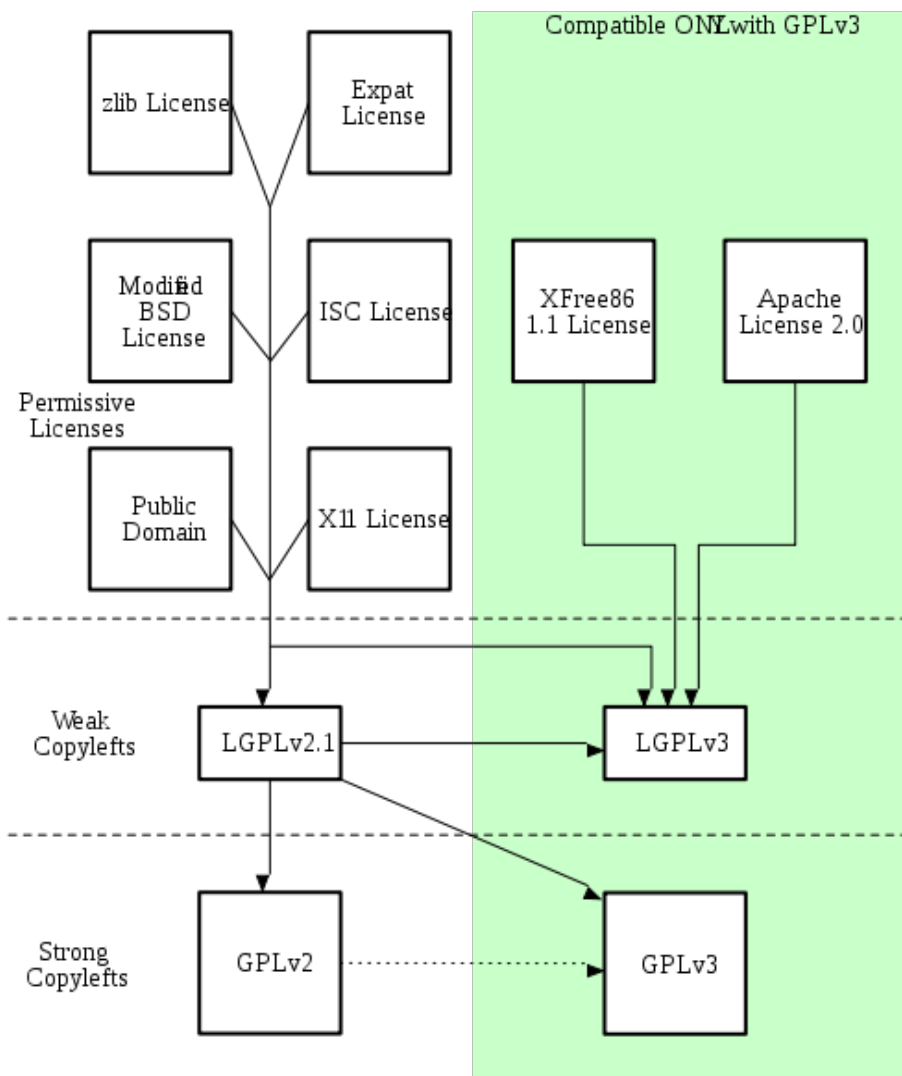


Bild 2: Uppställning över vanliga licenser och deras kopplingar till varandra. Bilden visar även hur de olika licenserna passar ihop.[9]

### Programöversikt

Jag kommer här att redogöra för de program som installerats och använts i Vörå-Maxmo. Samtliga program är installerade på en virtuell server, som i sin tur är uppdelad i flera mindre servrar. Dessa program körs alla genom webbläsaren, och är därför plattformsoberoende. De kan med andra ord användas oavsett om man har Windows, Linux eller Mac OSX installerat.

Jag kommer inte att gå igenom de program som ingår i Linux-distributionen eftersom dessa varierar väldigt mycket beroende på slutanvändaren, och det skulle

därför bli för omfattande.

## **2.4. Moodle**

Moodle är ett informationssystem som används inom skolvärlden för att hantera kurser och ett bra verktyg för lärare att göra undervisningsmaterial tillgängligt för elever.

Moodle faller under kategorin CMS – Course Management System, som ibland även kallas LMS – Learning Management Systems. Moodle används sedan många år tillbaka av en mängd olika högskolor och universitet runt om i Finland och är således ett välbeprövat och pålitligt system.

Moodle var för mig ett självklart program att ta med i projektet eftersom jag själv använt det under många år, och Moodle är trots sitt något karga utseende ett väldigt effektivt verktyg i undervisningen. Vi har inom Openkvarnen projektet under sommaren 2009 arbetat för att förbättra integreringen mellan Moodle och Openmeetings eftersom vi ansett att dessa två skapar en mycket bra inlärningsplattform.

Moodle vidareutvecklas kontinuerligt och mycket aktivt runt om i världen. Australiensaren Martin Dougiamas inledde projektet, och är därför dess huvudperson. Den första versionen släpptes 20.8.2002 efter flera år av tester och prototyper. Efter att den första versionen kom ut har plattformen genomgått en hel del förändringar - nya funktioner har lagts till och prestandan har förbättrats. I februari 2009 hade Moodle översatts till 75 olika språk.

Moodle är utgiven under GNU General Public License, även kallad GLP-licens. Detta betyder att vem som helst får ladda ned applikationen, använda den, modifiera den samt distribuera den vidare. Moodle är PHP-baserad, vilket innebär att den kan köras i vilket operativsystem som helst så länge operativsystemet har en webbläsare. All data lagras i en enda databas, och det är således ingen komplicerad procedur att migrera hela Moodle-systemet till en annan server ifall behov uppstår.

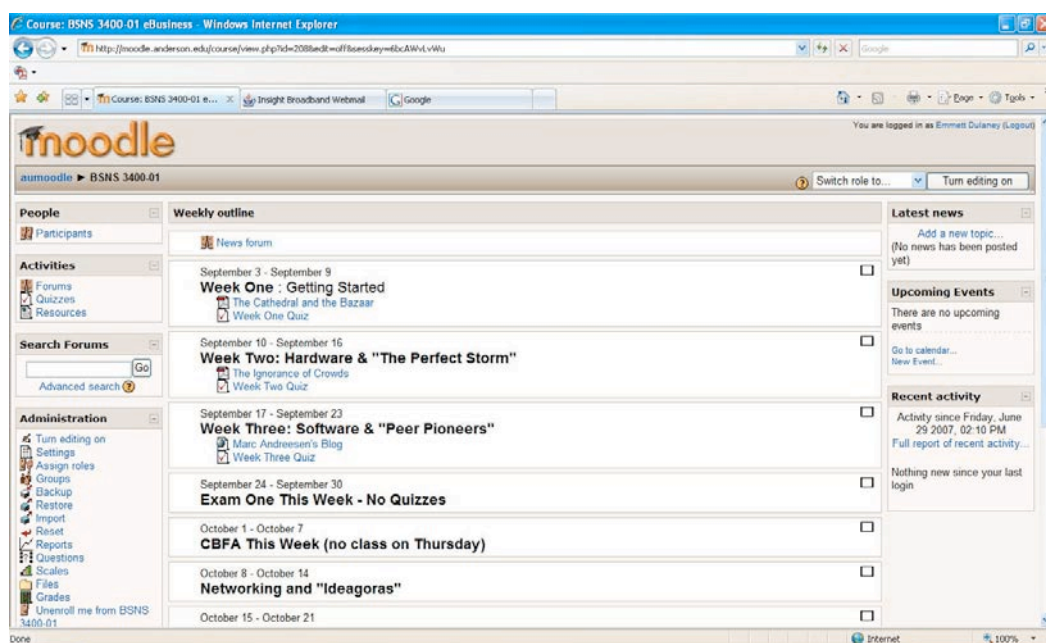


Bild 3: Användargränssnittet i Moodle.[10]

En mindre skola har kanske inte möjlighet att handha och underhålla en egen server, och väljer kanske att hyra utrymme hos ett webb-hotell istället. Om man i ett senare skede bestämmer sig för att byta till ett annat webb-hotell eller kanske upprätta en egen server, behöver man bara spara ned databasen till en dator och sedan ladda upp den till det nya webb-hotellet och den nya Moodle installationen. Ingen information går förlorad.

Det finns etablerade företag i Finland som säljer tjänster och teknisk support till Moodle. Många av dessa ordnar även skolning i och kring systemet.

En lärare kan enkelt bygga upp en kurs i Moodle. Till kursen kan ges innehåll

såsom t.ex. PDF-dokument eller Powerpoint-presentationer. En kurs kan även hållas helt virtuellt. Förinspelade videoklipp kan spelas upp inne i Moodle kursen, och man kan ange tidpunkter för när olika delar av materialet skall göras offentligt. Uppgifter kan således enkelt delas ut på veckobasis, och deadline för olika inlämningsuppgifter kan anges. Inlämningsuppgifterna laddas enkelt upp via moodle och sparas i systemet. Moodle har även ett fullt utvecklat meddelandesystem som inte bara innebär snabbmeddelanden inne i själva systemet, man kan även få t.ex. forum-inlägg och feedback på inskickade uppgifter direkt till sin egen e-post.

## **2.5. Zimbra**

Jag valde att ta med Zimbra som en del i Vörå-Maxmo projektet eftersom jag tyckte att det var viktigt att kunna erbjuda en helhetslösning, och ingen arbetsmiljö är komplett utan epost.

Zimbra erbjuder förutom e-post även möjlighet att ladda upp och spara filer på användarens Zimbrakonto. Detta gör att en användare har tillgång till sina filer oavsett vilken dator man arbetar från.

Jag började med att skapa en virtuell maskin på vår egen server. På den servern installerades sedan Zimbra. Jag ville vara säker på att den produkten jag levererade faktiskt fungerade, så vi testkörde Zimbra internt i några månader, utan att stöta på några problem. Zimbra finns tillgängligt i en rad olika varianter, beroende på hur mycket stöd man behöver från företaget. Open source versionen är helt gratis, men kan inte kombineras med Zimbras avgiftsbelagda tilläggstjänster. I open source varianten saknas även t.ex. funktionen att söka efter bilagor inne i mailen samt versionshantering. Jag gillar Zimbra eftersom det är en komplett ersättare till Microsofts Exchange, och även om man väljer den dyraste Zimbra varianten med alla godsakerna så är den billigare än Exchange.

Zimbra Collaboration Suite, i detta fall (ZCS) 5.0 har utvecklats med fokus på

epost och kalenderfunktioner. Det är den just nu ledande open source lösningen inom området för storföretag, tjänsteleverantörer, utbildningssektorn och myndigheter. Zimbra köptes av Yahoo! för 350 miljoner dollar i september 2007.

Zimbra är licensierat under Yahoo Public License (YPL), vilket är en derivat av MPL. Zimbra Collaboration Suite erbjuder en kraftfull Ajax webbklient som integrerar e-post, kontakter, delade kalendrar, VoIP och dokument-auktorisering i ett rikt browserbaserat gränssnitt. Nedan följer en genomgång av några av de viktigaste funktionerna i Zimbra.

Man kan enkelt skapa, publicera och dela dokument online. Man kan även lägga till bilder och kalkylark med en WYSIWYG-editor och förvandla vanliga dokument till wikin som andra användare kan samarbeta runt.

Meddelandehantering och sök-funktionen i Zimbra är mycket bra. Man kan välja "Konversationsvy" för att endast se de meddelanden som är relevanta till en viss konversation och gömman undan resten. Det går även att tagga viktiga meddelanden för att lättare hitta dem i efterhand, samt använda WYSIWYG verktyget för att skapa sökfrågor.

Zimbra har även en väldigt robust och funktionsrik personlig och delad kalender. Med den kan man enkelt schemalägga resurser och grupper samt dela, publicera och prenumerera på kalendrar. Zimbras kalender är även integrerad med Zimbra Mail. Man kan enkelt hantera kalendern och bokningar medan man skriver sin e-post.

Zimbra använder sig även av en mekanism som kallas Zimlets. Med denna mekanism kan man integrera Zimbra med tredje partens informationssystem och innehåll, samt skapa "mashup" användargränssnitt inne i själva Zimbra. Zimlets sparar tid för slutanvändare genom att koncentrera dagliga sysslor. Istället för att öppna flera applikationer så kan man integrera dem med Zimbra. Till exempel så kan man med Zimlets få kartor eller kalenderinformation att synas i e-posten.

Zimbra har även en kraftfull online dokumenthantering, enkel delning och publicering, ljudintegrering, personalisering. Det faktum att man kunde editera dokument direkt på servern är något som jag själv blev väldigt förtjust i.



Zimbra Collaboration Suite 5.0 har även ett Ajax-baserat admingränssnitt och skriptverktyg för att hantera ZCS servern. Fullt stöd ges för standardbaserade API:n (IMAP / POP / iCal / CalDAV) så väl som MAPI och iSync, vilket gör den kompatibel med andra klienter så som Microsoft Outlook, Apple desktop suite och Mozilla Thunderbird.

En av Zimbras starka sidor är även dess säkerhet. Det har inbyggt anti-virus (ClamAV) och anti-spam (SpamAssassin och DSPAM) Zimbra har även ett säkerhetsramverk med auktoriseringar och krypteringar.

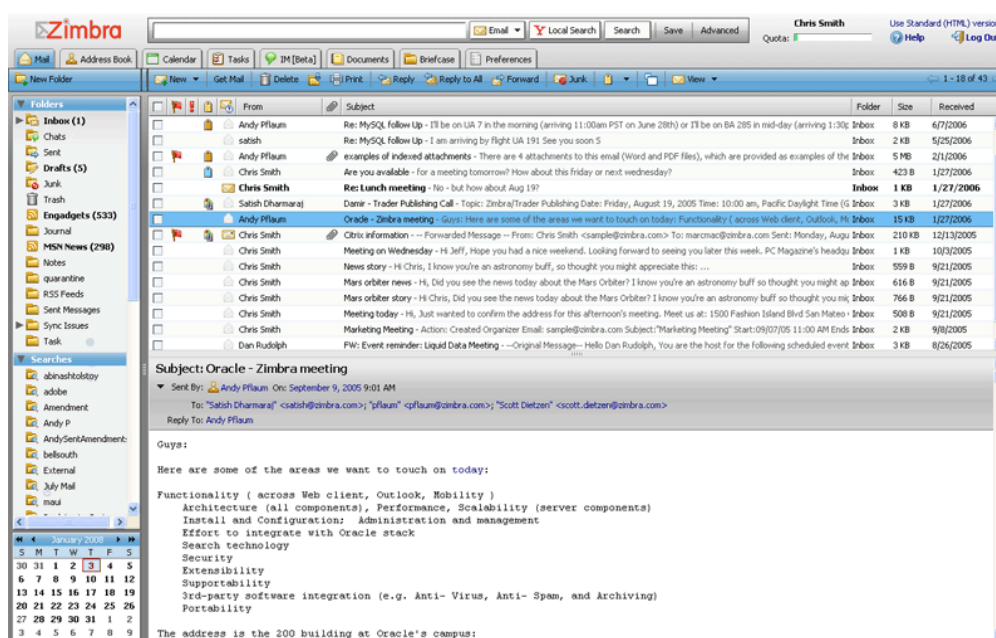


Bild 4: Användargränssnittet för en vanlig användare i Zimbra. Zimbra liknar på många sätt en vanlig e-post klient. [11]

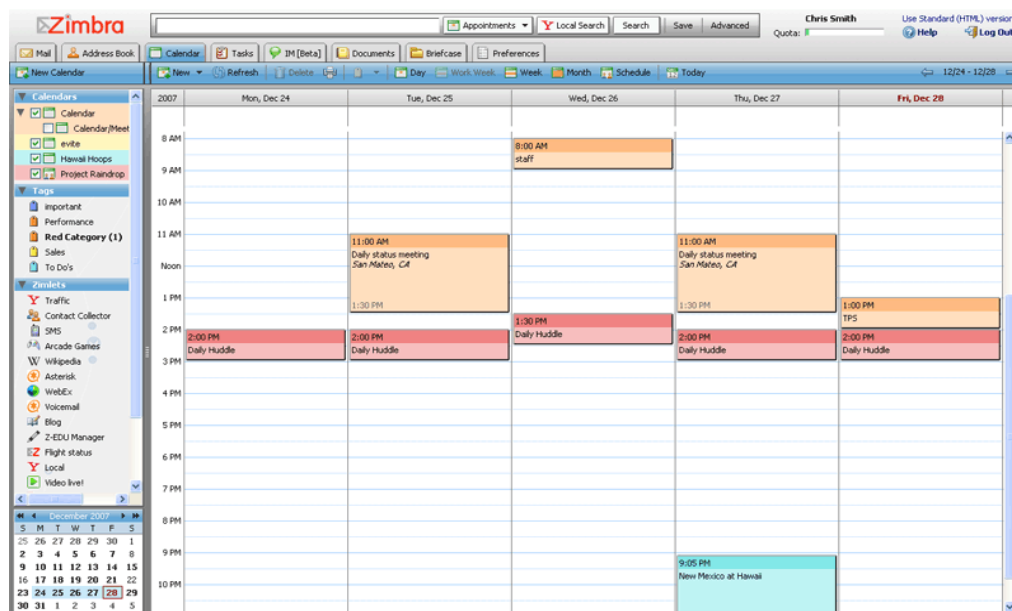


Bild 5: Zimbra har även en avancerad kalender som kan kopplas ihop med t.ex. kalendern i en telefon, eller synkroniseras mot Microsoft Exchange om så önskas. [12]

## 2.6. Föräldramötet

Föräldramötet är resultatet av ett svenskt projekt som drivits inom ramen för Samhällets Informationssystem och är ett samarbete mellan Mittuniversitetet, Sundsvalls kommun, Åkroken och Steria.

Avsikten med programmet är att förbättra, fördjupa och förenkla vardagskommunikationen mellan hem och förskola. Programmet är webbaserat, så var du än befinner dig har du genom Föräldramötet möjlighet att kommunicera med ditt barns förskola.

Det nominerades till Guldlänken 2006, en tävling som belönar den bästa e-tjänsten i offentlig sektor för medborgare och företag. Motiveringen var just att tjänsten ansågs underlätta och förbättra kommunikationen mellan förälder och skola.

I slutet av 2008 granskades Föräldramötet av Datainspektionen. Datainspektionen är en svensk statlig förvaltningsmyndighet som har till uppgift att verka för att

människor skyddas mot att deras personliga integritet kränks genom behandling av personuppgifter. Datainspektionen fann ingenting att anmärka på enligt personuppgiftslagen. Datainspektionen har som uppgift att granska att personuppgifter lagras och används på det sätt som anges i lagen. Föräldramötet har redan använts aktivt i flera år i Sundsvalls kommun.

Föräldramötet i sig är ett lättanvänt program och tar inte många minuter att lära sig. Lärarna eller resurspersonerna bjuder in barnens föräldrar genom att skicka en inbjudan till deras e-post. Samtidigt sätter de även in namnen på deras barn, t.ex. Per Andersson (Förälder till Kalle).

Programmet har en rad användbara funktioner. Lärarna kan använda Kalendern för att informera föräldrarna om kommande händelser i skolan. I Forumet kan såväl lärare som föräldrar skapa trådar och lämna synpunkter, tankar och funderingar. Lärare kan även skriva Nyhetsbrev som föräldrar kan ta del av. De kan även ladda upp och dela med sig av bilder i Bildarkivet och filer i Filarkivet.

Då föräldrarna loggar in på sidan får de en resumé på av det som hänt sedan de senast var inloggade. Därtill kan de även välja att dagligen få samma resumé till deras e-post.

Administratörerna eller resurspersonerna kan dela upp skolorna i olika klasser, t.ex. 1-3 och 4-6, för att få informationen mera inriktad om så önskas.

Föräldramötet påminner väldigt mycket om en vanlig webbsida, och alla funktioner är väldigt lätta att hitta.

Jag valde att inkludera Föräldramötet i mitt projekt eftersom jag tycker att det är ett ypperligt program som verkligen tjänar sitt syfte. När jag presenterat programmet för lärare så har de flesta varit positiva till det. De negativa röster som hörts har varit kring hur tillgängliga lärare kan förväntas vara, och om man kan kräva att föräldrar ska lära sig programmet. Nu ska jag inte börja generalisera, men jag anser att om man kan sätta tid på att lära sig hur Tjäreborgs och

Finnmatkats rese och bokningssystem fungerar samt spendera tiotals timmar i månaden på dessa sidor utan att någonsin boka en resa, då kan man även undvara en halvtimme för att lära sig ett verktyg som gör det möjligt att kommunicera effektivare med lärare och andra föräldrar gällande sitt barns framtid. Gällande

The screenshot shows the 'Föräldramötet' web application. At the top, there is a logo for 'SUNDSVALLS KOMMUN' and the title 'Föräldramötet'. A user is logged in as 'Daniel Lindblad'. The main content area is titled 'Förstasidan' and 'Kalender resumé'. It shows a calendar for the year 2009, with events listed for February and March. The events are as follows:

Dag	Tid	Rubrik	Grupp
15 - Söndag	13:00 - 15:00	Föräldramöte	Skolor 1-3
21 - Lördag	09:00	Skolresa till Vasa	Skolor 1-3
4 - Onsdag	17:00 - 19:00	Hem och Skola seminarie	Skolor 1-3

Below the calendar, there is a section for 'Skolor 1-3' with a 'Kalender' and 'Arkiv' section. The 'Kalender' section shows 'Skolresa till Vasa (13/2)' and the 'Arkiv' section shows 'Nya filer i kategori Information'.

huruvida detta verktyg gör lärarna "för anträffbara" kan jag inte uttala mig om. Enligt migsjälvt är detta dock ett bättre alternativ än att telefonen ringer när favoritserien visas på tv.

Bild 6: Användargränssnittet i Föräldramötet [13]

## 2.7. Openmeetings

OpenMeetings är utvecklat av ett tyskt företag vid namn WebBase-Design. Huvudmannen heter Sebastian Wagner. Det är i skrivande stund fortfarande i beta-stadie, men fungerar redan nu ganska bra.

Programmet är ett webbaserat vilket gör det helt plattformsoberoende. Det spelar ingen roll om du använder Windows, Linux eller Mac. Det enda som behövs är att du har en webbläsare som stöder flash och java. Openmeetings är skrivet mestadels i XML och Java och är licensierat enligt LGPL.

OpenMeetings är anpassningsbart och har flera språk. Vi har inom Openkvarken

projektet översatt programmet till både finska och svenska. Språktilläggen finns tillgängliga på Openmeetings hemsida.

Openmeetings har stöd för ljud och bild. Det innehåller en whiteboard till vilken det är möjligt att importera en mängd olika filformat t.ex. PDF, PPT eller ODT och visa dem för alla deltagare. Programmet innehåller även mindre funktioner såsom inbjudningar till möten och möjlighet att spara en bild av vad som visas på whiteboarden. Openkvarken-projektet planerar även att stödja utvecklingen av en funktion som gör det möjligt att spela in videokonferanser.



Bild 7: Användargränssnittet i Openmeetings. I vänstra hörnet finns en lista på deltagare och deras rättigheter. Det finns även en whiteboard som man kan dela dokument på, samt en chat för de som kanske inte har tillgång till mikrofon.

OpenMeetings är väl lämpad för användning i utbildningen tack vare den utmärkta Moodle integration. Möten kan schemaläggas direkt i Moodlekurser och mötet öppnas inuti Moodles användargränssnitt.[14]

## 2.8. LDAP, lightweight directory access protocol

LDAP är ett protokoll som används av en rad programvaror och lösningar. Det används för att över ett nätverk söka i en katalog. Katalogen kan innehålla många

olika uppgifter såsom person- och företagsuppgifter, certifikat för olika säkerhetstjänster och behörighetsregler. LDAP håller alltså reda på vilka rättigheter ett visst användarkonto har, och distribuerar resurser därefter.

LDAP utvecklades från början för att kunna söka i X.500-kataloger som en internetanpassning av OSI-protokollet DAP. LDAP har under åren utkommit i flera versioner. Man rekommenderar nu användning av LDAP v3, som innehåller en bättre funktionalitet för bl.a. nationella tecken och hänvisningar mellan olika servrar. LDAP v3 är baserat på ISO 10646, dvs Unicode. En internationell teckenkod med stöd för all världens tecken, samma som är basen i XML, the Extensible Markup Language. LDAP använder som standard port 389.[15]

I motsats till många andra TCP-baserade protokoll, som HTTP, FTP och TELNET, så är LDAP inget protokoll som sänder kommandon i klartext över nätet. LDAP använder en kodning av transaktionerna som heter BER, Binary Encoding Rules.

## 2.9. Definition av Linuxdistribution

Ett operativsystem baserat på en Linux-kärna kallas för en Linux-distribution. Den programvara som finns i ett linuxsystem är helt eller till största delen fri programvara, vilket betyder att den lätt kan laddas ned över internet och lagligt modifieras, ompaketeras och vidare distribueras. Arbetet att samla programvara och källkod för att skapa en fungerande helhet är mycket tekniskt krävande. När Linux blev populärt var det därför naturligt att en del av dem som gjort detta arbete lade ut sitt verk att utnyttjas av andra, varvid de första linuxdistributionerna föddes. Olika delar från olika distributioner slogs samman, och efter hand växte distributionerna till mycket sofistikerade helheter som kunde installeras och administreras utan alltför stora kunskaper.

Det har under de senaste åren kommit ett tiotal olika distributioner som vänder sig till en specifik användargrupp. De vanligaste distributionerna idag är Red Hat och Debian, men distributioner såsom Fedora och Ubuntu ökar kraftigt i användarantal. Alla distributioner kan framstå som en djungel för den som inte är insatt, eftersom vissa distributioner baserar sig på andra linux-distributioner. Ett exempel är Ubuntu som ursprungligen är en Debian distribution, men som försetts med ett grafiskt gränssnitt kallat X Window System av versionen Gnome.

GNU-projektets GPL-licens, som används av många centrala program kräver att alla innehavare av ett program skall ha tillgång till full källkod samt ha tillstånd att modifiera och vidare distribuera programmet. Därför finns nästan alla linuxdistributioner fritt tillgängliga på Internet, både som källkod och som installerbara paket.

Användargrupper och ändamål inom Linux.

En del distributioner har utvecklats för att rikta sig till en specifik användargrupp, andra är avsedda för att vara lämpliga i så många avseenden som möjligt.

Ofta är avsikten att ett företag väljer en distribution för att använda den på alla sina Linux-datorer, såväl arbetsstationer som servrar och brandväggar. Dessa distributioner är vanligtvis Debian GNU/Linux eller Red Hat Enterprise Linux

eftersom dessa i allmänhet är ett tryggt val för den som inte har specialbehov. Distributionerna skiljer sig ifråga om omfattningen av programutbudet, finslipning av användagränssnitten men även på punkter såsom hur ny och hur vältestad programvara som anses lämplig för distributionen.

En del distributioner såsom t.ex. Ubuntu och Mandriva är avsedda i första hand för persondatorer. Distributioner som speciellt utvecklats för att stöda grafik och spel är Linspire och Xandros. Dessa har även ökad kompatibilitet med Windows.

Det finns också ett antal projekt för ovanligare hårdvaruplattformar eller speciella ändamål. Bland dessa hittar man t.ex. Damn Small Linux för situationer där resurser som minne och hårddiskiva är begränsade, Edubuntu och Skolelinux som är inriktat på skolor och Knoppix för körning från CD-skiva utan installation.

## **2.10. Distributioner som använts i projektet**

Ubuntu är en Debianbaserad distribution av Linux som utvecklats av företaget Canonical Ltd. Namnet Ubuntu kommer från Zuluspråket, och betyder "medmänsklighet". Detta ord sammanfattar ganska bra filosofin bakom Ubuntu, nämligen att vara ett enkelt och stabilt operativsystem som riktar sig till genomsnittsanvändaren. Ubuntu's starkaste fokus ligger på användarvänlighet och enkel installation. Ubuntu och människorna inblandade samlas runt en grundläggande filosofi som ska garantera användaren vissa friheter: att mjukvara ska finnas tillgänglig utan kostnad, kunna användas av alla oavsett språk eller eventuella handikapp, samt att människor ska ha friheten att modifiera och anpassa sin mjukvara på vilket sätt de själva vill. Dessa friheter gör att Ubuntu skiljer sig markant från traditionell, proprietär programvara. Alla verktyg och program i Ubuntu finns tillgängliga helt gratis och användaren har fullständig rätt att ändra på dem tills de uppfyller dennes behov.



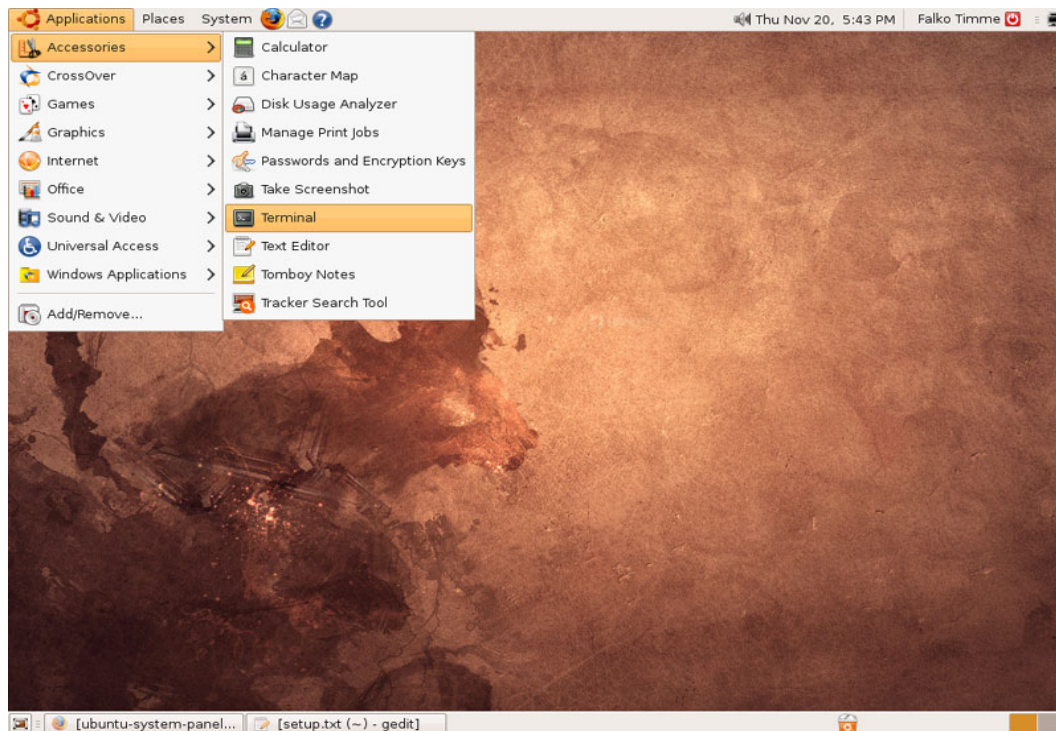


Bild 8: Ubuntu 8.10 skrivbordet i standardutförande [16]

Ubuntu har vidareutvecklats till en rad olika varianter, bland annat:

Edubuntu är speciellt anpassad för att användas i skolor.

Kubuntu använder KDE som skrivbordshanterare (istället för Gnome som är standard i Ubuntu).

Xubuntu använder Xfce som skrivbordshanterare och är speciellt anpassad för att kunna köras på äldre datorer.

Edubuntu innehåller ett stort antal pedagogiska applikationer som GCompris, KDE Edutainment Suite, and Schooltool Calendar. Edubuntu är utvecklat för att lärare med mindre datorkunskaper enkelt ska kunna sätta upp en fungerande datormiljö och förbättra undervisningen.

Edubuntu använder sig precis som Ubuntu av Gnome-gränssnittet, och är i det avseendet närmare besläktat med Ubuntu än sina syskon Kubuntu och Xubuntu.



Bild 9: Edubuntu 8.10 i standardutförande [17]

Det stod i ett tidigt skede klart att jag skulle komma att basera min egen distribution på Edubuntu, eftersom den innehöll skolprogram och hade ett stabilt användargränssnitt tillsammans med en väl fungerande filhantering. Tyvärr medförde användningen av Gnome att många av de KDE-baserade skolprogrammen inte fungerade som de skulle, men jag förklarar längre fram varför jag valde Gnome framom KDE.

Xubuntu (uttalas Zubuntu) är en variant av linuxdistributionen Ubuntu som använder sig av skrivbordsmiljön Xfce. Xubuntu är därigenom mindre systemkrävande än motsvarigheterna Ubuntu (som använder GNOME) och Kubuntu (som använder KDE). Xubuntu lämpar sig för att användas på datorer som kan anses vara för långsamma för att köra de andra Ubuntu-varianterna, för användare som uppskattar snabbheten i ett lättviktigt system, eller för användare som helt enkelt tycker om fönsterhanteraren Xfce bättre.

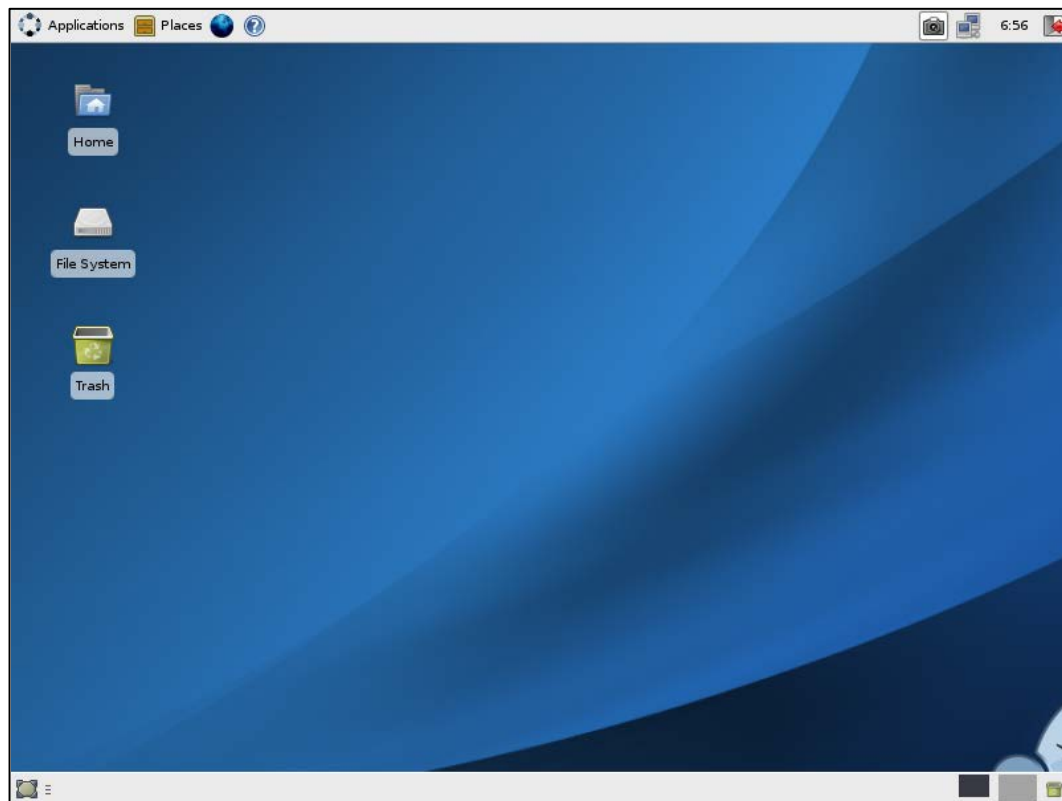


Bild 10: Xubuntu 8.10 skrivbordet i standardutförande. [18]

Jag använde mig av Xubuntu vid konstruerandet av Linux-datasalen i Oravais eftersom det var den distribution som fungerade bäst på deras gamla datorer. Personligen hade jag helst installerat Ubuntu på alla datorer, men efter att ha konstaterat att Ubuntu blev alldeles för långsamt valde jag Xubuntu med en del tillägsprogram installerade.

Kubuntu är en variant av Ubuntu försett med KDE grafiskt gränssnitt. K Desktop Environment även kallat KDE, är en skrivbordsmiljö som är tänkt att efterlikna utseende och känsla från windows. Största delen av gränssnittet går att konfigurera grafiskt, vilket gör det lättare för personer som inte är vana vid att hantera terminaler och arbeta textbaserat. Jag övervägde att använda KDE gränssnittet för att minska tröskeln från Windows, men till följd av att KDE 4.2 är mer prestandakrävande än Gnome gränssnittet valde jag att avstå.



Bild 11: KDE 4.2 skrivbordet den 28:e november 2008.[19]

I detta skede var KDE 4 inte ännu klassat som ”stable”, vilket betyder att en del ”barnsjukdomar” och oförutsedda fel kunde uppstå. Dessa skulle givetvis ha försvunnit genom att man uppdaterat systemet när KDE 4 väl klassats som stable, men jag valde att i nuläget använda Gnome-gränssnittet för min egen distribution.

Möjligheten att övergå till KDE-gränssnitt i ett senare skede kvarstår ifall att man om några år känner att KDE blivit mer fördelaktigt än Gnome. I praktiken innebär detta att man installerar KDE-gränssnittet på datorerna, och vid inloggningsskärmen kan man sedan välja vilket gränssnitt man vill göra. Det hela är dock en avvägning man måste göra eftersom KDE 4.2 och senare KDE 4.3 märkbart kommer att försämra datorns prestanda till förmån för grafisk upplevelse. Enligt mig börjar de verkliga fördelarna med KDE vid införandet av plasmoid, dvs version 4.3.

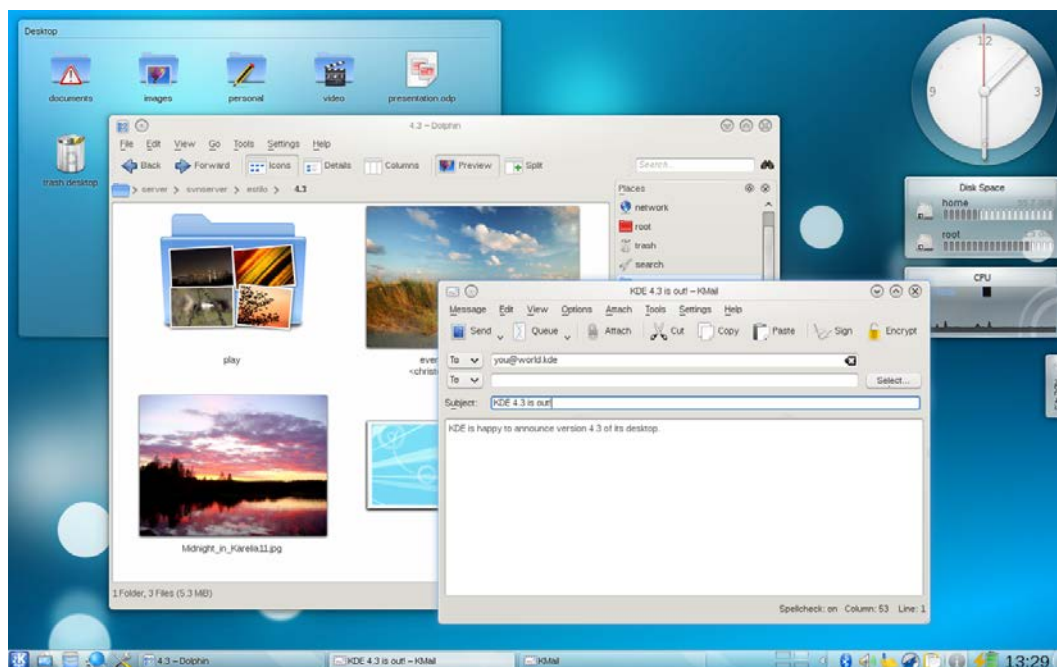


Bild 12: KDE 4.3.2 skrivbordet. Införandet av plasmoid gör att alla komponenter på skrivbordet fritt kan flyttas runt, motsvarande funktion hittas i Windows Seven. [20]

#### Vörå – Maxmo distributionen

Den färdiga produkten blev en skräddarsydd EdUbuntu distribution med gDesklets panel installerad. Den grafiska utformningen lånade jag till en del från Skol-linux, men jag valde Gnome skrivbordet som bas eftersom den kändes som det stabilaste alternativet. Jag övervägde länge att använda KDE4 gränssnittet eftersom jag tror att inläringströskeln skulle ha varit lägst och övergången smidigast, men till följd av att gränssnittet i KDE4 i nuläget inte är helt färdigutvecklat och dras med en del barnsjukdomar valde jag att satsa på det gedigna Gnome gränssnittet.

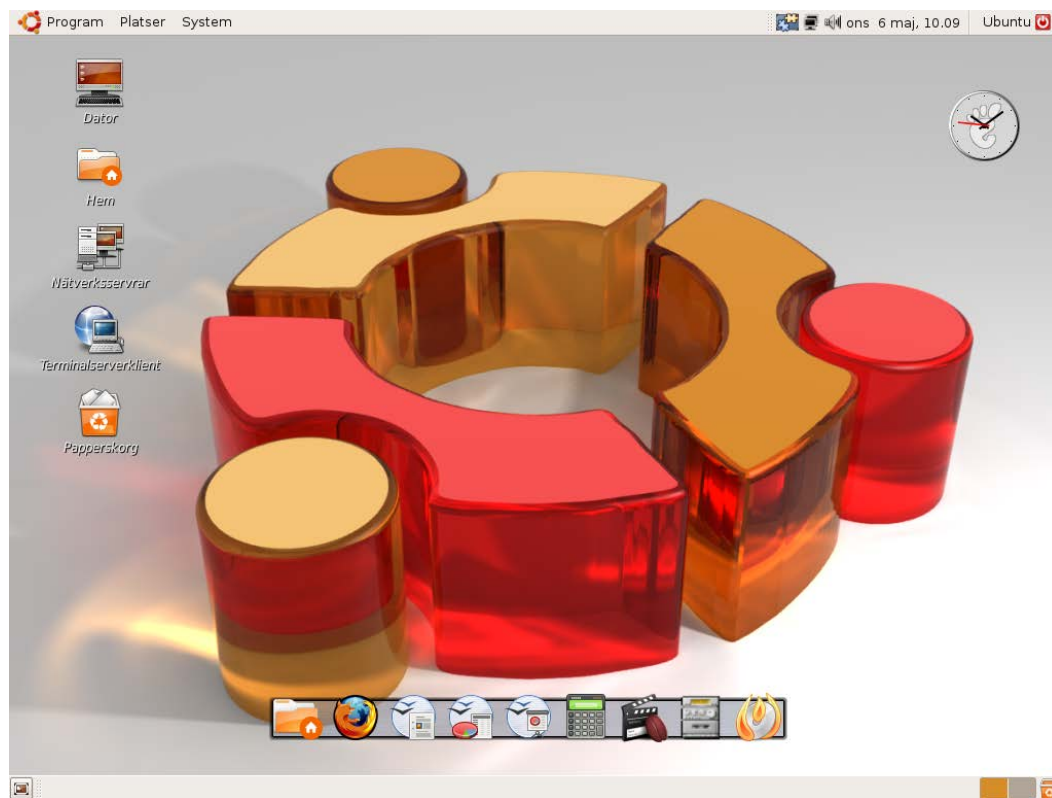


Bild 13: Den färdiga distributionen med gDesklets starterbar installerad.

Ubuntu 9.04 lanserades dagarna före färdigställandet av min EdUbuntu version åt Vörå-Maxmo, och då väcktes även frågan ifall jag skulle köra 8.10 eller 9.04 som bas. Efter ett par dagars testning och efterforskningar kom jag dock till den slutsatsen att 9.04 inte var stabilt nog för att implementera i så stor skala. Vörå-Maxmos IT-ansvarige Kenneth Nickull var även han av den åsikten att systemets stabilitet ska ha högsta prioritet.

Jag har försökt hålla utseendet så minimalistiskt som möjligt, samtidigt som jag placerat de viktigaste genvägarna lättillgängligt. Idén med panelen och genvägarna lånade jag från Skol-linux, som i sin tur förmodligen inspirerades av Mac OSX.

På panelen finns genvägar till webbläsare, Open office programmen Write, Calc och Impress samt en kalkylator och mediaspelare. Jag placerade dessa program lättillgängliga eftersom jag antog att det var de program som skulle komma att användas flitigast.

För varje elev finns en mapp på skrivbordet som heter "Hem". Alla filer i denna mapp sparas på filservern, och elever har alltid tillgång till filer i denna mapp oavsett vilken dator de loggar in från. En elev kan endast se sin egen mapp och sina egna filer emedans en lärare har möjlighet att kontrollera innehållet i alla elevers mappar. Tanken bakom användandet av mapparna var att inga användarfiler skulle sparas på klientdatorn, och ifall att ett problem uppstår med en dator så kan man på några minuter återställa den med Clonezilla utan att några användarfiler försvinner. En återställning med Clonezilla tar ca 5 minuter, vilket ur en systemadministratörs perspektiv är en mycket effektiv lösning när man har ett stort antal datorer att underhålla.

Ett problem som uppstår när man har många datorer kopplade mot en relativt långsam internetuppkoppling är säkerhetsuppdateringar. Istället för att 80 st datorer laddar ned sina uppdateringar skilt för sig så konfigurerades LDAP-servern till att distribuera säkerhetsuppdateringar i det egna nätet. Säkerhetsuppdateringarna laddades alltså ned till LDAP servern, och alla andra datorer laddade sedan ned sina säkerhetsuppdateringar från LDAP servern. Detta ökade effektiviteten avsevärt eftersom det mellan klientdatorerna och LDAP servern finns kapacitet på 100mb/sekund, emedans hastigheten från internet skulle ha varit endast 8mb/sekund.

USB-minnen och liknande enheter som kopplas till datorn kommer att dyka upp som ikoner på skrivbordet. På Kenneth Nickulls begäran lade jag även till en genväg för att använda terminalserver. Detta innebär att man från dessa datorer kan logga in mot Windows servern i Vörå och arbeta på den genom ett virtuellt skrivbord, förutsatt att man har de rätta användarrättigheterna. Man kör med andra ord Windows server fönstret inuti sin Ubuntu dator. Orsaken till detta var att alla betyg skrivs in på huvudservern i Vörå, och man önskade en lösning som innebar att lärarna i Maxmo inte behövde köra till Vörå bara för att skriva in elevers betyg.

Skol Linux är en Debian-distribution framtagen av Roger Andersson och Magnus Sundfors.

I samband med anmälningarna till vårt första seminarium i Vasa kom vi i kontakt med Roger Andersson och Magnus Sundfors i Korsholm. Roger och Magnus arbetar som IT-stöd personer för Korsholms kommun och skolor, och de har sedan några år tillbaka endast använt Skol-linux i lågstadie och högstadieskolorna. På Rogers initiativ har Korsholms kommun även övergått till att använda OpenOffice framom Microsoft Office 2003.

Skol Linux togs fram för att öka effektiviteten i dataklasserna. Med de tidigare programvarorna kunde en uppstart kräva 15 minuters väntan, vilket betyder att 1/3 del av den 45 minuter långa lektionen var bortkastad. En uppstart med Skol linux tar under en minut.

Tack vare Skol linux kunde man blåsa nytt liv i den alltför snabbt åldrande datasalen och effektivera undervisningen. Man kunde även bli fri från alla licenskostnader, och på så sätt investera pengar i bättre hårdvara. För de inbesparade licenspengarna har man t.ex. köpt ett antal Asus Eee-PC, s.k. ”mini-laptops” som eleverna kan använda. Man har även installerat trådlöst internet i skolorna som står till elevernas och personalens förfogande.

Fördelarna med Skol-linux förutom prestandan och kostnadsbesparingarna är gränssnittet.

Användarmiljön och placeringen av ikonerna är anpassad för att vara så lättillgänglig som möjligt. Man har även gjort begränsningar i användarrättigheterna för att försöka förhindra att skoleleverna orsakar skador eller installerar egna program på datorn.[21]



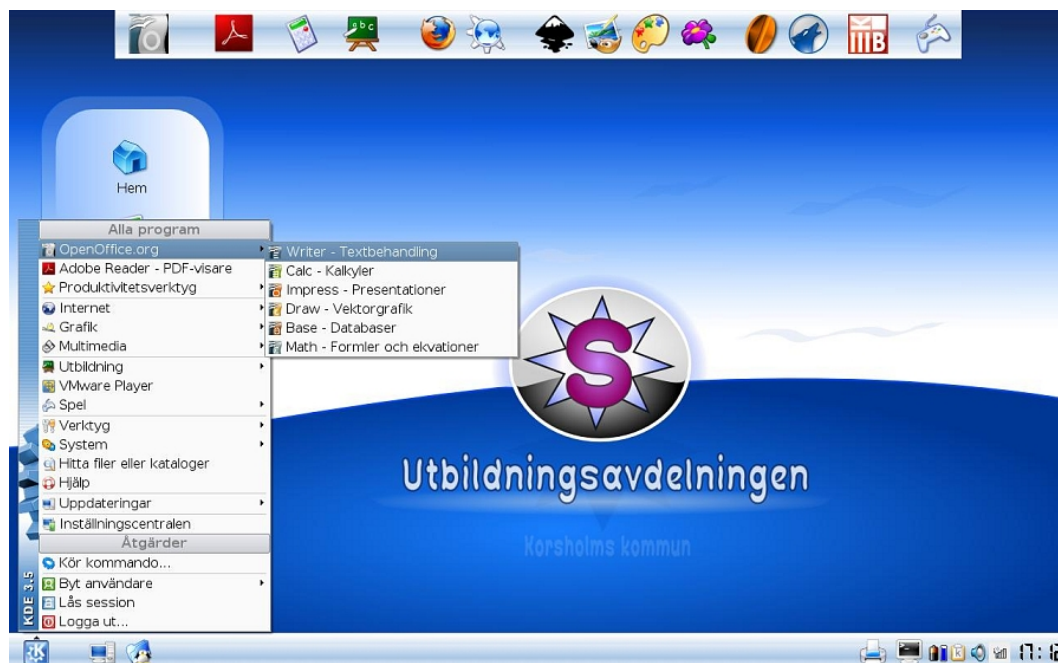


Bild 14: Skol-linux används på 90% av alla datorer i Korsholms skolor [22]

### **3. FAS 1: TESTNING OCH UTVÄRDERING**

#### **3.1. Acer minilaptops**

Jag ville lyfta fram s.k. mini-laptops med Linux operativsystem som ett bra hjälpmedel för undervisning i t.ex. grundskolor. Openkvarken-projektet införskaffade 16 st Acer Aspire 150A med 8,9” skärm. Eftersom dessa datorer har en begränsad kraft, gick vi metodiskt igenom alla operativsystem som kunde vara relevanta. Vi testade flera operativsystem för att undersöka vilka som fungerade smidigast, däribland Ubuntu, Skol-linux, Windows Xp, Vista och betaverisionen av Seven.

Genom Roger Andersson och Magnus Sundfors vid Korsholm kommun kom vi i kontakt med deras egen distribution Skol-linux. Vi testade alla funktioner, och operativsystemet fungerade överlag mycket bra. Problemet med en egen distribution är att det kräver en Linux-kunnig person för att anpassa och vidareutveckla. Många skolor har dock ingen egen IT-person, utan köper istället kunskap utifrån. Vi bestämde att vi kommer att hålla oss till Ubuntu-distributionen framöver eftersom det är ett projekt som fått finansieringen tryggad för många år framöver, och Ubuntu är tämligen känt hos allmänheten.

Vi testade Ubuntu 8.10, men stötte på en del problem eftersom minimi-storleken på en del av fönstren var större än skärmen.

I sambande med att Ubuntu 9.04 lanserades släpptes en version med namnet Ubuntu Netbook Remix. Detta är en Ubuntu distribution som är skräddarsydd för att passa datorer med små skärmar och har drivrutiner till bland annat webcammen och det trådlösa nätverkskortet, vilket saknas i en vanlig distribution.

Menyerna har även anpassats för att bättre passa en liten laptop. En annan bra funktion är att alla fönster automatiskt fyller ut skärmen.

Dessa datorer lånades ut till olika skolor för testning.

I de första testerna användes Skol-linux, och responsen var positivare än väntat.

Man ska även ta i beaktande att vi knappt gav någon utbildning i hur man använder operativsystemet.

Som en del i projektet hade 16st Acer Aspire One datorer, s.k. mini-laptops införskaffats. Tre av dessa skickades till Umeå för att användas som marknadsföring, de övriga 13 datorerna stannade i Vasa. På datorerna som stannade i Vasa installerades Skol-Linux. Dessa 13 datorer lånades ut till lågstadielklasser runt om i Österbotten.

Detta hade tre huvudsyften.

1. Att utreda hur väl skolbarn anpassar sig till en ny användarmiljö (övergång från Windows XP till Debian KDE Distribution)
2. Att sätta Skol-Linux i användning och få feedback som kan användas för framtida utveckling av systemet.
3. Att utreda vilka effekter en minilaptop som läromedel medför.

### **3.2. Skol-Linux och Mini-laptops i Nykarleby**

Den första utlåningen skedde den 08.02.2009.

Gruppen var årskurs 4 vid Normen skola i Nykarleby.

Maria Sjöblom arbetar som klasslärare för årskurs 4. Hon hade läst en av broschyrerna vi sänt ut och kontaktade mig den 4 februari angående möjligheten att få låna datorerna.

Jag besökte Normen skola fredagen den 8 februari och höll en kort presentation av Openkvarken projektet samt en snabb förklaring av konceptet öppen källkod. Jag höll även en snabb skolning där jag presenterade funktionerna och programmen i Skol-Linux.

Jag fick ett väldigt varmt bemötande och intresset för de små datorerna var stort.

Vi kom överens om en lånetid på 3 veckor. Jag bad Sjöblom att skriva en kort redogörelse över hur de uppfattat Skol-linux arbetsmiljö samt hur det fungerat att använda datorerna i en lågstadielklass.



Bild 15: Vi använde Skol-linux på våra egna minilaptops i utvärderingssyfte. Gränssnittet är till synes mycket likt det Korsholm använder.

#### Feedback från Nykarleby

Den feedback jag fick från Maria Sjöblom var mycket positiv.

Hon berättade att eleverna arbetat kring ett projekt i Natur och miljö, och fick som uppgift att skriva faktatexter kring de nordiska länderna. De skulle söka fakta och bilder på internet och sedan sammanställa dessa till ett dokument på minilaptopsen.

För att testa hur eleverna arbetar om de får ha en egen dator med sig fick några av eleverna ta med sig datorn hem när skoldagen var slut. Detta fungerade enligt henne alldeles utmärkt, och hennes vision är att varje elev skulle kunna bära med sig sin egen dator i en vadderad ryggsäck.

De enda problem de stött på var att de kört datorerna på batteridrift och batteriet tagit slut utan att eleverna sparat sitt arbete.

Hon betonade de stora fördelarna med att eleverna kunde arbeta var de ville samt att datorerna passade mycket bra även för barn.

## **4. FAS 2: KONVERTERING**

### **4.1. Konvertering av datasal i Oravais**

Jag kontaktades den 19 februari av klasslärare Jan Hermans vid Centrumskolan i Oravais.

Han skickade en kort beskrivning av situationen, där han berättade om en datorklass med gamla datorer som hade Windows NT eller XP installerat. Datorerna var i nuvarande form näst intill oanvändbara eftersom de låste sig med jämna mellanrum och elevernas arbeten gick förlorade.

Vi diskuterade möjligheten att installera Ubuntu/Edubuntu på datorerna, och jag erbjöd mig att göra i ordning en dataklass under det kommande sportlovet.

### **4.2. Datorer och problemställningar**

Jag och min kollega Mikael Torp åkte till Oravais den 20:e februari för att träffa Jan Hermans och för att granska datorerna. Datorerna i fråga var Fujitsu-Siemens Scenic T och Scenic P modeller från tidiga 2000-talet. De nio datorerna hade som bäst 256mb ram att tillgå och samtliga var i väldigt dåligt skick. Det stod ganska snabbt klart att Ubuntu i dess standardform var uteslutet.

Efter en kort diskussion valde vi att försöka installera Xubuntu istället för Ubuntu. Xubuntu är en kraftigt nedskalad variant av Ubuntu, och har därför mycket lägre krav på systemet.

Xubuntu i sitt standardutförande är dock väldigt begränsat, och innehåller t.ex. inte OpenOffice eller liknande program. Alla program man vill använda i Xubuntu måste därför laddas ned och installeras.

Här uppstod därmed ett annat problem, nämligen den kraftigt begränsade tillgången på bandbredd.

Centrumskolan hade endast 2mb internetanslutning, vilket delat på hela skolan gav en verklig nedladdningshastighet av 30kb/sekund. Vi var därför tvugna att

planera och schemalägga installationer och uppdateringar noggrant.

### **4.3. Installationer i Oravais**

Tisdagen den 24 februari åkte jag och Daniel Lindblad till Oravais för att installera Xubuntu på datorerna. När vi påbörjade installationen visade det sig att datorerna inte hade den kapacitet som krävdes för att köra en vanlig installation, den så kallade "Live CD"-installationen. Detta var dock ett problem som jag förutsett, och jag hade därför även skapat 5 stycken "Alternate install CD" vilket är en av datorn mycket mindre krävande installationsform. Slutresultatet är detsamma, men en "Alternate install" är mer komplicerad och kräver större kunskaper av den som utför installationen.

Alternate install versioner av Xubuntu fungerade utan större problem men det visade sig att de tre äldsta datorerna, de av Scenic T modell inte riktigt fungerade på önskvärt sätt. De verkade relativt stabila, men eftersom de endast hade 128mb ram-minne var de märkbart långsammare än Scenic P modellerna som hade 256mb minne. Vi lät dock saken bero och startade uppdateringsprocesserna innan vi återvände till Vasa. Vi beräknade den totala tiden för uppdateringen till 30 timmar.

Under onsdagen den 25 februari fick jag genom Teemu Haapoja tag i avlagda Osborne datorer donerade från Vasa yrkeshögskola. Datorerna hade 2,4ghz processor och 1gb ram-minne, och fyllde mer än väl kriterierna för en Ubuntu/Xubuntu installation. Jag beslutade att ta med tre av dessa till Oravais för att ersätta Scenic T datorerna. Jag installerade Xubuntu på Osborne datorerna och gjorde alla behövliga uppdateringar.

Torsdagen den 26 februari åkte jag, Daniel Lindblad och Mikael Torp till Oravais. Med oss hade vi de tre Osborne datorer som donerats av Vasa yrkeshögskola.

### 4.3.1. Printning

En av datorerna agerade som print-server. Eftersom jag var osäker om några andra dataklasser använde denna dator för att printa så valde jag att göra en double-boot installation, d.v.s. jag behöll den gamla XP installationen samtidigt som jag installerade Xubuntu vid sidan om. Användaren har vid uppstart möjlighet att välja om han eller hon vill köra Windows XP eller Xubuntu. Jag gjorde även alla inställningar som behövdes för att datorn ska kunna agera som print-server i Xubuntu miljö. Ironiskt nog gick denna dator sönder efter några dagar, och eftersom det visade sig att de inte hade något behov av att behålla Windows-installationen så gjorde jag en Ubuntu-installation som fick agera som CUPS server.

Det visade sig efteråt att de önskade att alla printningar skulle ske genom deras nätverksskrivare. Skrivaren var av modellen Xerox Workcentre 7242. Enligt linuxprinting.org borde nätverksskrivaren ha klarat av att hantera postscript. Att använda tillverkarens postscript-drivrutiner resulterade dock bara i att printern printade 1000 sidor med drivrutinens namn. Efter att ha diskuterat med Xerox återförsäljare fick jag veta att skrivaren antingen har PCL eller Postscript installerat beroende på vad som beställts vid inköpstillfället, samt att man kan stöta på problem ifall man försöker installera PCL och Postscript sida vid sida. Jag fick nätverksskrivaren att fungera genom att använda Generic PCL 6 Foomatic drivrutiner.

### 4.3.2. Fil- och Serverhantering

Jan Hermans hade påpekat behovet av att kunna spara filer på deras Windows 2008 server. Servern i fråga administrerades av KD Soft, och jag kontaktade dem för att få stöd för att gällande att lägga till Linux klienter i Windows Active directory samt att eventuellt skapa inloggning för klienterna.

KD Soft hade tyvärr ingen erfarenhet av integrering av Linux klienter i Windows miljö, och jag var tvungen finna lösningen på egen hand.

Jag övergav tankarna på en inloggning via LDAP eller motsvarande eftersom :

- 1) Detta var något jag aldrig försökt tidigare
- 2) Jag ville inte experimentera med någon annans server

### Serverkonfigurationer

Standardanvändar-gruppen på servern tillät inte anslutningar från icke-windows klienter. För att åtgärda detta skapade jag en användargrupp på servern som jag döpte till "linux" och gjorde samtliga användare till medlemmar i den gruppen. Jag skapade med andra ord en grupp som kunde nås på adressen [orask.local/Users/linux](http://orask.local/Users/linux) och gjorde samtliga Xubuntu datorer medlemmar i den gruppen.

Jag var även tvungen att skapa en länk mellan lärarnas mappar och deras användarkonton eftersom dessa tidigare varit åtskilda på servern men bundits ihop genom en specialkonfiguration i Windows. Detta har troligen gjorts för att mapparna inte ska synas när man bläddrar i nätverket. Efter att jag gjort mina ändringar syntes alla lärares mappar i nätverket, men man kunde inte nå dem utan rätt lösenord.

### **4.3.3. Problemlösning för kommunikation med Windows servern**

Nu när samtliga datorer kunde ansluta till servern så gällde det att få fram ett sätt för användaren att nå sina egna mappar på servern. Som jag tidigare nämnt så var en standardinloggning via LDAP uteslutet, och filhanteringssystemet i Xubuntu saknade stöd för "bläddring" i nätverk.

Jag och Mikael Torp diskuterade problemet och bestämde oss för att den snabbaste och säkraste lösningen får vi genom att programmera den själv.

Ett ständigt återkommande problem var att det inte finns något säkert sätt att eliminera processen med "Sudo"-lösenord och autentisering i Ubuntu-distributioner, och vi bestämde slutligen att vi måste programmera en lösning där



även sudo-lösenord anges första gången. Att eliminera sudo-lösenord från systemet samt att integrera lösenordet i programmet ansågs båda vara säkerhetsrisker samt att det kunde orsaka problem längre fram. Efter en del programmerande hade vi en lösning som såg ut enligt följande.

```
echo Skriv in användarnamn, t.ex. start2003
//Skriver ut texten för att guida användaren.

read anvandar
//Sparar angiven text i variabeln anvandar

echo Skriv in lösenordet: qwerty
//Första gången scriptet körs kommer systemet att
fråga efter sudo-lösenord. Denna text är till för
att guida användaren.

sudo " " //Ifall att lösenordet för sudo angetts kommer
detta bara att

echo " " skapa ett tomrum mellan nästa textsträng

echo Skriv in lösenord för användarkontot $anvandar
//Skriver ut text som frågar efter lösenord till det
användarkonto man angett.

sudo mount -t smbfs //192.168.xxx.xx/$anvandar
/media/windowsdomain -o username=$anvandar
//Skapar en virtuell enhet av användarmappen på
servern inne i en Xubuntu mapp.
```

Vår lösning resulterade i att användaren fick en mapp på datorn som var direkt kopplad till mappen på servern, och alla ändringar i mappen sparades på servern i realtid. Mappen kunde nås genom en ikon på skrivbordet som vi döpte till "Filserver".

#### **4.3.4. Distributionsinnehåll**

Jag och Mikael Torp färdigställde datorerna och valde vilka program som skulle finnas på datorerna. Vi raderade några färdigt installerade program såsom en Bittorent-klient och en IRC-chat klient.

Vi hade tidigare planerat att installera OpenOffice 2.4 eftersom detta torde vara den version som är stabilast samtidigt som den kräver minst prestanda från datorn. Vid denna tidpunkt tillhandahöll dock inte Openoffice.org version 2.4 för nedladdning, utan endast version 3.0. Vi valde att testa 3.0 med motiveringen att det endast rörde sig om 9 stycken datorer, och en eventuell ominstallation till version 2.4 skulle därför inte vara någon tidskrävande process.

Vi installerade även alla s.k. utbildningsprogram som tillhandahålls av Canonical för att i mån av möjlighet få Xubuntu installationen att efterlikna Edbuntu distributionen.

#### **4.3.5. Skolning**

Även Xubuntu är tämligen lättanvänt så önskade personalen skolning i hur de nya datorerna fungerade. Vi presenterade projektet och gick igenom de vanligaste funktionerna och programmen. Den största förändringen var givetvis filhanteringen som skiljer sig en aning i Linux.

Vi hade blivit förvarnade om att vi skulle få många frågor, och vi försökte förbereda oss på vad de kunde tänkas fråga.

Under själva skolningen konstaterade vi dock att alla förstod själva

operativsystemet tämligen bra. Vad de egentligen efterfrågade var en skolning i hur man använder openoffice.

Vi förklarade även för Jan Herrmans hur han kan använda Clonezilla för att återställa en dator som fått något fel. Här ser man än en gång fördelarna med att kлона en fungerande dator och hur man sedan enkelt kan återställa den ifall något går fel. Barn har en tendens att pröva saker som man inte ens kan föreställa sig, och det tog inte många dagar innan startmenyn på en av datorerna försvunnit. Detta var dock inget problem eftersom man med Clonezilla kunde återställa datorn till originalskick på 5 minuter, och ingen information gick förlorad eftersom alla användardokument sparats på servern. Clonezilla kräver inga specialkunskaper, och man kan enkelt följa en steg-för-steg manual som vi inom projektet tagit fram.

#### **4.4. Feedback samt fler datorer**

Den feedback vi fått från Centrumskolan i Oravais har varit mycket positiv. När vi besökte skolan ett par veckor senare frågade vi hur personalen som helhet upplevt operativsystemet, samt om de stött på några komplikationer. Det enda egentliga problem som uppstått var att deras Windows server inte tolkade filnamn med specialtecken såsom ÅÄÖ på ett korrekt sätt från Linux-klienterna. Istället för att felsöka servern kom vi överens om att de i framtiden skulle försöka undvika specialtecken i filnamn.

Centrumskolan hade även uppgraderat RAM-minnet i de äldsta datorerna för att förbättra prestandan. Vi var glada att se att de numera såg datasalen som användbar samt att vi hjälpt dem att ta ett steg närmare lösningar med öppen källkod.

Vi var dock alla överens om att Xubuntu var en alltför nedbantad version av Ubuntu för att användas som undervisningsmedel. Jag kunde genom Vasa yrkeshögskola ordna tio stycken datorer på vilka jag installerade samma Linux-

distribution som jag använt i Vörå-Maxmo. Den enda förändring som gjordes i inställningarna var att jag kopplade bort möjligheten till inloggning via LDAP. De tio nya datorerna ersatte den gamla dataklassen, och de datorer som tidigare använts placerades ut i klassrummen för att användas i vardagsundervisningen. Ubuntu har ett mycket mer avancerat användargränssnitt när det gäller nätverkshantering jämfört med Xubuntu. Vad detta i praktiken betydde var att man i Ubuntu kunde bläddra sig fram till filservern och ansluta till sin egen mapp, och kunde på så sätt undvika proceduren med att köra scriptet. Datorerna som tidigare använts i dataklassen, och som nu hade Xubuntu installerat blev utplacerade i klasserna för att användas som "surfdatorer" och som stöd i den vanliga undervisningen.

Den här typen av konvertering kan jag personligen se att blir ett vanligt fenomen runt om i våra kommuner och skolor, eftersom det finns väldigt många datasalar som är på väg att bli föråldrade men inga pengar finns att få till skolor på den här utbildningsnivån. Jag anser att det även ökar den allmänna kompetensnivån hos individer om man redan från en ung ålder får lära sig att finns alternativ till t.ex. Microsoft eller Apple.

## **5. FAS 3: IMPLEMENTERING OCH ÖVERGÅNG**

### **5.1. Vörå-Maxmo och Eduvom projektet**

Projektet inleddes med ett möte i Vörå den 12 januari 2009.

Jag och mina kollegor Mikael Torp, Daniel Lindblad och Rainer Lytz presenterade OpenKvarken-projektet samt en rad Open source lösningar såsom Skol-linux, Ubuntu och OpenOffice.

Mottagandet var mycket positivt, och intresset var stort för licensfria produkter som kan köras i windows-miljö. Jag presenterade även min idé om att återanvända avlagda datorer från högskolor genom att installera Linux på dem och ge ut dem till grundskolor.

Jag uppfattade att från lågstadiesidan var intresset för skol-linux med dess möjligheter stort. Högstadie och gymnasiet såg den ”fria filserverhanteringen” som ett problem, och möjligheten till användarkonton med egna server-utrymmen efterfrågades.

Jag fick intrycket att man från högstadie och gymnasiesidan hellre ville ha en lösning mer i stil med Ubuntu och Gnome-gränssnittet.

Vi lånade ut 6st Acer Aspire One datorer (5st skol-linux och 1 EdUbuntu).

Vi kom överens om en två veckors utvärderingsperiod varpå beslut om fortsatt samarbete skulle tas.

Vörå-Maxmo var inte intresserade av distributionen Skol-linux, men var mycket intresserade av att delta i själva projektet.

Vörå-Maxmos IT-ansvarige Kenneth Nickull gjorde en rundringning till samtliga skolor och uppskattade behovet till totalt 70 datorer.

## 5.2. Serverinstallation

På servern installerades Debians serverprogramvara.

Jag hade ursprungligen tänkt använda Ubuntu server 8.10, eftersom den var mest bekant för mig personligen. Efter att ha konsulterat Jonny Ström fick jag dock uppfattningen att Debian var ett mycket säkrare val i fråga om stabilitet, och eftersom huvudfokus med projektet var att bygga upp en stabil data-miljö med minimalt underhåll så valde jag att gå på Jonnys linje. Ubuntu är för övrigt baserad på Debian, så skillnaderna i arbetsmiljö var minimala.

I Debian fanns redan ett hanteringssystem för användare och användargrupper, men eftersom Vörå-Maxmo även hade efterfrågat en lokal Moodle installation valde jag en lösning med inloggning via LDAP.

Vi gjorde ursprungligen serverinstallationer på en mindre filserver, men under projektets gång framkom det att man i Vörå-Maxmo hade en tämligen nyinförskaffad Xen server som kunde användas som filserver för de skolor som var kopplade till nätverket inne i Vörå. Denna server skulle således komma att användas som filserver för högstadiet, gymnasiet samt ett par lågstadieskolor. De skolor som inte kunde koppla upp mot Xen servern fick egna små filservrar, och i ett par lågstadieskolor användes datorerna som "stand alone", dvs man använde endast lokala användare och sparade alla filer lokalt på arbetsdatorn.

Jag gjorde med hjälp av Jonny Ström en virtualiserad server vid kontoret på Brändö.

En färdigt konfigurerad LDAP server med Samba filserver fanns att tillgå, men eftersom Vörå-Maxmo under de senaste månaderna övergått till virtualiserade servrar och införskaffat en Xen server så valde vi att köra allting virtuellt för den centrala delen av Vörå. Fördelen med detta var att vi kunde bygga upp en fungerande servermiljö i Vasa, spara ned hela servern på en portabel hårddisk och sedan starta upp den direkt som en virtuell server i Vörå.

### 5.2.1. Vad skiljer en virtuell server från en vanlig?

När man tänker på en traditionell server så föreställer man sig kanske en maskin, med en installation. Denna traditionella server är vanligtvis endast avsedd för att sköta en sak, t.ex. e-mail.

Till följd av att vi de senaste åren fått mycket effektivare processorer har man kunnat ta i bruk så kallade virtuella servrar. Man kan egentligen föreställa sig den nya servermodellen som en vanlig bordsdator där ett centralt operativsystem sköter alla de mindre processerna. De mindre processerna är i detta fall virtuella servrar som utåt sett ser ut som vanliga servrar.

Fördelen med detta är att man enkelt kan flytta en server, utan att någonsin behöva flytta på själva hårdvaran. Ett scenario är att t.ex. hårdvaran på en server blivit föråldrad. Allt man behöver göra för att flytta servern är att spara ned den på t.ex. en portabel hårddisk, kopiera in den till den nya servern, starta upp den nya servern. Det är även en optimal miljö för att testa nya program. Om man vill testa ett nytt e-mail system, men inte riskera att störa det nuvarande systemet så kan man skapa en ny virtuell maskin på servern, och sedan installera den nya programvaran på den nya virtuella servern. Slutresultatet blir att man får två servrar som körs inuti en och samma maskin, båda helt oberoende och ovetandes om varandra.

Den enda begränsningen för hur många maskiner man skapar ligger på mängden RAM-minne man har till förfogande. När man skapar en virtuell maskin så måste man ange hur mycket RAM-minne och hur stor hårddisk som ska användas. Man kan således endast tilldela så mycket resurser som man har hårddisk och RAM-minne. En server med 16GB RAM kan med andra ord ha t.ex. 8st virtuella maskiner med 2GB RAM vardera.

Eftersom Vörå-Maxmo hade en mycket bra server med möjlighet att köra många virtuella servrar bestämde vi oss för att ta steget in mot en komplett arbetsmiljö med open source program som bas.

Den miljö jag planerade upp för Vörå-Maxmo bestod av:

En LDAP server för att sköta inloggningen för alla användare.

En e-mail server för att köra Zimbra.

En apache webbserver för att sköta Moodle och Föräldramötet.

En thin client server som gjorde det möjligt för användare att starta upp t.ex. en laptop via nätverkskortet och på så sätt arbeta i en Linux-miljö utan att någonsin installera något på sin egen dator.

### 5.2.2. Virtuellt server

Det första steget var att skapa en virtuellt server.

Eftersom vår egen server använder 64-bitars arkitektur, och Vörå-Maxmos använder 32-bit var vi tvungen att inkludera `--arch=i386` i kommandot för att på så sätt simulera en 32-bitars miljö.

Precis som på servern i Vörå användes två nätverkskort,

För intern trafik användes `eth0` på ip adressen `192.168.67.216`

För extern och internet trafik användes `eth1` på ip adressen `10.0.0.8`

De båda nätverksanslutningarna bryggades sedan enligt följande `192.168.67.216,mac=00:16:3E:0E:99:DA,'bridge=eth1,ip=10.0.0.8`

```
Logfile produced at:
```

```
    /var/log/xen-tools/ldap.vora.fi.log
```

```
hypervisor0:/etc/xen-tools# xen-create-image --
```

```
hostname ldap.vora.fi
```

```
-ip 192.168.67.216 --arch=i386 -size=200Gb --force
```

```
--role udev
```

General Information



```
-----  
Hostname      : ldap.vora.fi  
Distribution  : lenny  
Partitions    : swap          1024M (swap)  
               /              200Gb (ext3)  
  
Image type    : full  
Memory size   : 256M  
Kernel path   : /boot/vmlinuz-2.6.26-2-xen-amd64  
Initrd path   : /boot/initrd.img-2.6.26-2-xen  
amd64
```

#### Networking Information

```
-----  
IP Address 1 : 192.168.67.216 [MAC:  
00:16:3E:0E:99:DA]  
Netmask      : 255.255.255.0  
Gateway      : 192.168.67.254
```

```
Removing /dev/vm/ldap.vora.fi-swap - since we're  
forcing the install
```

```
Removing /dev/vm/ldap.vora.fi-disk - since we're  
forcing the install
```

```
Creating swap on /dev/vm/ldap.vora.fi-swap
```

```
Done
```

```
Creating ext3 filesystem on /dev/vm/ldap.vora.fi-  
disk
```

```
Done
```

```
Installation method: debootstrap
```

```
Konfigurationsfil :
```

```
# Configuration file for the Xen instance  
ldap.vora.fi, created
```

```
# by xen-tools 3.9 on Tue May 12 14:01:08 2009.
```

```
# Kernel + memory size
```

```
kernel      = '/boot/vmlinuz-2.6.26-2-xen-amd64'
```

```
ramdisk     = '/boot/initrd.img-2.6.26-2-xen-  
amd64'
```

```
memory     = '256'
```

```
# Disk device(s).
```

```
root       = '/dev/sda2 ro'
```

```
disk       = [
```

```
            'phy:/dev/vm/ldap.vora.fi-  
swap,sda1,w',
```

```
            'phy:/dev/vm/ldap.vora.fi-  
disk,sda2,w',
```

```
        ]
```

```
# Hostname
```

```
name          = 'ldap.vora.fi'

# Networking

vif           = [

    'ip=192.168.67.216,mac=00:16:3E:0E:99:DA', 'bridge=
eth1,ip=10.0.0.8,mac=00:16:3e:7a:60:f4'

]

# Behaviour

on_poweroff  = 'destroy'

on_reboot    = 'restart'

on_crash     = 'restart'

kernel       = '/boot/vmlinuz-2.6.26-2-xen-amd64'

och

ramdisk      = '/boot/initrd.img-2.6.26-2-xen-amd64'
```

ändrades senare till att köra 32-bitars kärnan på Vörrå servern.

Denna ”sista minuten” ändring medförde dock att alla de förkonfiguration som gjorts på klient datorerna inte längre gällde. Det var alltså frågan om 63 stycken datorer på vilka man ska göra förändringar i flera olika filer. Två datasalar med totalt 30 datorer på högstadiet konfigurerades var för sig. Efteråt gjorde Kenneth Nickull en klon av en färdigt konfigurerad datorer och använde sedan den klonen för att få de övriga datorerna konfigurerade.

### 5.2.3. LDAP i Vörå

Det andra steget var att installera LDAP på den virtuella servern.

Vad en LDAP server i praktiken gör är att den samlar alla användare och lösenord på en plats och distribuerar den till andra servrar och program. Med hjälp av LDAP servern kan man använda samma användare och lösenord till många olika platser. Man behöver med andra ord inte ha 5 olika inloggningsnamn och lösenord till 5 olika sajter, vilket i ett system med 1000 användare markant minskar underhållet för en systemadministratör.

LDAP servern fick ett grafiskt gränssnitt som kan skötas via en vanlig webbläsare.

Systemet byggdes upp på sådant sätt att en huvudadministratör skapar en grupp, t.ex. lärare. Alla lärare kan sedan skapa egna konton åt sina elever.

### 5.2.4. Datorer utan LDAP

Tre av de skolor som tog del i projektet hade inte möjlighet att ansluta mot Xen servern i Vörå. Rent teoretiskt fanns möjligheten, men det var i praktiken omöjligt att arbeta mot den på grund av långsamma internetanslutningar skolorna emellan.

Eftersom pengar för server-införskaffning saknades fanns för dessa skolor endast två alternativ.

Det första alternativet var att köra datorerna som självständiga maskiner, med andra ord använda lokala användarkonton på datorn utan möjlighet att spara filer mot en server.

Det andra var att installera serverprogramvara på en vanlig klientdator och låta den agera som filserver. Jag och IT-chefen Kenneth Nickull beslutade dock att skolornas användning som sådan inte krävde en filserver. Under projektets gång placerades även två datorer utan LDAP installerad på biblioteket i Vörå. På

biblioteket fanns från tidigare ett par datorer med Windows XP installerat, och vi gjorde detta som ett litet experiment för att se om någon skulle våga röra Linux-datorerna.

### 5.3. Thin & Fat Client lösning

Under tiden vi gjorde installationerna på plats i Vörå började jag och Kenneth Nickull diskutera vad man kunde göra med de datorer som nu användes i Vörå-Maxmo skolor.

De många av datorerna var väldigt gamla, och en egen installation av t.ex. Ubuntu var inte aktuell. Datorerna saknade helt enkelt kapacitet för att köra ett modernt operativsystem på egen hand. Det var här som idén med thin client och fat client väcktes.

Båda begreppen fungerar till stor del på samma sätt, men man det är ändå skäl att skilja dem åt.

Grundidén är att operativsystemet körs på en server mot vilken en dator kopplar upp vid uppstart. Klient-datorn laddar in operativsystemet via nätverkskortet, och allt man gör sparas på servern från vilken man startat upp. Klient datorn behöver i praktiken inte ha någon hårddisk.

Thin-klient är en mycket bra modell för t.ex. bibliotek eftersom man kan ha många okända användare men vill minimera underhållet. Thin klient bygger på att man har en modell-installation som man laddar in varje gång datorn startas upp, och underhållet blir på så sätt minimalt eftersom man i praktiken får en ny installation varje gång man startar upp datorn. Alla eventuella problem som en slutanvändare kan tänkas skapa kan åtgärdas med en omstart.

Det här var dock fråga om en skol-datasal, och som bekant har elever en massa filer som kopieras och flyttas fram och tillbaka via cd-skivor och usb-minnen. Det här ledde till att en lösning enligt fat client-modellen var mer passande. Fat client är relativt lik thin klient, skillnaden är att man kör en del av operativsystemet på själva klientdatorn, och en annan del på servern. Detta minskar belastningen på servern, men ställer även större krav på klient-datorns prestanda. Vi gjorde dock

den bedömningen att klientdatoren hade tillräckligt med prestanda. Att köra en del av operativsystemet på klientdatoren möjliggjorde även användning av t.ex. Dvd-stationer och USB-portar. Med Thin client är detta inte möjligt.[23]

### **5.3.1. LTSP workshop i Vörå**

Den 11-12.10.2010 arrangerade jag en LTSP workshop i Vörå. Till denna workshop hade jag bjudit in samtliga IT-chefer och IT-tekniker i kommunerna runt omkring Vasa. Workshoppens ledare var Elias Aarnio från Tammerfors och Asmo Koskinen från Karleby.

Vi hade över 20 besökare båda dagarna, vilket för mig var en förvånande stor uppslutning med tanke på vilket specialområde LTSP är.

Jag hade införskaffat två stycken servrar med Phenom X6 processorer och 12GB RAM vardera. Resultatet av tillställningen blev en fungerande LTSP miljö i Vörå som tjänar eleverna på skolan.

Detta betyder i praktiken att man kan koppla in vilken dator som helst i Vörå kommuns skolnät och genom att ställa den till PXE boot i BIOSen få en fullt fungerande Ubuntu installation i sin dator. Detta påverkar inte något innehåll eller det operativsystem man har installerat på datorn från tidigare.



Bild 16: Vi byggde en fungerande LTSP miljö under workshoppens gång.



Bild 17: Deltagare under workshoppens första dag. Förutom IT chefer deltog även lärare från bland annat Vasa yrkeshögskola som har planer på att integrera LTSP i IT-utbildningsprogrammen.

## 5.4. Brandvägg i Vörå

Brandväggen som skapades i Vörå är en uppgradering från en tidigare brandvägg. Den bygger på Linux operativsystemet och distributionen Debian Lenny. Den fysiska brandväggen är en vanlig PC (2800mhz + 2GB ram) med mjukvaru raid 1. Brandväggen gjordes i två exemplar. Det finns även en extra hårddisk som kan kopplas in ifall den nuvarande går sönder.

Följande program har installerats och konfigurerats. Alla konfigurationer gjordes av Johnny Ström.

Iptables = sköter grundläggande inställningar för brandväggen dvs. ställer in

INPUT samt FORWARD regler.

Havp = Är en transparent antivirus skanner för web trafik.

Clamav = Som är ett open source anti-virus program havp använder detta program för själva antivirus skanningen.

Bind9 = Används som mellan lagring av dns förfrågningar för att snabba upp dns förfrågningar.

OpenVPN = Open source program för vpn förbindelser.

Dhcp = Dhcp server som delar ut ip adresser dynamiskt till client datorerna.

Konfigurations filer som används i vörå brandvägen är följande:

`/usr/sbin/firewallon` = Brandvägs skript med alla regler.

`/usr/sbin/firewalloff` = Skript som stänger av alla brandväggs regler.

`/etc/init.d/firewall` = Skript som startar eller stoppar brandväggen.



`/etc/havp/havp.config` = Konfigurations fil för havp antivirus scanner.

`/etc/bind/named.conf.options` = Konfigurations file för bind9 dns server.

`/sbin/backup.sh` = skript som kör backup till backup brandväggen.

`/etc/openvpn/server.conf` = Konfigurations file för openvpn.

`/etc/dhcp3/dhcpd.conf` = Konfigurations fil för dhcp servern.

Konfigurationsfilerna finns bifogade som bilagor.

## 5.5. Moodle och Föräldramötet

Fredagen den 16 oktober åkte jag och Mikael Torp till Vörå för att utföra installationer.

Hela projektet hade fördröjts till följd av att Vörå-Maxmos serverutrymme tagit slut, och jag hade tvingats vänta tills nya hårddiskar installerats.

Eftersom Vörå-Maxmo kör den kommersiella Xen-server varianten så kunde jag enkelt skapa nya servrar med Xen Manager. Processen kan liknas vid att skapa virtuella maskiner i VMware eller Virtual PC 2007.

Som grund installerades Debian Etch 4.0 som sedan uppgraderades till 5.0. Jag började med att skapa en vanlig webbserver med Apache2, Mysql och PHP5. På denna server hade jag till först tänkt installera både Moodle och Föräldramötet, eftersom jag trodde att Föräldramötet var en PHP-baserad webbapplikation precis som Moodle. När jag väl fick tag i koden så visade det sig att Föräldramötet var byggd på Java, samt att den skulle köras genom en servlet container. Jag valde att

köra den genom Apache Tomcat eftersom det var den enda som jag var ens det minsta bekant med. Föräldramötet använde även något som hette Open Hierarchy, vilket visade sig vara en java-baserad framework vars uppgift var alla efterlikna en katalogstruktur i stil med den som används i ett operativsystem.

Alla dessa element var tyvärr helt främmande för mig, och för att inte ta riskera att förstöra Moodle installationen valde jag att installera dem på skilda servrar. Moodle i sig är en ganska enkelt installation. Moodle filerna behöver bara laddas in i Apachens www-mapp, och när man sedan besöker servern via en webbläsare så möts man av ett installationsgränssnitt. Efter det är det bara att koppla ihop installationen med SQL servern och databasen, samt skapa ett par användare. Det som var svårt och tidskrävande i detta fall var skapa servern, sätta den till en statisk ip adress samt installera alla beroenden t.ex. Apache och PHP.

## **5.6. Openmeetings**

Det går inte att beskriva Openmeetings som en varken svår eller lätt installation. Openmeetings använder en Red5 server, vilket kort kan beskrivas som en server för att streama ut video. I övrigt så bygger systemet på Java och MySQL. Openmeetings använder även en stor mängd mindre program t.ex. SwfTools, ImageMagic och OpenOffice. Det finns även vissa funktioner i Openmeetings som bygger på andra program. Bland annat kalender-funktionen i Openmeetings är byggd på Openlazzlo, vilket givetvis betyder att Openlazzlo måste installeras.

Om man vet hur detta skall göras, och om man har en saklig guide så går det överlag mycket smidigt. Detta förutsätter att man vet vad man håller på med.

Ett problem man kan stöta på i ett senare skede är att Openmeetings använder port 5080 och 1935, vilka kan vara stängda hos ett företag eller en skola. Orsaken till detta är att de vanligtvis inte används, och för att minimera eventuella säkerhetshål så stänger administratören ofta alla portar som inte behövs. För att vara 100% säker på att man får Openmeetings att fungera måste man alltså veta vem som administrerar ens brandvägg.

Jag kommer här att gå igenom en Openmeetings installation för att ge en

överblick av processen.[24]

Vad behövs?

- MySQL
- xvfb
- OpenOffice
- ImageMagic
- Ghostscript
- SwfTools
- Java
- Red5
- Ppenlazslo
- OpenMeetings

Vi inleder med att uppdatera och installera aptitude eftersom vi kommer att använda aptitudes installationspaket

```
$su  
#apt-get update  
#apt-get install aptitude
```

Efteråt skapar vi en mapp där vi lagrar alla våra filer

```
#su  
#mkdir /root/tmpOpenMeetings/  
#cd /root/tmpOpenMeetings/
```

Openmeetings kan användas med en mängd olika databas-system, men jag väljer MySQL som är en av de vanligaste (och enligt mig den bästa).

```
aptitude install mysql-server
```

Det är bra att lägga till följande i /etc/mysql/my.conf för att få rätt teckenstöd redan från början

```
[mysqld]
default-character-set=utf8
character-set-server=utf8
```

För att få stöd för presentationer och dokument i Openmeetings så installerar jag xvfb och OpenOffice.org

```
#aptitude install xvfb openoffice.org
```

Efteråt ska vi editera Setup.xcu:

```
#vim
/usr/lib/openoffice/share/registry/data/org/openoffice/Setup
.xcu
```

Följande ska klistras in under: <node oor:name="Office">: för att styra trafiken

```
<propoor:name="ooSetupConnectionURL" or:type="xs:string">
<value>socket,host=localhost,port=8100;urp</value>
</prop>
```

ImageMagick är ett program som vi kommer att använda för att manipulera bilder som laddas upp och används på whiteboarden.

Ghostscript används för att tolka PDF dokument.

För att installera dessa skriver vi:

```
#aptitude install imagemagick gs-gpl
```

Alla dokument som laddas upp till whiteboarden och openmeetings konverteras till flash. Detta sker via SWF Tools

```
#aptitude install swftools
```

If you get this message: "Couldn't find any package whose name or description matched "swftools" " please download swftools directly from web with running:

Av någon orsak fick jag felmeddelandet: "Couldn't find any package whose name or description matched "swftools"". Jag laddar ned den till server från webbplatsen genom att använda kommandot wget.

```
#wget
http://ftp.it.debian.org/debian/pool/main/s/swftools/swftool
s_0.8.1-1_i386.deb
```

Programmet klagar på att "libart-2.0-2 " saknas, så jag installerar det innan jag installerar SWFTools.

```
#aptitude install libart-2.0-2
```

Nu kan jag installera SWF Tools med kommandot:

```
#dpkg -i swftools_0.8.1-1_i386.deb
```

Red5 är en Open source flash server som Openmeetings baserar sig på. Som namnet kanske avslöjar så har den som uppgift att streama ut flash.

Innan jag installerar Red5 väljer jag dock att installera Java. Jag vet inte om ordningsföljden spelar någon roll, men erfarenheten säger mig att alltid installera Java först.

```
#aptitude install sun-java5-bin
#aptitude install sun-java5-demo
```

Med följande kommando laddar jag ned Red5 och installerar det.

```
#wget http://dl.fancycode.com/red5/0.6.3/debian/red5_0.6.3-
1_all.deb
#dpkg -i red5_0.6.3-1_all.deb
```

OpenLaszlo är en plattform som ger tillgång till applikationer inuti Openmeetings. Den används främst till utveckling och är inte nödvändig för en vanlig användare. I skrivande stund är det dock tal om att den nya kalenderfunktionen ska använda OpenLazslo, så jag väljer att installera den i.a.f.

Jag laddar ned den från deras hemsida och installerar.

```
#wget http://download.openlaszlo.org/4.0.6/openlaszlo-4.0.6-
unix.tar.gz
#tar -xvf openlaszlo-4.0.6-unix.tar.gz
#mv lps-4.0.6/ /opt/
```

Vi har nu den grund som behövs för att installera OpenMeetings.

```
#wget http://openmeetings.googlecode.com/files/red5-
openmeetings-rc5.zip

#unzip red5-openmeetings-rc5.zip
```

Jag anger var MySQL databasen finns genom att gå in i filerna.

```
#cp openmeetings/conf/mysql_hibernate.cfg.xml
openmeetings/conf/hibernate.cfg.xml

#vim openmeetings/conf/hibernate.cfg.xml
```

Jag editerar följande :

```
<property name="connection.username"><DB_USER></property>

<property
name="connection.password"><DB_PASSWORD></property>

<property
name="connection.url">jdbc:mysql://<DB_HOST>/<DB_NAME></prop
erty>
```

Resultatet i min configurationsfil blir:

```
<!-- User / Password -->

<property name="connection.username">root</property>

<property
name="connection.password">dj76shwrsh16xsvak1</property>

<!-- Database Settings -->

<property
name="connection.driver_class">com.mysql.jdbc.Driver</proper
ty>

<property
name="dialect">org.hibernate.dialect.MySQLInnoDBDialect</pro
perty>

<property
name="connection.url">jdbc:mysql://localhost/openmeetings?us
eUnicode=true&createDatabaseIfNotExist=true&character
Encoding=utf-8</property>
```

Ett par småkonfigurationer. Jag uppdaterar t.ex. mina språkfiler:

```
#cp
openMeetingsFromSVN/xmlcrm/java/webapp/openmeetings/language
s/* openmeetings/languages/.
```

Efteråt placerar jag openmeetings i red5 webbapplikationsmappen. Detta gör att Red5 kommer att hitta min openmeetingsinstalltion och köra den när Red5 servern startas.

```
#cp -r openmeetings/ /usr/lib/red5/webapps/
```

Nu är det dags att starta alltihop. Jag startar först OpenOffice.org och OpenLaszlo. Efteråt startar jag OpenMeetings (Red5 servern).

```
#xvfb-run --server-args='-screen 0 800x600x16' -a
/usr/lib/openoffice/program/soffice -headless -nologo -
norestore &

#export JAVA_HOME=/usr/lib/jvm/java-1.5.0-sun

#/opt/lps-4.0.6/Server/tomcat-5.0.24/bin/startup.sh

#/etc/init.d/red5 start
```

De sista installationsstegen görs i webbläsaren. Besök adressen med t.ex. Mozilla Firefox. Du måste ha flash installerat på din webbläsare för att OpenMeetings ska fungera.

```
http://$DIN_IP:5080/openmeetings/Install
```

Utfört från serverdatorn är adressen: <http://127.0.0.1:5080/openmeetings/Install>

Efter att de nödvändiga uppgifterna angetts och systemet konfigurerats klart kan Openmeetings nås på adressen <http://dinserveripadress:5080/openmeetings>. Porten 5080 måste skrivas in.

## 6. AVSLUTNING

Syftet med detta lärdomsprov var att påvisa att man kan bygga upp en fungerande IT miljö med endast avgiftsfria program. Jag visste själv inom mig att detta högst antagligen var möjligt, jag har ju själv arbetat med datorer de senaste 15 åren och jag vet hur utvecklingen av Linux-skrivbordet gått framåt de senaste åren. Fortfarande lever dock myten om att Linux inte är något för en vanlig användare kvar, vilket är mycket tråkigt. Linux må vara lite knepigare att konfigurera, men för slutanvändaren är det lika enkelt eller kanske t.o.m. enklare än Windows att använda.

Arbetet har enligt min uppfattning förlöpt väldigt bra, även om det blivit långt mer omfattande än vad som först var tänkt. Det är även väldigt lätt att glömma bort kringutrustning såsom t.ex. nätverkskrivare. Dessa kan inbland vara problematiska att få att fungera i linux-miljö. De största utmaningarna för mig har varit att få alla serverinstallationer integrerade med LDAP samt att lista ut hur man autenticerade Linux klienter mot LDAP. Jag hittade ett 50-tal olika guider för att göra detta, men ingen av dem fungerade. Jag fick pröva mig fram på egen hand, och i processen skrev jag en egen guide för hur man gör autenticeringen. Guiden hittas bland bilagorna.

Jag hoppas att detta lärdomsprov skall kunna komma att fungera som en guide för skolor och kommuner som funderar på att övergå till Linux och programlösningar med öppen källkod. Det är ett stort steg om man väljer ett migrera hela systemet på en gång, men den investerade tiden återbetalas mångfalt under de kommande åren genom inbesparade licenser, underlättat underhåll och framför allt frihet.



## KÄLLFÖRTECKNING

[1] Wikipedia.org, 2009. GNU General Public License [online]. Uppdaterad i November 2009 [hänvisning 3.10.2009]. Tillgänglig i form av www-dokument: <URL: [http://en.wikipedia.org/wiki/GNU\\_General\\_Public\\_License](http://en.wikipedia.org/wiki/GNU_General_Public_License)>.

[2] GNU General Public License v1.0 - GNU Project - Free Software Foundation (FSF), 2009 [online]. Uppdaterad i Februari 1989 [hänvisning 9.2.2010]. Tillgänglig i form av www-dokument: <URL: <http://www.gnu.org/licenses/old-licenses/gpl-1.0.html>>.

[3] GNU General Public License v2.0 - GNU Project - Free Software Foundation (FSF), 2009 [online]. Uppdaterad i Juni 1991 [hänvisning 9.2.2010]. Tillgänglig i form av www-dokument: <URL: <http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>>.

[4] A Quick Guide to GPLv3 - GNU Project - Free Software Foundation (FSF), 2009 [online]. Uppdaterad i Juni 2007 [hänvisning 9.2.2010]. Tillgänglig i form av www-dokument: <URL: <http://www.gnu.org/licenses/quick-guide-gplv3.html>>.

[5] GNU Lesser General Public License v2.1 - GNU Project - Free Software Foundation (FSF), 2009 [online]. Uppdaterad i Februari 1999 [hänvisning 6.3.2010]. Tillgänglig i form av www-dokument: <URL: <http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html>>.

[6] GNU Free Documentation License - GNU Project - Free Software Foundation (FSF), 2009 [online]. Uppdaterad i November 2002 [hänvisning 6.3.2010]. Tillgänglig i form av www-dokument: <URL: <http://www.gnu.org/licenses/old-licenses/fdl-1.2.html>>.

[7] GNU Affero General Public License - GNU Project - Free Software Foundation (FSF), 2009 [online]. Uppdaterad i November 2007 [hänvisning

6.3.2010]. Tillgänglig i form av www-dokument: <URL: <http://www.gnu.org/licenses/agpl.html>>.

[8] Black Duck Open Source Resource Center, 2009. GPLv3, LGPLv3 and AGPLv3 Adoption Rates [online]. Uppdaterad i september 2009 [hänvisning 21.9.2009]. Tillgänglig i form av www-dokument: <URL:<http://www.blackducksoftware.com/oss/licenses>>.

[9] Alan Hawrylyshen, 2009. Diagram showing [GNU General Public License](#) version 3 compatibility [online]. Uppdaterad i juli 2008 [hänvisning 15.7.2009]. Tillgänglig i form av www-dokument: <URL:<http://en.wikipedia.org/wiki/File:Quick-guide-gplv3-compatibility.svg>>.

[10] Moodle.org, 2009. Moodle.org: open-source community-based tools for learning [online]. Uppdaterad i mars 2009 [hänvisning 3.3.2009]. Tillgänglig i form av www-dokument: <URL: <http://moodle.org>>.

[11] Zimbra, the leader in open source email and collaboration, 2009. Inbox - View Messages as Conversation Threads or Individually, With Message Text Preview [online]. Uppdaterad i juni 2009 [hänvisning 9.7.2009]. Tillgänglig i form av www-dokument: <URL: [http://www.zimbra.com/products/ss\\_inbox.html](http://www.zimbra.com/products/ss_inbox.html)>.

[12] Zimbra, the leader in open source email and collaboration, 2009. Calendar - See All of Your Scheduled Appointments and Their Status; Mouseover for Full Details [online]. Uppdaterad i juni 2009 [hänvisning 9.7.2009]. Tillgänglig i form av www-dokument: <URL: [http://www.zimbra.com/products/ss\\_calendar.html](http://www.zimbra.com/products/ss_calendar.html)>.

[13] Föräldramötet, 2009. Kommunikation mellan hem och förskola [online]. Uppdaterad i mars 2009 [hänvisning 19.3.2009]. Tillgänglig i form av www-dokument: <URL: <http://foraldramotet.sundsvall.se/>>.

[14] Openmeetings, 2009. Open-Source Web-Conferencing [online]. Uppdaterad i november 2009 [hänvisning 14.11.2009]. Tillgänglig i form av www-dokument: <URL: <http://code.google.com/p/openmeetings/>>.

[15] LDAP, 2009. Lightweight Directory Access Protocol [online]. Uppdaterad i juli 2009 [hänvisning 17.7.2009]. Tillgänglig i form av www-dokument: <URL:[http://en.wikipedia.org/wiki/Lightweight\\_Directory\\_Access\\_Protocol/](http://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol/)>.

[16] HowtoForge, 2009. Installing ubuntu-system-panel (USP) On Ubuntu 8.10 [online]. Uppdaterad i november 2008 [hänvisning 23.04.2009]. Tillgänglig i form av www-dokument: <URL:<http://www.howtoforge.com/installing-ubuntu-system-panel-on-ubuntu-8.10/>>.

[17] Ubuntu Suomi, 2009. **Edubuntu - Ubuntu opetukseen** [online]. Uppdaterad i november 2008 [hänvisning 23.04.2009]. Tillgänglig i form av www-dokument: <URL:<http://wiki.ubuntu-fi.org/Edubuntu/>>.

[18] Català: Captura de pantalla de Xubuntu 8.10 amb entorn d'escriptori XFCE amb català, 2009. **Rbuj, Wikipedia** [online]. Uppdaterad i oktober 2009 [hänvisning 3.11.2009]. Tillgänglig i form av www-dokument: <URL:<http://commons.wikimedia.org/wiki/File:Xubuntu.8.10.XFCE.cat.png/>>.

[19] KDE – KDE 4.2, 2009. KDE 4.2 Beta 1 Release Announcement [online]. Uppdaterad i november 2008 [hänvisning 23.07.2009]. Tillgänglig i form av www-dokument: <URL:<http://www.kde.org/announcements/announce-4.2-beta1.php/>>.

[20] KDE – KDE 4.3.2, 2009. KDE 4.3.2 Stabilizes Free Desktop [online]. Uppdaterad i oktober 2009 [hänvisning 23.11.2009]. Tillgänglig i form av www-dokument: <URL: <http://www.kde.org/announcements/announce-4.3.2.php>>.

[21] Skol-Linux, 2009. Linux i skolan [online]. Uppdaterad i mars 2009

[hänvisning 2.4.2009]. Tillgänglig i form av www-dokument:  
<URL:<http://81.209.9.136/skolor/skol-linux/svenska/main-swe.htm>>.

[22] Skol-Linux, 2009. Skärmdumpar [online]. Uppdaterad i mars 2009  
[hänvisning 2.4.2009]. Tillgänglig i form av www-dokument:  
<URL:<http://81.209.9.136/skolor/skol-linux/svenska/screens-swe.htm>>.

[23] LTSP, 2009. The Linux Terminal Server Project [online]. Uppdaterad i juli  
2009 [hänvisning 3.7.2009]. Tillgänglig i form av www-dokument:  
<URL:<http://www.ltsp.org/>>.

[24] InstallationDebian – openmeetings, 2009. Installing !Openmeetings on  
debian etch [online]. Uppdaterad i oktober 2009 [hänvisning 17.10.2009].  
Tillgänglig i form av www-dokument:  
<URL:<http://code.google.com/p/openmeetings/wiki/InstallationDebian/>>.

## **BILAGOR**

### **Bilaga 1: Utvärdering av Mini-laptops på Normen skola**

Vi var lyckliga och glad att få 13 små datorer till klassen. 17 elever finns i 4B och i klassrummet finns från tidigare 4 bordsdatorer av äldre datum. I skolans datasal finns 8 datorer.

De 13 datorerna fungerade super i klassrummet och korridorer där eleverna arbetade. Här borde foto av elever som ligger på golvet, sitter i soffor och vid olika bord finnas med.

Eleverna arbetade kring ett projekt i Natur och miljö och skulle skriva faktatexter kring de nordiska länderna, söka fakta och bilder och sammanställa till ett dokument.

Tyvärr saknar vår skola ett trådlöst nätverk, så vi fick nöja oss med de få Internetanslutningar som finns tillgängliga. Dessa elever är tio år, så jag tycker att de jobbade bra.

För att testa hur eleverna skulle arbeta om de fick ha en egen dator i ryggsäcken, kom jag överens med några föräldrar som för och hämtar sina barn till skolan att de får ta datorn med hem, på förälders ansvar. Detta fungerade utmärkt. Det är nog målet att varje elev skulle kunna ha med i ryggsäcken i en vadderad påse en egen mini-laptop.

Under de veckor vi arbetade på datorerna skapade varje elev c 7 sidor faktatext.

Vi hade inte andra problem än att batteriet laddat ur och eleverna glömt att spara. När vi skulle spara till USB-minne så tog vi ut stickan för snabbt att vi förlorade allt arbete som några elever gjort. Tyvärr hade jag datorerna på rad, såg att mappen skapats, men kollade inte att den var tom. Då blev det mycket jobb för läraren att från handskrift föra över allt till maskinskrift. Det var dyrköpt erfarenhet av Linux för mig.

Många tack att vi fick vara med i Open Kvarken-projektet.

I vår budget kommer vi att an hålla om Mini-laptops i klassuppsättning. De är lätta att transportera och hantera och fungerar utmärkt också för barn.

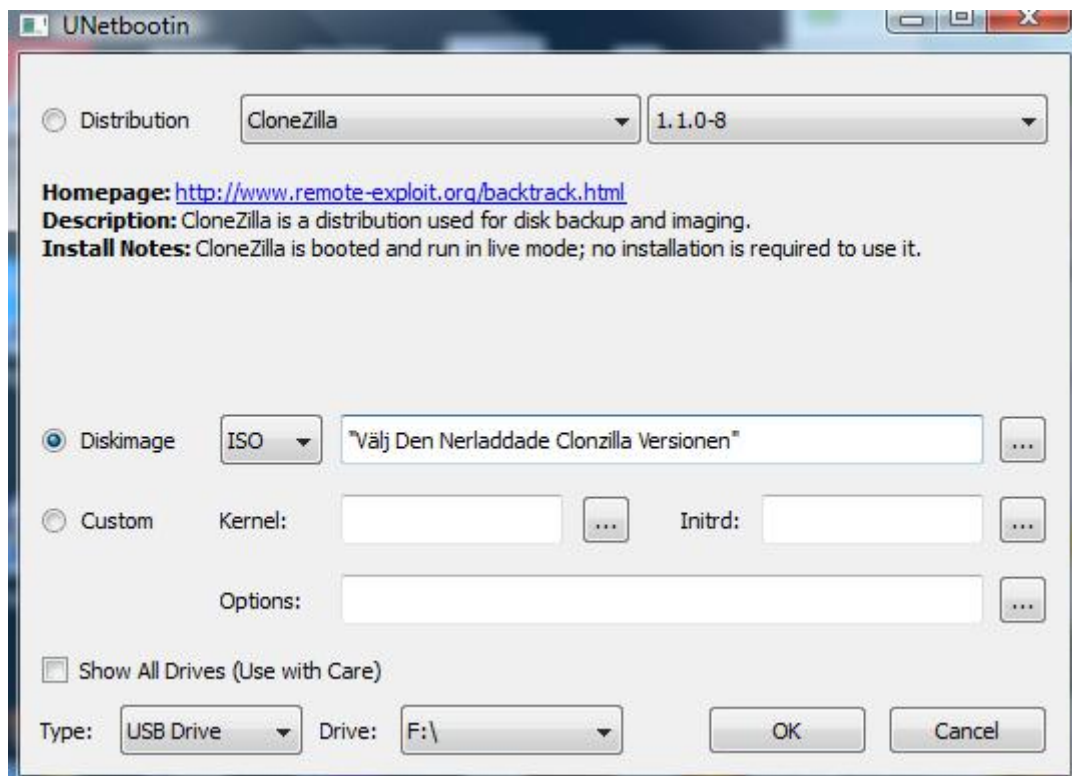
Maria Sjöblom

## Bilaga 2. Clonezilla – Kloning av operativsystem

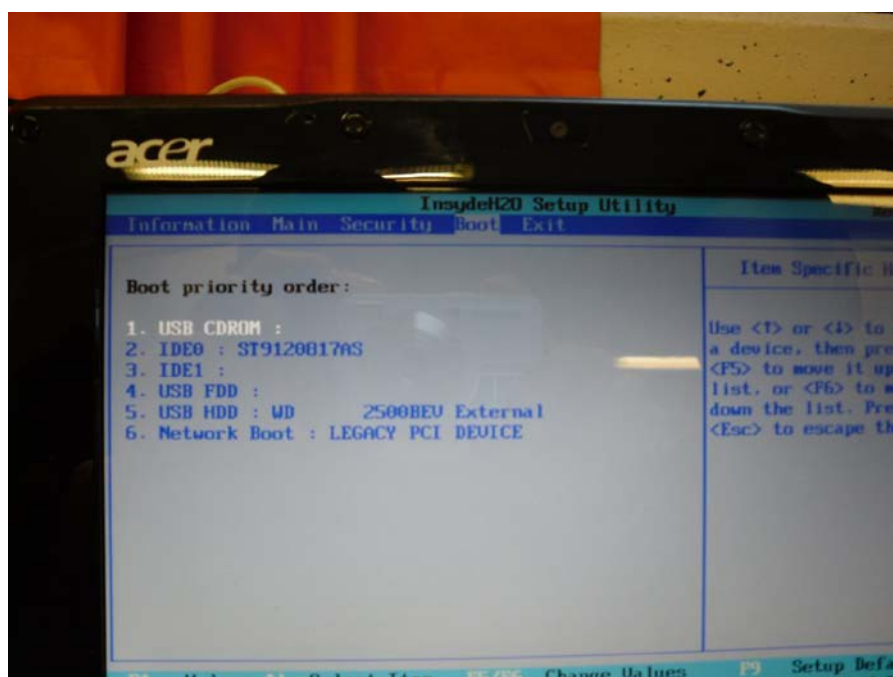
Clonezilla är ett backup-program med öppen källkod. Det kan användas för såväl backups som återställningar av hårddiskor, partitioner, system och så vidare.



För att klona ett befintligt operativsystem så behöver vi Clonezilla live bränt som en image på en cd-skiva eller på en usb-sticka. Om vi vill använda en cd-skiva så måste vi använda en extern cd-rom eftersom Aspire One saknar cd-läsare.

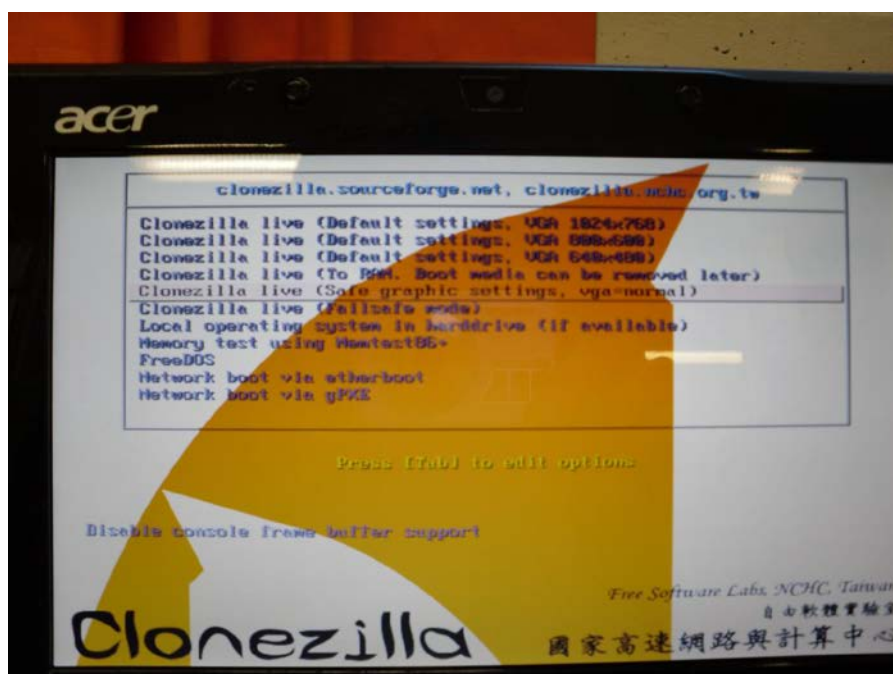


För att få Clonezilla bootbart från en usb-sticka måste vi först ladda ner programmet **Unetbootin** och **Clonezilla Usb-versionen(ISO-fil)**. Som **Distribution** väljer vi Clonezilla version 1.1.0-8. Efter det väljer vi en ISO **Diskimage** och sökvägen till vår Clonezilla version som blivit nerladdat(ISO format) Sen är det bara att klicka på **OK**.

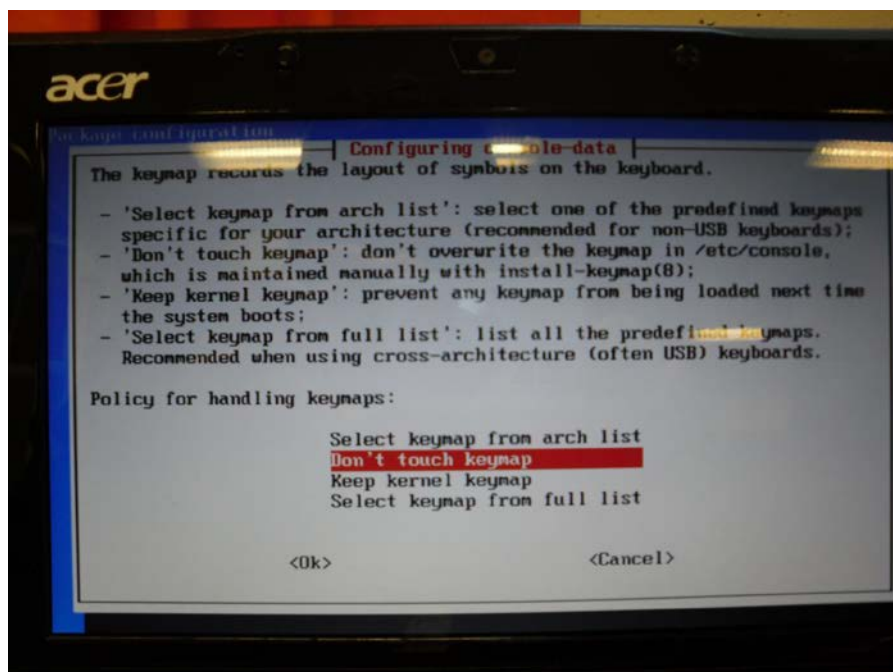




Eftersom vi vill ha i gång Clonezilla live och inte hårddisken med det befintliga operativsystemet så krävs det att vi går in i BIOSen och ändrar prioriteringen på de enheter som skall bootas. För att göra detta startar vi upp datorn och trycker på **F2** tills vi kommer in i BIOSen. Väl i BIOSen navigerar vi till **Boot** och därefter väljer vi som första alternativ i Boot Priority Order det lagringsmedium där Clonezilla live finns installerat. Efter det går vi till **Exit** och väljer att spara ändringarna.



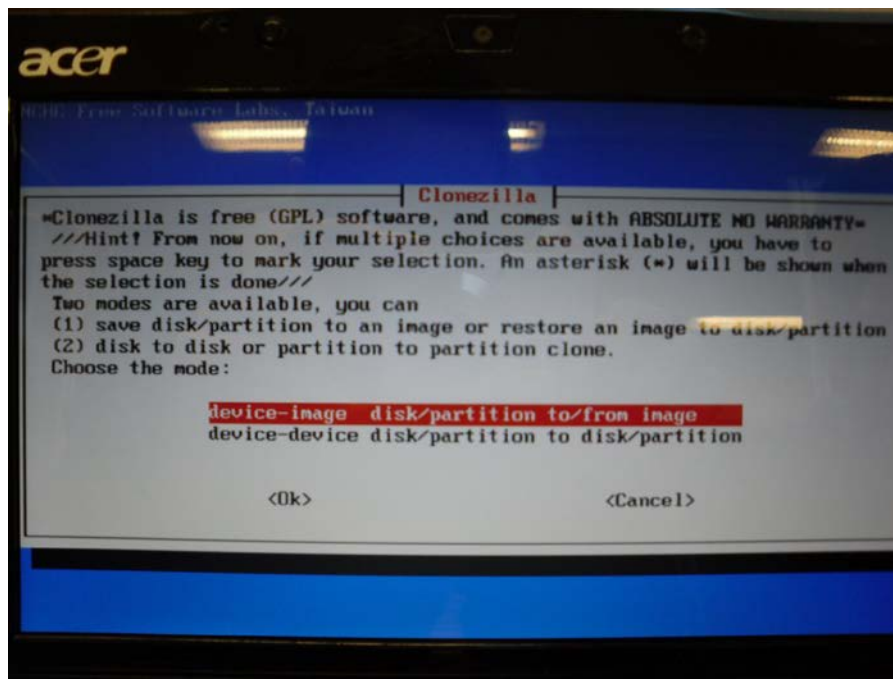
När vi nu startar om datorn startas Clonezilla automatiskt. Vi väljer därefter alternativet Clonezilla live (**Safe graphic settings, vga=normal**).



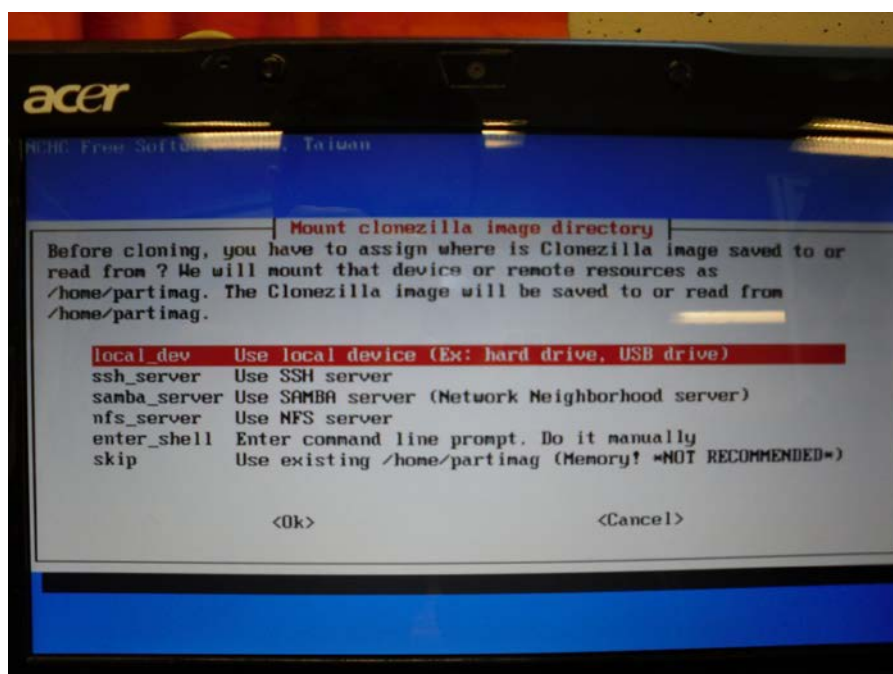
I nästa steg väljer vi **Don't touch keymap**.



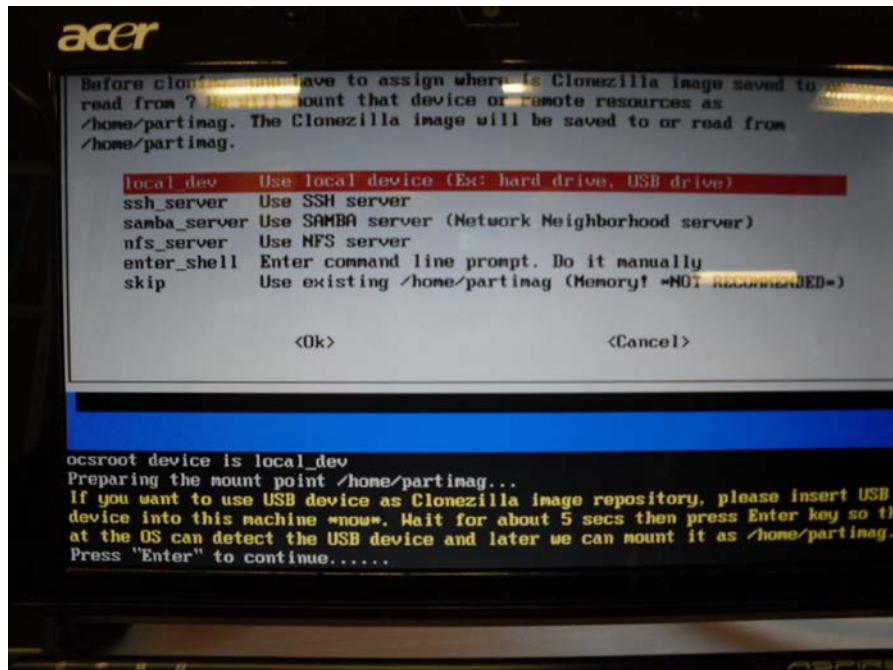
Sen väljer vi **Start Clonezilla**.



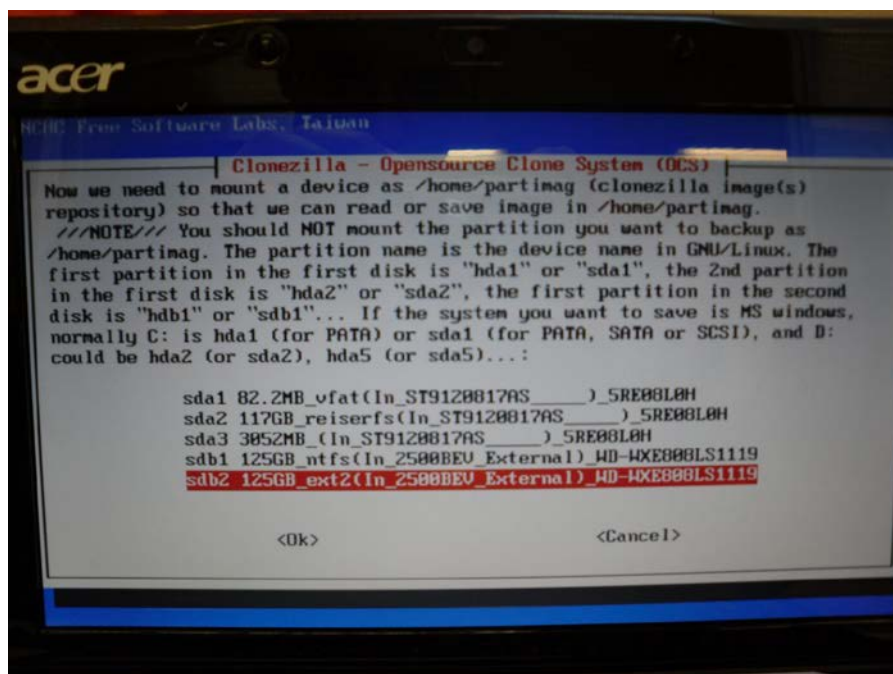
Nu får vi två alternativ att välja mellan. Spara/återställa en disk eller partition till en image, samt skapa en disk till disk eller partition till partition klon. Här väljer vi **device-image** eftersom vi vill skapa en image av vårt operativsystem.



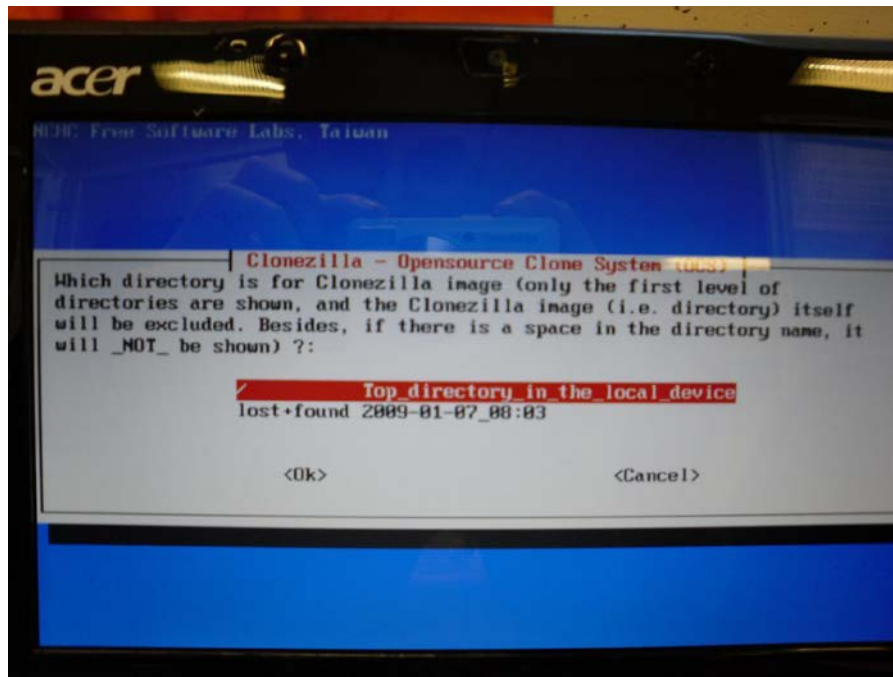
Efter det får vi välja var vi vill spara imagen. Eftersom vi vill spara den till en extern HDD väljer vi **local-dev**.



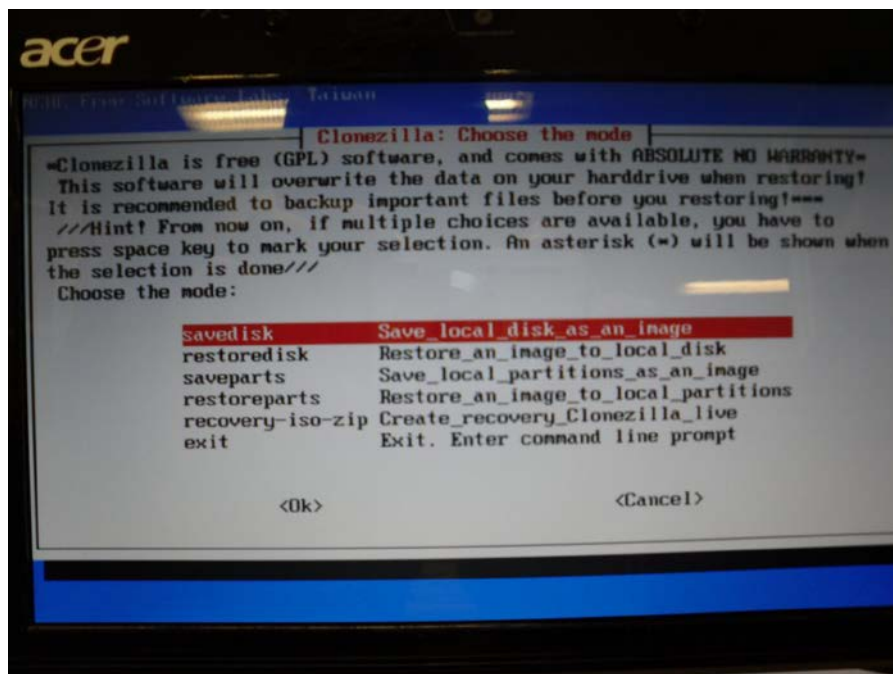
Vänta fem sekunder och tryck därefter **Enter**.



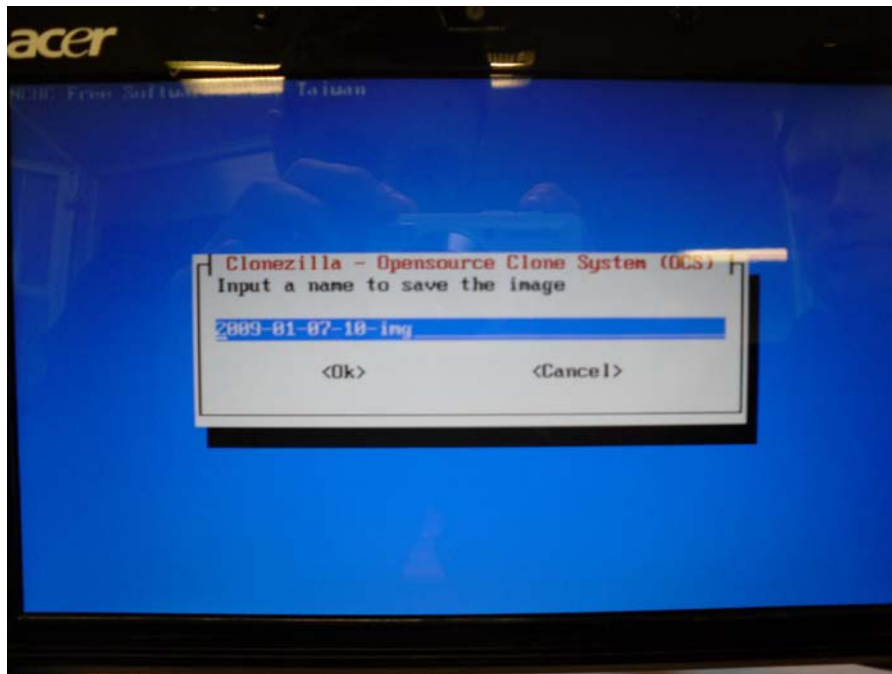
Nu får vi välja till vilken hårddiska och partition vi vill spara imagen. Eftersom vi vill spara den på andra partitionen på vår externa HDD väljer vi **sdb2**.



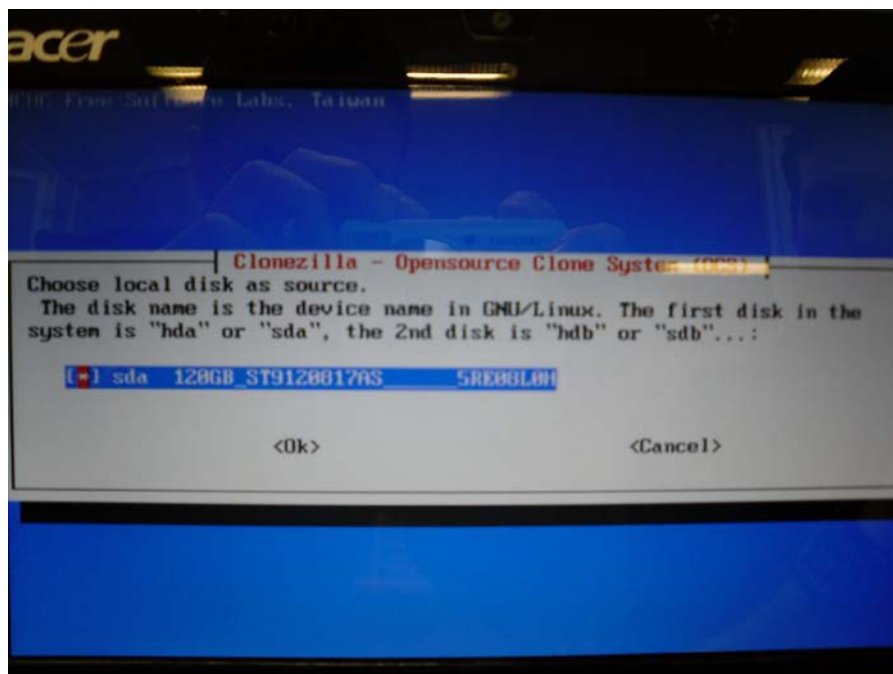
Vi vill inte spara imagen i någon mapp, utan i HDD:ns högsta katalog. Tryck **Enter**.



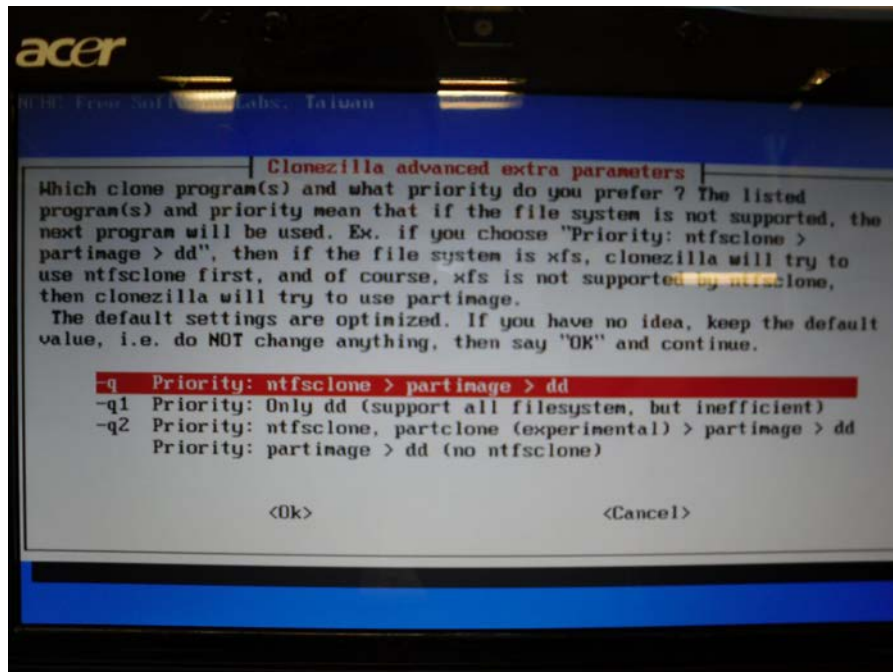
Nu får vi välja om vi vill spara vår lokala hårddisk/partition som en image eller om vi vill återställa en hårddisk/partition från en image. Vi vill spara, därför väljer vi **savedisk**.



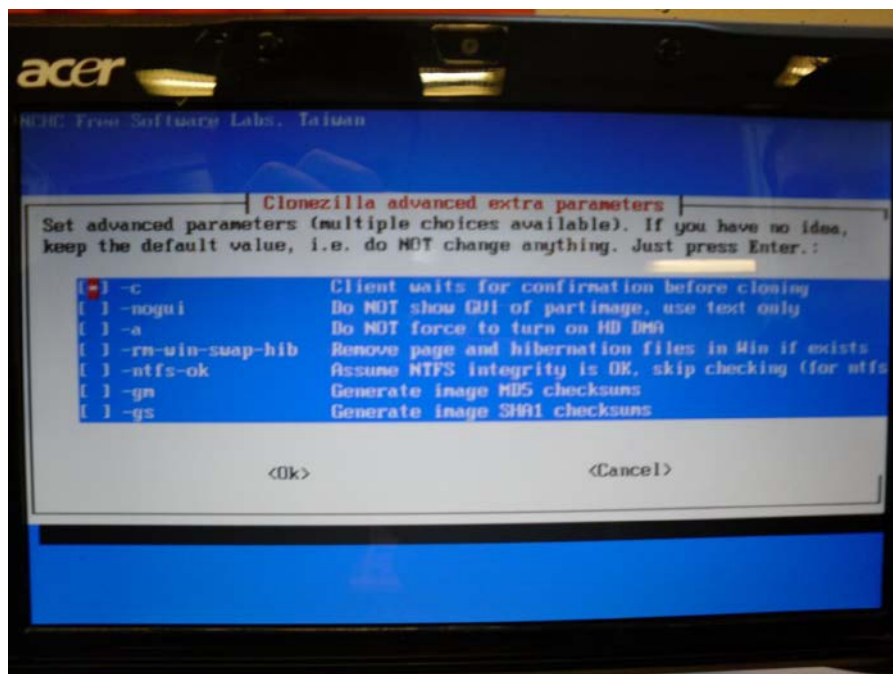
Därefter får vi välja namnet på image vi ska skapa. Välj ett valfritt namn eller tryck **Enter**.



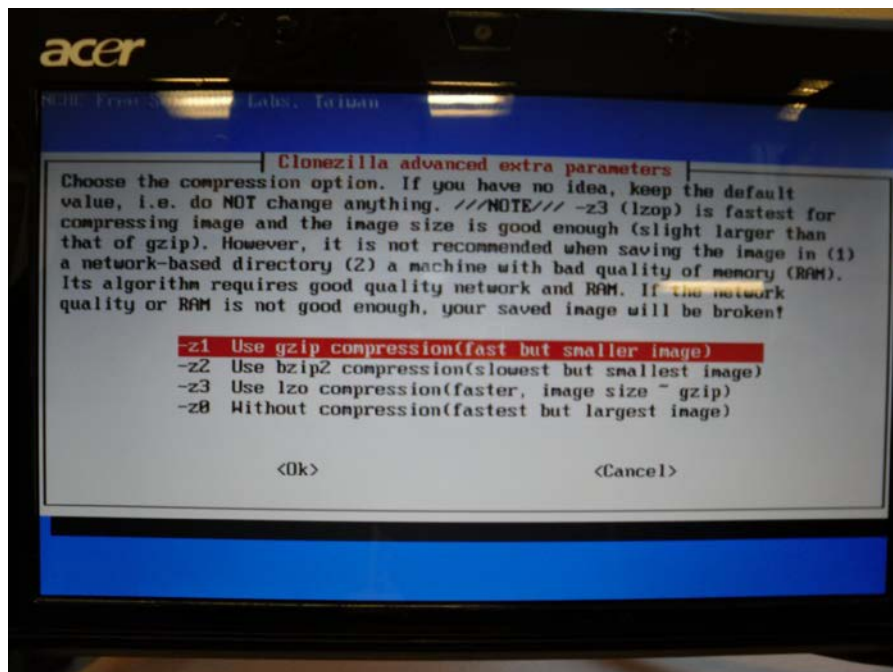
Nu får vi välja vilken hårddisk vi vill skapa en image av. Eftersom vi i det här fallet bara har en hårddisk har vi bara ett alternativ att välja. Tryck **Enter**.



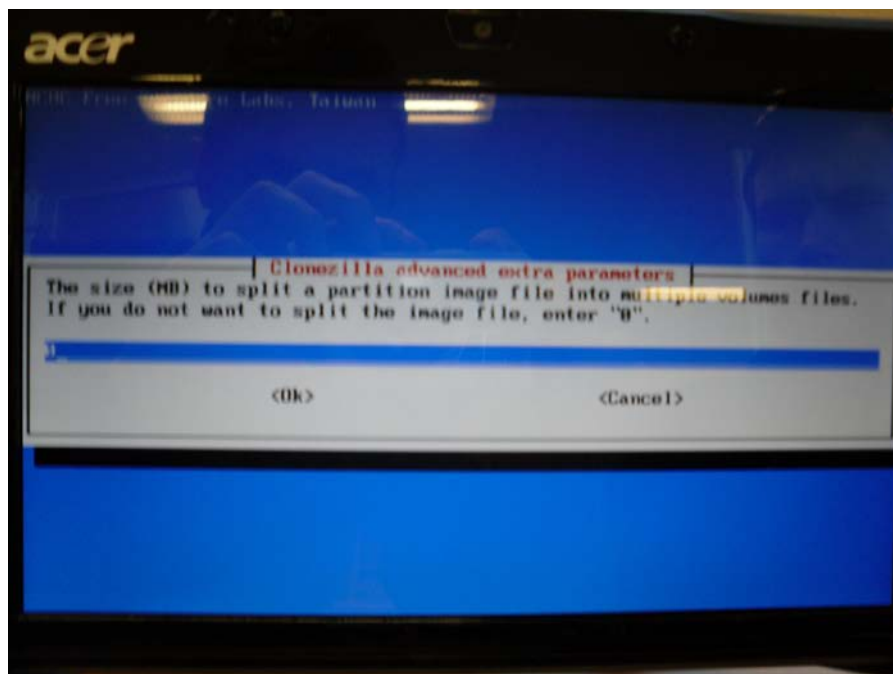
Sen får vi välja vilka klonprogram och vilken prioritering vi föredrar. Vi väljer det första alternativet. Tryck **Enter**.



Tryck **Enter**.



Efter det får vi välja hur vi vill komprimera vår image. Vi vill använda oss av gzip. Tryck **Enter**.

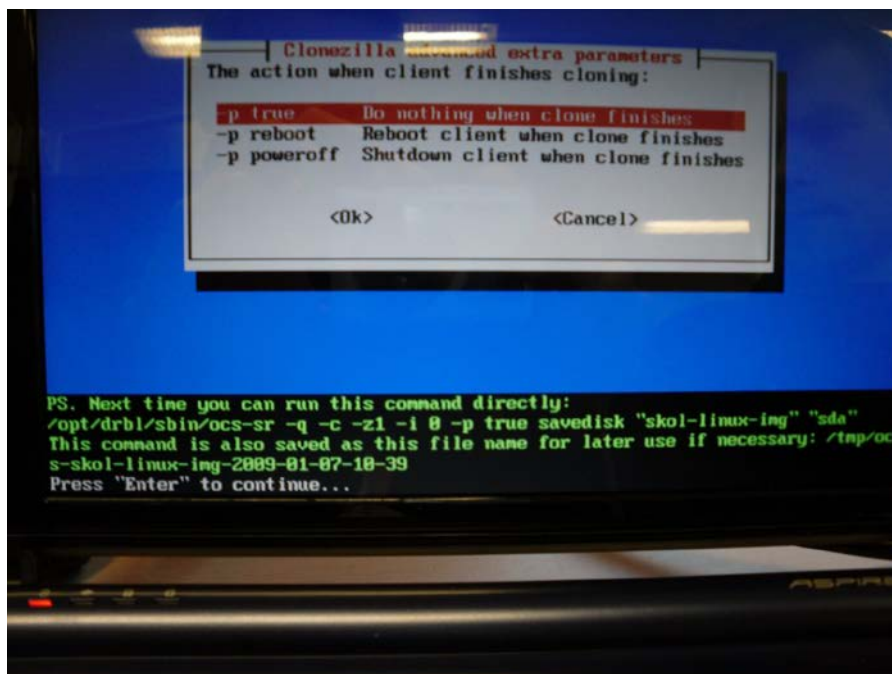


Sen får vi välja om vi vill dela upp imagen och storleken på dessa delar. Vi vill inte dela upp vår image och därför skriver vi **0**.





Här väljer vi huruvida datorn ska startas om, stängs av eller förbli i samma läge när kloningen är klar. Vi väljer **-p true**.



Tryck **Enter**.



Efter det här skedet påbörjas kloningen. Vi måste bekräfta två gånger att vi verkligen vill skapa imagen. Tryck **Y** och sedan **Enter**.

```
Use gzip to compress the image.
Image file will be split with size limit 2000 MB.
*****
Volume size: 0 bytes (0 MiB)
partimage: status: initializing the operation.
partimage: status: Partition: 0.6.1
partimage: status: Image type: NONE
partimage: status: Saving partition to the image file...
partimage: status: reading partition properties
partimage: status: writing header
partimage: status: copying used data blocks
File Name      Size      T:Elapsed/Estimated  Rate/min      Progress
stdout         S: 32M    T:00:00:05/00:00:00  R: 387M/min   P:100%

partimage: status: committing buffer cache to disk.
>>> Time elapsed: 5.02 secs (~.083 mins)
*****
Finished saving /dev/etch/home as /home/partinag/2008-09-29-08-ing/etch-home.XXX
*****
*****
*****
This program is not started by Clonezilla server, so skip notifying it the job is done.
Finished!
Now syncing - flush filesystem buffers...

*****
If you want to use clonezilla again:
(1) Stay in this console (console 1), enter command line prompt
(2) Run command "exit" or "logout"
*****
When everything is done, remember to use 'poweroff', 'reboot' or follow the menu to do a normal poweroff/reboot procedure. Otherwise if the boot media you are using is a writable device (such as USB flash drive), and it's mounted, poweroff/reboot in abnormal procedure might make it FAIL to boot next time!
*****
Press "Enter" to continue...
```

Kloningen är klar! Tryck **Enter**.

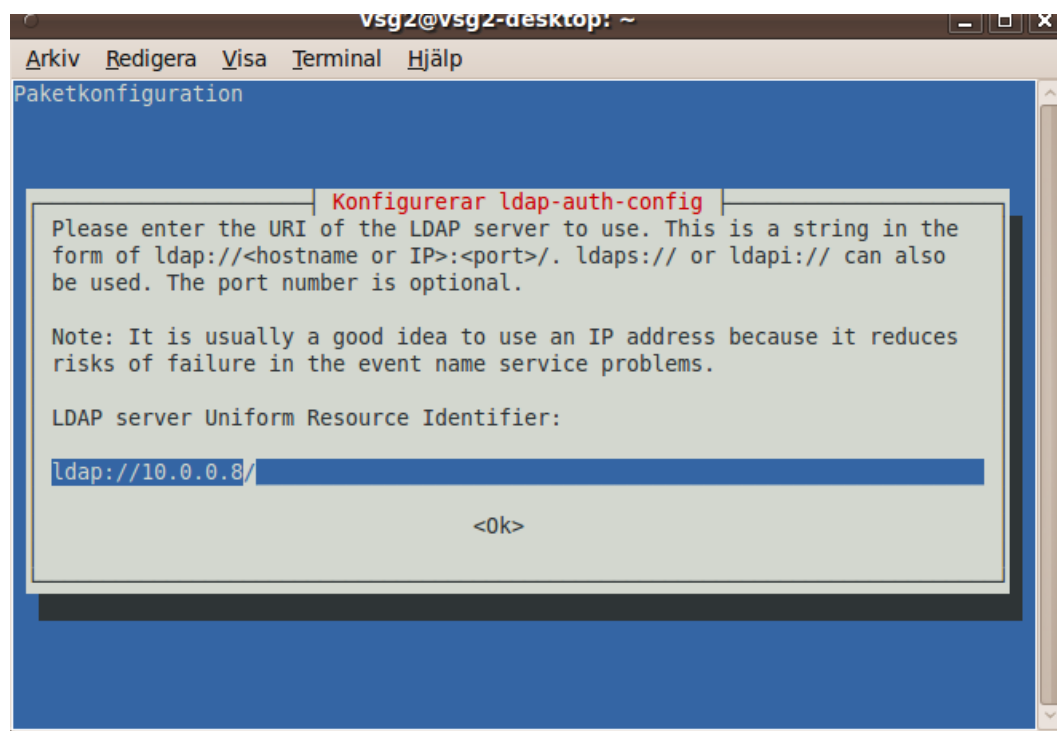
### Bilaga 3. Guide för att autentisera Linux klienter mot LDAP

1) sudo apt-get install auth-client-config

ange sudo lösenord

2) System/ Administration/ Software sources. Aktivera alla källor (även other sources)

3) sudo apt-get install libpam-ldap libnss-ldap nss-updatedb libnss-db



4) ldap://10.0.0.8/

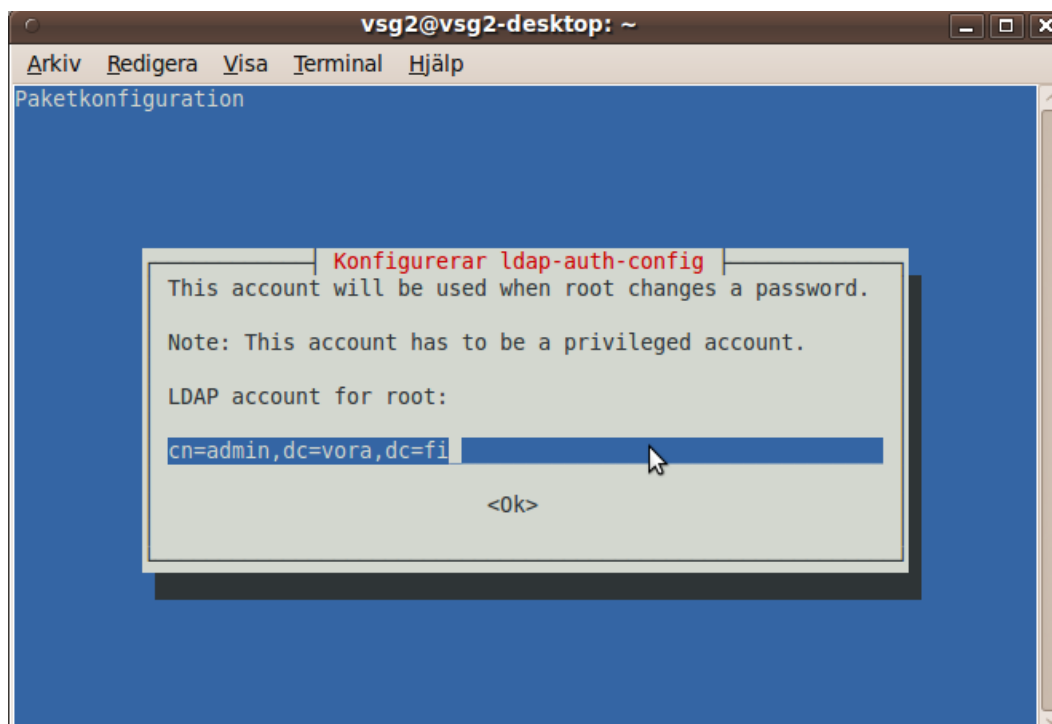
5) **dc=vora,dc=fi**

6) version 3

7) yes

8) no

9) cn=admin,dc=vora,dc=fi



10) ange losenord: v0m200&

11) `sudo nano /etc/pam.d/common-auth`

12) lagg till

`auth optional pam_group.so`

I slutet av filen

13) kopiera in filen `/etc/ldap.conf` (terminal → `sudo nautilus`)

14) installera:

`sudo apt-get install nfs-common libnfsidmap2`

15) `sudo nano /etc/fstab`

16) lagg till

`10.0.0.8:/home/users /home/users nfs defaults 0 0`

langst ned i filen `fstab`.

17) `Sudo nano /etc/nsswitch.conf`

lagg till "ldap" efter `compat`

18) `mkdir /home/users`

## LDAP.CONF

```

####DEBCONF###
##
## Configuration of this file will be managed by debconf as long
as the
## first line of the file says '####DEBCONF###'
##
## You should use dpkg-reconfigure to configure this file via
debconf

# @(#) $Id: ldap.conf,v 1.38 2006/05/15 08:13:31 lukeh Exp $
#
# This is the configuration file for the LDAP nameservice
# switch library and the LDAP PAM module.
#
# PADL Software
# http://www.padl.com

# Your LDAP server. Must be resolvable without using LDAP.
# Multiple hosts may be specified, each separated by a
# space. How long nss_ldap takes to failover depends on
# whether your LDAP client library supports configurable
# network or connect timeouts (see bind_timelimit).
#host 127.0.0.1

# The distinguished name of the search base.
base dc=vora,dc=fi

# Another way to specify your LDAP server is to provide an
uri ldap://10.0.0.8/
# Unix Domain Sockets to connect to a local LDAP Server.
#uri ldap://127.0.0.1/
#uri ldaps://127.0.0.1/
#uri ldapi://%2fvar%2frun%2fldapi_sock/
# Note: %2f encodes the '/' used as directory separator

# The LDAP version to use (defaults to 3
# if supported by client library)
ldap_version 3

# The distinguished name to bind to the server with
# if the effective user ID is root. Password is
# stored in /etc/ldap.secret (mode 600)
#rootbinddn cn=manager,dc=example,dc=net
# Filter to AND with uid=%s
pam_filter objectclass=posixAccount

# Group member attribute
pam_member_attribute memberUid

nss_initgroups_ignoreusers avahi,avahi-
autoipd,backup,bin,daemon,dhcp,games,gdm,gnats,guest,haldaemon,hpl
ip,irc,klog,libuuid,list,lp,mail,man,messagebus,news,openldap,polk
ituser,proxy,pulse,root,saned,sshd,statd,sys,syslog,uucp,www-
data

```

## Bilaga 4. Konfigurationsfiler för brandväggen

```
backup.sh
```

```
# This is a shell script to do backup over a network to
```

```
# an other computer.
```

```
# The script uses scp to copy the data so that backups can be
```

```
# done over insecure connections if needed.
```

```
# To avoid the need for entering a password so needs a
```

```
# public key to be created on the computer
```

```
# and on the specific account from where the
```

```
# backup will be copied.
```

```
# This can be done like this:
```

```
# ssh-keygen -t rsa
```

```
# and press enter when asked for a password phrase.
```

```
# Now copy the id_rsa.pub to the .ssh directory of the
```

```
# remote host you want to do the backup from And to
```

```
# the .ssh directory of the account from where the
```

```
# backup will be done. This is probably from the
```

```
# root account.
```

```
#          scp          ~/.ssh/id_rsa.pub          root@remote-computer-  
name.com:/root/.ssh/authorized_keys2
```

```
# Remote host to where the copy should be copied.
```

```
REMOTEHOST=10.0.0.49
```

```
# Copy the backup file to remote host.

scp -q /etc/iptables.up.rules
root@$REMOTEHOST:/etc/iptables.up.rules

scp -q /etc/dhcp3/dhcpd.conf
root@$REMOTEHOST:/etc/dhcp3/dhcpd.conf

scp -q /etc/havp/havp.config
root@$REMOTEHOST:/etc/havp/havp.config

scp -q /etc/bind/named.conf.options
root@$REMOTEHOST:/etc/bind/named.conf.options

scp -q /etc/network/interfaces
root@$REMOTEHOST:/etc/network/interfaces.working.mode

scp -q /usr/sbin/firewallon
root@$REMOTEHOST:/usr/sbin/firewallon

scp -q /usr/sbin/firewalloff
root@$REMOTEHOST:/usr/sbin/firewalloff

scp -q /etc/openvpn/server.conf
root@$REMOTEHOST:/etc/openvpn/server.conf

scp -q -r /etc/openvpn/keys/*
root@$REMOTEHOST:/etc/openvpn/keys/
```

## dhcpd.conf

```
option domain-name "vora.fi";
option domain-name-servers 10.0.0.7,10.0.0.1;

option subnet-mask 255.255.255.0;
default-lease-time 600;
max-lease-time 7200;

subnet 10.0.0.0 netmask 255.255.255.0 {
    range 10.0.0.50 10.0.0.199;
    option broadcast-address 10.0.0.255;
    option routers 10.0.0.1;
    option subnet-mask 255.255.255.0;
}

#Backup firewall.
host fw {
    hardware ethernet 00:11:2f:aa:40:6d;
    fixed-address 10.0.0.49;
}
```



## Firewall

```
#!/bin/sh

# Start/stop the firewall.

test -f /usr/sbin/firewallon || exit 0

#LSBNAMEs='-1' # Uncomment for LSB name support in
/etc/cron.d/

case "$1" in
start)      echo -n "Starting the firewall"
            /usr/sbin/firewallon
            echo "."
            ;;
stop) echo -n "Stopping the firewall"
      /usr/sbin/firewalloff
      echo "."
      ;;
restart) echo -n "Restarting periodic command scheduler:
cron"
        start-stop-daemon --stop --retry 5 --quiet --pidfile
/var/run/crond.pid --exec /usr/sbin/cron
        start-stop-daemon --start --quiet --pidfile
/var/run/crond.pid --exec /usr/sbin/cron -- $LSBNAMEs
        echo "."
        ;;
reload|force-reload) echo -n "Reloading configuration files
for periodic command scheduler: cron"
                    # cron reloads automatically
                    echo "."
                    ;;
*)
```

```

        ;;
        *)    echo "Usage: /etc/init.d/cron
start|stop|restart|reload|force-reload"
        exit 1
        ;;
    esac
    exit 0

```

### Firewall off

```

#Point this to your copy of ip_tables
IPT="/sbin/iptables"

```

```

    echo "Flush old rules, delete the firewall,tcp-fin chain if
it exists.."
    $IPT -F
    $IPT -F -t nat
    $IPT -X firewall
    $IPT -X tcp-fin

    echo "Setup Masquerading.."
    #Change the IP to your internal network and uncomment
    #this in order to enable it.
    $IPT -t nat -A POSTROUTING -o eth0 -j MASQUERADE
    $IPT -P FORWARD ACCEPT

    echo 1 > /proc/sys/net/ipv4/ip_forward

```

### Firewall on

```

#!/bin/bash
#

```

```
# This is a sample firewall for ip_tables, the tool for doing firewalling
# and masquerading under the 2.3.x/2.4.x series of kernels.
# Be warned, this is a very restrictive set of firewall rules (and they
# should be, for proper security). Anything that you do not _specifically_
# allow is logged and dropped into /dev/null, so if you're wondering why
# something isn't working, check /var/log/messages.
# This is about as close as you get to a 'secure' firewall. It's nasty,
# it's harsh, and it will make your machine nearly invisible to the rest
# of the internet world. Have fun.
# To run this script you must 'chmod 700 iptables-script' and then execute
# it. To stop it from running, run 'iptables -F'
# Turn on Source Address Verification and get
# spoof protection on all current and future interfaces.

    if [ -e /proc/sys/net/ipv4/conf/all/rp_filter ]; then
        echo -n "Setting up IP spoofing protection..."
        for f in /proc/sys/net/ipv4/conf/*/rp_filter; do
            echo 1 > $f
        done
        echo "done."
    else
        echo PROBLEMS SETTING UP IP SPOOFING PROTECTION.  BE
WORRIED.
    fi

    echo "Setting up SYN flood protection"
    echo 1 >/proc/sys/net/ipv4/tcp_syncookies
```

```
#Point this to your copy of ip_tables
IPT="/sbin/iptables"

echo "Flush old rules, delete the firewall,tcp-fin chain if
it exists.."

$IPT -F

$IPT -F -t nat

$IPT -X firewall

$IPT -X tcp-fin

echo "Setup Masquerading.."

#Change the IP to your internal network and uncomment
#this in order to enable it.

$IPT -t nat -A POSTROUTING -o eth0 -j MASQUERADE

$IPT -P FORWARD ACCEPT

echo 1 > /proc/sys/net/ipv4/ip_forward

echo "Set up the firewall table.."

$IPT -N firewall

$IPT -A firewall -j LOG --log-level info --log-prefix
"Firewall:"

$IPT -A firewall -j DROP

echo "Set up the firewall table fin stealth scan
detection.."

$IPT -N tcp-fin

$IPT -A tcp-fin -m state --state ! ESTABLISHED -j LOG --log-
level info --log-prefix "Firewall fin stealth scan:"

$IPT -A tcp-fin -m state --state ! ESTABLISHED -j DROP
```

```
$IPT -A tcp-fin -m state --state ESTABLISHED -j ACCEPT

$IPT -t nat -I PREROUTING 1 -i eth1 -p tcp -s 10.0.0.0/24 --
dport 80 -j REDIRECT --to-port 8080

echo "Forward port 3389 to ip 10.0.0.7"

$IPT -t nat -A PREROUTING -i eth0 -p udp -d 62.148.198.219 -
-dport 3389 -j DNAT --to 10.0.0.7:3389

$IPT -t nat -A PREROUTING -i eth0 -p tcp -d 62.148.198.219 -
-dport 3389 -j DNAT --to 10.0.0.7:3389

echo "INPUT rules start here."

echo "Accept ourselves.."

$IPT -A INPUT -s 127.0.0.1/32 -d 127.0.0.1/32 -j ACCEPT

# VPN network

$IPT -A INPUT -s 10.11.10.0/24 -j ACCEPT

# Local network.

$IPT -A INPUT -s 10.0.0.0/24 -j ACCEPT

echo "Accept DNS, 'cause it's warm and friendly.."

$IPT -A INPUT -p udp --source-port 53 -j ACCEPT

echo "Accept SSH."

$IPT -A INPUT -p tcp --destination-port 22 -j ACCEPT

echo "Accept OpenVPN (udp 1194)."
```

```
$IPT -A INPUT -p udp --destination-port 1194 -j ACCEPT
```

```
    echo "Allow NTP time server"

    $IPT -A INPUT -p tcp --sport 123 -m state --state
    ESTABLISHED,RELATED -j ACCEPT

    $IPT -A INPUT -p udp --sport 123 -m state --state
    ESTABLISHED,RELATED -j ACCEPT

    echo "Allow webmin."

    $IPT -A INPUT -p tcp --destination-port 10000 -j ACCEPT

    echo "Send everything else to the firewall and log it and
    drop it...."

    echo "Drop ping packets."

    $IPT -A INPUT -p icmp -m state --state ! ESTABLISHED -j
    firewall

    echo "Drop connections with sync, used to request TCP
    connection initiation."

    $IPT -A INPUT -p tcp --syn -j firewall

    echo "Drop packets associated with no known connection."

    $IPT -A INPUT -m state --state INVALID -j firewall

    echo "Drop udp packets."

    $IPT -A INPUT -p udp -m state --state ! ESTABLISHED -j
    firewall

    echo "Send FIN packets to the tcp-fin to check if they are
    belonging to an established connection."

    $IPT -A INPUT -p tcp --tcp-flags ALL FIN -j tcp-fin

    echo "FORWARD rules start here."
```

```
echo "Allow NTP time server"

$IPT -A FORWARD -p tcp --sport 123 -m state --state
ESTABLISHED,RELATED -j ACCEPT

$IPT -A FORWARD -p udp --sport 123 -m state --state
ESTABLISHED,RELATED -j ACCEPT

echo "Accept DNS"

$IPT -A FORWARD -p udp --source-port 53 -j ACCEPT

echo "Accept f-secure backweb."

$IPT -A FORWARD -p udp --destination-port 370 -j ACCEPT
$IPT -A FORWARD -p udp --source-port 371 -j ACCEPT
$IPT -A FORWARD -p udp --destination-port 371 -j ACCEPT
$IPT -A FORWARD -p udp --destination-port 9370:9400 -j
ACCEPT

echo "Allow VPN-traffic between internal networks."

$IPT -A FORWARD -s 10.11.10.0/24 -d 10.0.0.0/24 -j ACCEPT
$IPT -A FORWARD -s 10.0.0.0/24 -d 10.11.10.0/24 -j ACCEPT

echo "Allow localnet as forward."

$IPT -A FORWARD -i eth1 -j ACCEPT

echo "Allow related connections for ftp"

$IPT -A FORWARD -m state --state ESTABLISHED,RELATED -j
ACCEPT

echo "Allow remote desktop upd 3389"

$IPT -A FORWARD -p udp --destination-port 3389 -j ACCEPT
$IPT -A FORWARD -p tcp --destination-port 3389 -j ACCEPT
```

```
    echo "Send everything else to the firewall for local
computers"

    echo "Drop ping packets."

    $IPT -A FORWARD -p icmp -m state --state !
ESTABLISHED,RELATED -j firewall

    echo "Drop connections with sync, used to request TCP
connection initiation."

    $IPT -A FORWARD -p tcp --syn -j firewall

    echo "Drop packets associated with no known connection."

    $IPT -A FORWARD -m state --state INVALID -j firewall

    echo "Drop udp packets."

    $IPT -A FORWARD -p udp -m state --state !
ESTABLISHED,RELATED -j firewall

    echo "Send FIN packets to the tcp-fin to check if they are
belonging to an established connection."

    $IPT -A FORWARD -p tcp --tcp-flags ALL FIN -j tcp-fin
```



havp

```
SERVERNUMBER 80  
TRANSPARENT true  
MAXSCANSIZE 0  
KEEPBACKBUFFER 2000000  
ENABLECLAMLIB true  
ENABLECLAMD false  
ENABLEFPROT false  
ENABLEAVG false  
ENABLEAVESERVER false  
ENABLESOPHIE false  
ENABLETROPHEIE false  
ENABLENOD32 false  
ENABLEAVAST false  
ENABLEARCAVIR false  
ENABLEDRWEB false
```

interfaces

# This file describes the network interfaces available on your system  
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface

auto lo

iface lo inet loopback

# The primary network interface

#allow-hotplug eth0

#iface eth0 inet dhcp

post-up iptables-restore < /etc/iptables.up.rules

auto eth0

iface eth0 inet static

address 62.148.198.219

netmask 255.255.255.248

network 62.148.198.216

broadcast 62.148.198.223

gateway 62.148.198.217

post-up iptables-restore < /etc/iptables.up.rules

auto eth1

iface eth1 inet static

address 10.0.0.1

netmask 255.255.255.0

broadcast 10.0.0.255

network 10.0.0.0

## named.conf.options

```
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers
you want

    // to talk to, you may need to fix the firewall to
allow multiple

    // ports to talk. See
http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for
stable

    // nameservers, you probably want to use them as
forwarders.

    // Uncomment the following block, and insert the
addresses replacing

    // the all-0's placeholder.

    // forwarders {
    //     0.0.0.0;
    // };

    auth-nxdomain no;    # conform to RFC1035

    listen-on-v6 { any; };

    allow-transfer { none; }; // no transfers allowed
    allow-query { any; }; // query access for all
    allow-recursion { any; }; // restrict recursion
};
```

```
server.conf

# Which TCP/UDP port should OpenVPN listen on?

# If you want to run multiple OpenVPN instances
# on the same machine, use a different port
# number for each one. You will need to
# open up this port on your firewall.

    port 1194

# TCP or UDP server?

;proto tcp

    proto udp

# "dev tun" will create a routed IP tunnel,
# "dev tap" will create an ethernet tunnel.
# Use "dev tap0" if you are ethernet bridging
# and have precreated a tap0 virtual interface
# and bridged it with your ethernet interface.
# If you want to control access policies
# over the VPN, you must create firewall
# rules for the the TUN/TAP interface.
# On non-Windows systems, you can give
# an explicit unit number, such as tun0.
# On Windows, use "dev-node" for this.
# On most systems, the VPN will not function
# unless you partially or fully disable
```

```
# the firewall for the TUN/TAP interface.

;dev tap

    dev tun

# Any X509 key management system can be used.

# OpenVPN can also use a PKCS #12 formatted key file

# (see "pkcs12" directive in man page).

    ca keys/ca.crt

    cert keys/server.crt

    key keys/server.key

# Diffie hellman parameters.

# Generate your own with:

# openssl dhparam -out dh1024.pem 1024

# Substitute 2048 for 1024 if you are using

# 2048 bit keys.

    dh keys/dh1024.pem

# Configure server mode and supply a VPN subnet

# for OpenVPN to draw client addresses from.

# The server will take 10.8.0.1 for itself,

# the rest will be made available to clients.

# Each client will be able to reach the server

# on 10.8.0.1. Comment this line out if you are

# ethernet bridging. See the man page for more info.

    server 10.11.10.0 255.255.255.0
```

```
# Maintain a record of client <-> virtual IP address
# associations in this file. If OpenVPN goes down or
# is restarted, reconnecting clients can be assigned
# the same virtual IP address from the pool that was
# previously assigned.
```

```
ifconfig-pool-persist ipp.txt
```

```
# Push routes to the client to allow it
# to reach other private subnets behind
# the server. Remember that these
# private subnets will also need
# to know to route the OpenVPN client
# address pool (10.8.0.0/255.255.255.0)
# back to the OpenVPN server.
```

```
;push "route 192.168.10.0 255.255.255.0"
```

```
;push "route 192.168.20.0 255.255.255.0"
```

```
push "route 10.0.0.0 255.255.255.0"
```

```
# Certain Windows-specific network settings
```

```
# can be pushed to clients, such as DNS
```

```
# or WINS server addresses. CAVEAT:
```

```
# http://openvpn.net/faq.html#dhcpcaveats
```

```
;push "dhcp-option DNS 10.8.0.1"
```

```
;push "dhcp-option WINS 10.8.0.1"
```

```
push "dhcp-option WINS 192.168.0.1"
```

```
# The keepalive directive causes ping-like
# messages to be sent back and forth over
# the link so that each side knows when
# the other side has gone down.

# Ping every 10 seconds, assume that remote
# peer is down if no ping received during
# a 120 second time period.
```

```
keepalive 10 120
```

```
# Enable compression on the VPN link.
# If you enable it here, you must also
# enable it in the client config file.
```

```
comp-lzo
```

```
# The persist options will try to avoid
# accessing certain resources on restart
# that may no longer be accessible because
# of the privilege downgrade.
```

```
persist-key
```

```
persist-tun
```

```
# Output a short status file showing
# current connections, truncated
# and rewritten every minute.
```

```
status openvpn-status.log
```

```
# Set the appropriate level of log
# file verbosity.
# 0 is silent, except for fatal errors
# 4 is reasonable for general usage
# 5 and 6 can help to debug connection problems
# 9 is extremely verbose

    verb 3
```