BACHELOROPPGAVE:

OurBox

FORFATTERE:

Gavin Thomas Garrad

Stepan Maluchev

DATO:

15.05.2015

Sammendrag av Bacheloroppgaven

Tittel:	OurBox	Nr: -
		Dato: 15.05.2015
Deltakere:	Gavin Thomas Garrad Stepan Maluchev	
Veiledere:	Erik Heljmås	
Oppdragsgiver:	Høgskolen i Gjøvik	
Kontaktperson:	Thommas Kimmerich	
Stikkord	Norway, Norsk	
Antall sider: 125	Antall vedlegg: 15	Tilgjengelighet: Åpen

Kort beskrivelse av bacheloroppgaven:

Cisco labben på HiG består per dags dato av gamle Windows 7 maskiner som er tilgjengelige for alle som befinner seg på labben. Disse maskinene har studentene full administrator rettigheter på, noe som tilsier at en student kan gjøre hva de vil på den, mens maskinenes egentlige hensikt er kun å brukes sammen med Cisco utstyr for konfigurasjon og testing.

Prosjektets mål er å lage en prototype hvor studentene ikke har full administrator rettigeheter på selve maskinen, men heller administrator rettigheter på de virtuelle maskinene som blir laget, slik at studentene har sin egen testomgivelse som ikke kan på virke andre studenter.

Rapportens innhold har blitt skrevet slik at de som skal videreutvikle prosjektet vil kunne forstå våre valg og vurderinger utifra teoretiske og praktiske begrunnelser.

Summary of Graduate Project

Title:	OurBox	Nr: -
		Date: 15.05.2015
Participants:	Gavin Thomas Garrad Stepan Maluchev	
Supervisor:	Erik Heljmås	
Employer:	Høgskolen i Gjøvik	
Contact person:	Thommas Kimmerich	
Keywords	Virtualisation, LDAP, Window Manager, Ubuntu	
Pages: 125	Appendixes: 15	Availability: Open

Short description of the main project:

The Cisco lab at HiG consist of old computers running on Windows 7 and is available for all students who has access to the lab. On these machines students has full administrator rights, which means that the students can do what ever they want on them, while the purpose of these machines are to be used with Cisco equipment for configuration and testing.

The project's goal is to make a prototype where the students does not have full administrative rights on the host machine, but rather on the virtual machines on the host. The students will have their own test domains which will not effect the other students in any way.

The contents of the report has been written in such way that for those who are going to further develop this project will understand our choices based on theoretical and practical reasons.

Preface

This bachelor project has been a very interesting and challenging project. We have learned a lot of new things and have now a much better understanding on how Ubuntu, virtualization, Virtualbox, KVM, Fluxbox, LDAP,AD and Google-Search works. The ability to find the specific information that is needed online is an art, and this skill has improved a lot.

The most interesting subject throughout this project has been the authentication part. When we first encountered this subject, we had absolutely no idea on how an authentication through a Linux system worked. Now we can set up this system without any problems.

- A huge thanks to our tutor Erik Hjelmås and Jon Langseth who has guided and helped us to stay on the right path throughout this project.
- Thanks to our employer Thomas Kimmerich who has given us this interesting project.
- Thanks to our family who has been supportive and understandable that this has been a tough semester with no time for mingling.

Contents

List of Figures

List of Tables

Listings

Summary

A short project description:
The Cisco lab at HiG consist of old computers running on Windows 7 and is available for all students who has access to the lab. On these machines students has full administrator rights, which means that the students can do what ever they want on them, while the purpose of these machines are to be used with Cisco equipment for configuration and testing.

The project's goal is to make a prototype where the students does not have full administrative rights on the host machine, but rather on the virtual machines on the host. The students will have their own test domains which will not effect the other students in any way.

The contents of the report has been written in such way that for those who are going to further develop this project will understand our choices based on theoretical and practical reasons.

Kort beskrivelse av bacheloroppgaven:

Cisco labben pÃě HiG bestÃěr per dags dato av gamle Windows 7 maskiner som er tilgjengelige for alle som befinner seg pÃě labben. Disse maskinene har studentene full administrator rettigheter pÃě, noe som tilsier at en student kan gjÃÿre hva de vil pÃě den, mens maskinenes egentlige hensikt er kun Ãě brukes sammen med Cisco utstyr for konfigurasjon og testing.

Prosjektets mÃěl er Ãě lage en prototype hvor studentene ikke har full administrator rettigeheter pÃě selve maskinen, men heller administrator rettigheter pÃě de virtuelle maskinene som blir laget, slik at studentene har sin egen testomgivelse som ikke kan pÃěvirke andre studenter.

Rapportens innhold har blitt skrevet slik at de som skal videreutvikle prosjektet vil kunne forstÃě vÃěre valg og vurderinger utifra teoretiske og praktiske begrunnelser.

Foreword

1 Introduction

1.1 Problem

The way that the students work in the Cisco lab today, is not in a secure and responsible way. Therefor, Thomas Kimmerich wants to do some upgrades and improve this. He also want it to be possible to do different activities on those large networks, such as CTF (Capture the flag).

The way the Cisco lab is set up today does not give the possibility for making any large network. There is about 12 computers in a room and they are not strong enough to run virtual machines. These 12 machines are not connected together, and they all got a pod where they can connect to the switches and routers. Also, when users log on to one of these 12 computers, there is no kind of authentication, so every user has administrator rights on the host PC. Because of this, there has been cases where the machines has been "destroyed" and Thomas had to reinstall the OS on the computers. He wants HiG to have a similar system to what they got at his last workplace in Germany, Bremen.

The system should be easy to configure and maintain through a management interface.

1.1.1 Demarcation

This project is only a little part of a big project, which will at the end be very usable and helpful for solving many tasks, as mentioned in section 1.1. In this first part, we have some demarcations. This is meant so we can be done with this part to the deadline in May, and therefore we have some limits set:

- No need for developing/creating a GUI for managing the hypervisor/VMs. This can be done later by someone else. Our requirements will be to let the management be done by a command-line interface.
- We don't need to set up and make all hosts be manageable from one manager-node or let them be manageable through VPN. The important part here is to have in mind that this will and should be possible at a later stage.
- We only need to make this solution work on one host, but this is something that should be easily done on many hosts, if not by the end of this project, then short time after.

- We are only going to create a prototype of this system.

1.1.2 Task description

This project has its main focus on the authentication part, however, several other important aspects are included to form this project, and they are:

1. Only a hypervisor will be running.
2. Login-prompt for authentication.
3. Be able to connect to the internet and Cisco internal network.
4. Make a prototype running on one host.

Now lets fulfill those points and make them more detailed:

The first thing we will be doing, will be to make one host (PC) to boot directly into a hypervisor, without booting up an entirely OS (like windows 7). This way, we will not need to waste resources of having an entirely OS running, and upon that a hypervisor. This will also minimize the ability for users to ruin the host's configuration settings.When we mention that no OS will be started, only a hypervisor, we meant that this will be our goal. On the other hand, if we cannot make this happen, a hypervisor running upon an OS will also be accepted.

Next step will be to make/create a login prompt. This way, the users/administrators must enter a username and a password, this will be the same username and password as a user will enter when logging into fronter.no/hig or into HiG‘s website. The purpose of this step is to make a clear authority line between what an administrator can do and what a user can do inside the hypervisor. To mention the differences, a user will be able to create, modify and delete virtual machines, while an administrator can make changes on the hypervisor itself, and create, etc. VMs. Also, a guest account will be created.

The third important thing will be to let the virtual machines have access to different networks. A host will be able to connect to the "internet" (HiG‘s backbone) and the intern Cisco network (which will be the local Cisco network inside the Cisco-lab). This way, each VM will have the ability to be connected on both networks, if necessary.

The fourth step, as mentioned above where we mentioned that this solution should be implemented on one host to begin with, will be our prototype solution. At a later stage, our prototype should be implemented on 6 hosts. This will be done by project owner. Also another thing we should have in mind, is that those hosts should be able to be configured remotely. This will allow an administrator to sit in front of one host (his/her own PC) and configure several hosts simultaneously (e.g. creating mail servers).

1.2 Target Group

The target for this report will mainly be for those who are going to do further work on this solution, and for those in general who is interested to learn about our system. This report should help the other bachelor thesis to see what we have done and why we have done it as we have.

The actual project is targeted to the students/teachers who are going to use it, mainly Thomas Kimmerich and the Cisco courses he does in the Cisco lab.

1.3 Subject

So why did we choose to write about this subject? Well, the topic of this thesis do we both find very interesting, challenging and learnful, and afteral this topic is very relevant for our background (see section 1.4). This thesis will not only let us be able to help and upgrade the HiG's Cisco-lab, but also if everything goes as planed and the Cisco-lab will take in use our new solution, we will have our names in that lab, at least for some time.

1.4 Our background

We both have the same background from a three year experience at HIG. The course we are taking is "Drift av nettverk og datasystemer" and contains the subjects:

- **Basics of programmering**
- Introduction to information security
- Mathematics for information technology
- Object oriented programming
- **Data communication and network security**
- IT Service Management
- Statistics
- **Network administration**
- Data modeling og database design
- Algorithmic methods

- **System development**
- **Operating systems**
- Database- og application running
- **System administration**
- Mainframe
- Ethical Hacking and Penetration Testing (Stepan Maluchev)
- IT leadership (Stepan Maluchev)
- Software development (Gavin Garrad)
- WWW-Technology

The highlighted subjects is what we have had most use of in our bachelor thesis.

1.5 Framework

How did we solve the planning and frameworks, and what did we have in mind for getting this project done to the deadline? We are now going to take a closer look at what was needed to be done before we could start working with the actual project.

1.5.1 Development model

We agreed right away that the system development model we are going to use is incremental model. Since we think that this will benefit us the most, we could have gone for waterfall model, but we feel that this is too rigid, we want to have the possibility to change previous steps. That is why we think it will be easier for us to use the incremental model. Scrum and XP where out of the question. We did not want to go with a very agile method development models, since we have a deadline for when this project has to be finished. Basically why we chose incremental model is because it will make it some what easier for us to just focus on getting one part of the system to work, then check if it really is working and then integrate it with our system. This is mainly why we want to use this method, since in this point of the model we feel we also can make changes to the system and make it work better. Had we gone for the waterfall model we could not have done that. After we have integrated it into our system, we will need to validate that the whole system works.

1.5.2 Gantt-diagram

To make and show our activities against time, we created a gantt-diagram (see Figure on page 6). On the left side of the diagram is the activities listed up, and on the right side is the time scale we think will be suitable for each activity. Notice that there are 5 deadlines (marked with red), and those will be very important to not override. It's also important to have in mind that this gantt-diagram is a "working plan" we think will be the right way to work after. Often there are problems which comes along the way, and therefor the schedule must be changed.

And yes, when we was half way in the project, we needed to create a new gantt-diagram which was more specific and precise (see Figure on page 7). The new gantt-diagram shows that our first "working plan" were relatively good structured when it comes to the time scope and how much time we needed for getting those activities done. However, we spend more time then we thought on testing and making the final decision on which hypervisor we are going to use.

1.6 Roles

In this project there will be four persons involved:

- **Employer:** *Is the one who have given us this project. He is the system's owner.*
  Thomas Kemmrich
- **Tutor:** *He will be our tutor throughout this project. He will help us if we get something we don't know how to do. He is not responsible to give us the answer to the problems along the way, but he can tell us a possibility how to solve a problem.*
  Erik Hjelm?s
- **Project leader:**
  Stepan Maluchev
- **Member:** *Team worker on this project.*
  Gavin Thomas Garrad

1.7 Terminologi

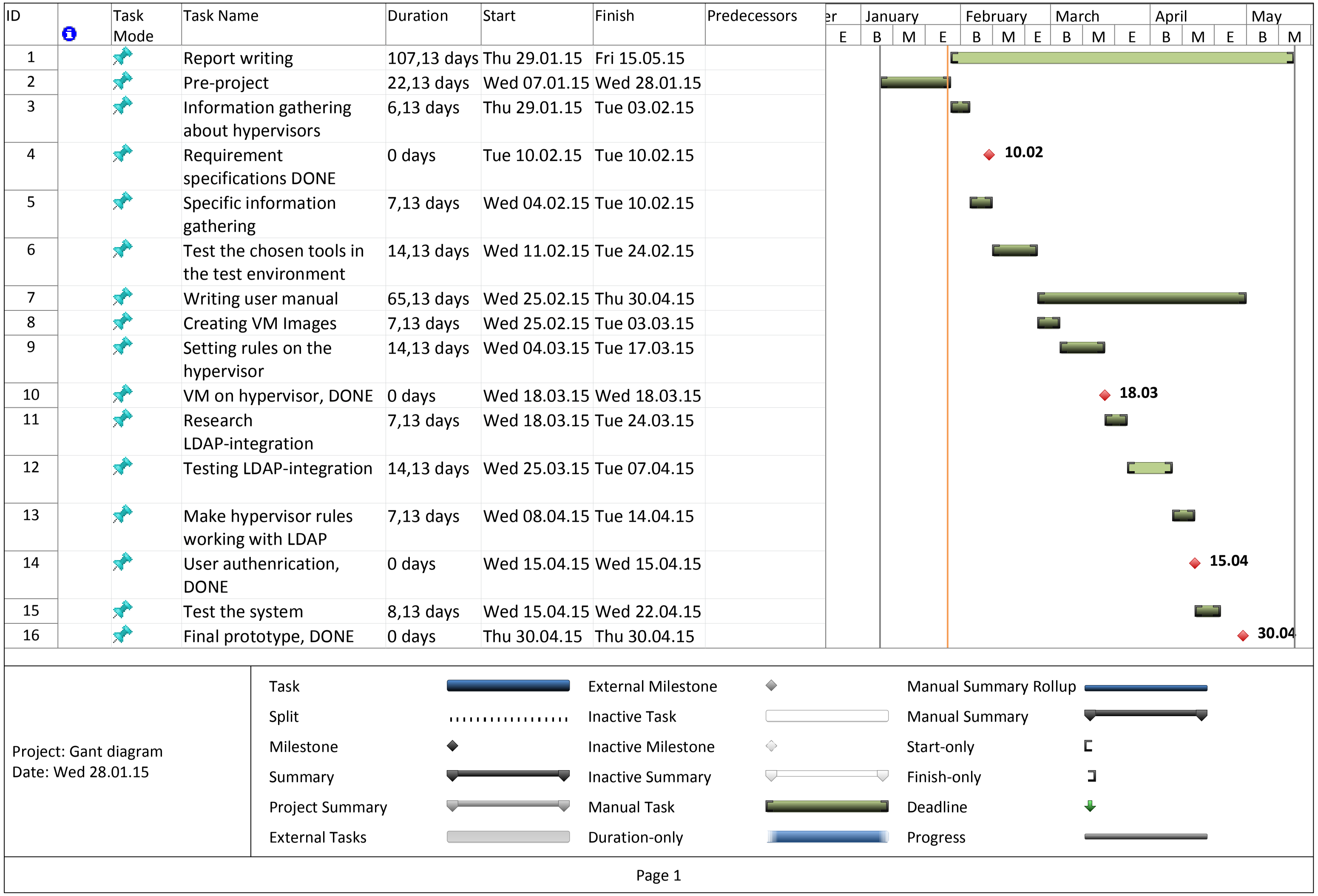
LDAP is a protocol used for accessing and maintaining information over the internet (see section 3.1).

PAM is a mechanism (framework) which makes it possible for applications to authenticate against an LDAP-server and authenticate related activities (see section 3.2.1).

Cisco is basically a manufacture of network equipments and design. A Cisco-lab is referred to a lab containing Cisco equipments where testing and learning is performed.

TTY is a shell/terminal where a user only get a CLI to work with.

OS is an operating system.

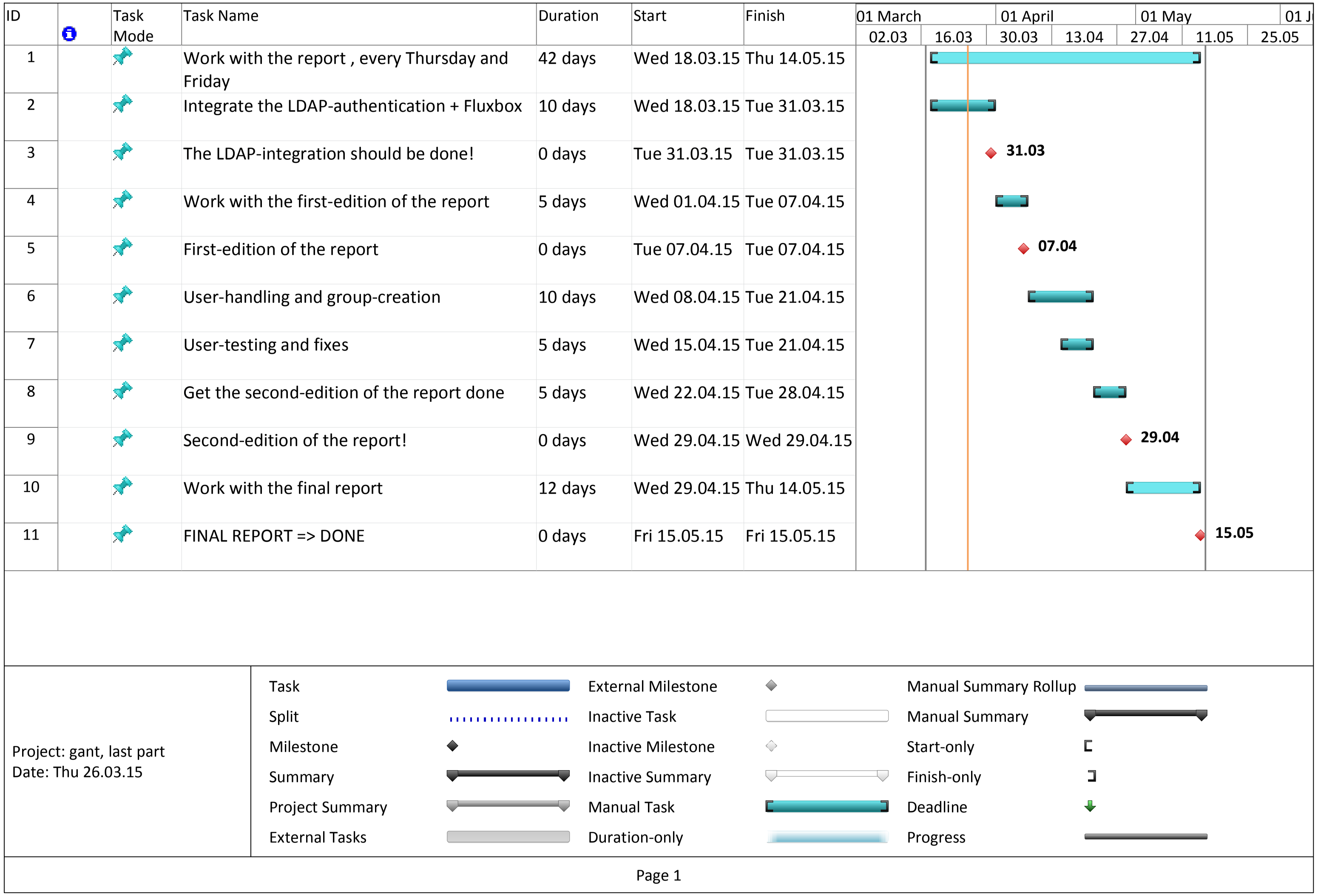


ID		Task Mode	Task Name	Duration	Start	Finish	Predecessors
1			Report writing	107,13 days	Thu 29.01.15	Fri 15.05.15	
2			Pre-project	22,13 days	Wed 07.01.15	Wed 28.01.15	
3			Information gathering about hypervisors	6,13 days	Thu 29.01.15	Tue 03.02.15	
4			Requirement specifications DONE	0 days	Tue 10.02.15	Tue 10.02.15	
5			Specific information gathering	7,13 days	Wed 04.02.15	Tue 10.02.15	
6			Test the chosen tools in the test environment	14,13 days	Wed 11.02.15	Tue 24.02.15	
7			Writing user manual	65,13 days	Wed 25.02.15	Thu 30.04.15	
8			Creating VM Images	7,13 days	Wed 25.02.15	Tue 03.03.15	
9			Setting rules on the hypervisor	14,13 days	Wed 04.03.15	Tue 17.03.15	
10			VM on hypervisor, DONE	0 days	Wed 18.03.15	Wed 18.03.15	
11			Research LDAP-integration	7,13 days	Wed 18.03.15	Tue 24.03.15	
12			Testing LDAP-integration	14,13 days	Wed 25.03.15	Tue 07.04.15	
13			Make hypervisor rules working with LDAP	7,13 days	Wed 08.04.15	Tue 14.04.15	
14			User authenrication, DONE	0 days	Wed 15.04.15	Wed 15.04.15	
15			Test the system	8,13 days	Wed 15.04.15	Wed 22.04.15	
16			Final prototype, DONE	0 days	Thu 30.04.15	Thu 30.04.15	

Project: Gant diagram
Date: Wed 28.01.15

Task		External Milestone		Manual Summary Rollup	
Split		Inactive Task		Manual Summary	
Milestone	◆	Inactive Milestone	◇	Start-only	[
Summary		Inactive Summary		Finish-only	]
Project Summary		Manual Task		Deadline	⇩
External Tasks		Duration-only		Progress	

Page 1



ID		Task Mode	Task Name	Duration	Start	Finish
1			Work with the report , every Thursday and Friday	42 days	Wed 18.03.15	Thu 14.05.15
2			Integrate the LDAP-authentication + Fluxbox	10 days	Wed 18.03.15	Tue 31.03.15
3			The LDAP-integration should be done!	0 days	Tue 31.03.15	Tue 31.03.15
4			Work with the first-edition of the report	5 days	Wed 01.04.15	Tue 07.04.15
5			First-edition of the report	0 days	Tue 07.04.15	Tue 07.04.15
6			User-handling and group-creation	10 days	Wed 08.04.15	Tue 21.04.15
7			User-testing and fixes	5 days	Wed 15.04.15	Tue 21.04.15
8			Get the second-edition of the report done	5 days	Wed 22.04.15	Tue 28.04.15
9			Second-edition of the report!	0 days	Wed 29.04.15	Wed 29.04.15
10			Work with the final report	12 days	Wed 29.04.15	Thu 14.05.15
11			FINAL REPORT => DONE	0 days	Fri 15.05.15	Fri 15.05.15

Project: gant, last part
Date: Thu 26.03.15

Task	External Milestone	Manual Summary Rollup
Split	Inactive Task	Manual Summary
Milestone	Inactive Milestone	Start-only
Summary	Inactive Summary	Finish-only
Project Summary	Manual Task	Deadline
External Tasks	Duration-only	Progress

Page 1

2 Requirements

To get a better understanding of our work and thesis, the requirements have been important for us to understand what Thomas wants. It took some for us to understand what he really wanted. Here is the requirements:

1. If possible no OS will be running on the hosts, only a hypervisor which boots up.
2. No access to the hypervisor by the users.
3. A list of virtual machines can be start by the users.
4. The virtual machines must be able to have accounts/users.
5. Connect with HIG LDAP for authentication.
6. Authorised by a local entity (cisco lab).
7. As a Guest user.
8. Groups has to exists.
9. Each group contains of one or more users.
10. They share one folder, where the VMs for the group will be stored.
11. Each user virtual machine will be store in the group folder.
12. Administrators will have full permission to change configuration settings on the hypervisor.
13. Students should be able to choose which NIC the virtual machine is going to use. Virtual machines shall have access to the HW.
14. Students will be able to choose how much memory, etc. the virtual machine are going to use. Within defined limits.
15. The student will not be able to make changes on the host, only make changes to running virtual machine's.
16. After a student has created a snapshot, the snapshot should be able to be stored on the host, so that later, the student can find that virtual machine on the same host.
17. Each virtual machine must be separated, no shared folders, etc. (SF shall be possible)
18. Each host will run several virtual machine's when the host is at full capacity.

19. The administrator will be able to change how many virtual machine's the host can run when the host is at full capacity, but this will of course be restricted to the host's hardware itself.
20. Configuration will be possible through command line or GUI.
21. A centralized management solution shall be possible.

3 Theory

3.1 LDAP

The word "LDAP" (Lightweight Directory Access Protocol) [7] is a protocol used for accessing and maintaining information over the internet. The main purpose of this protocol is that it can send and retrieve records with a hierarchical structure, e.g. information about persons, mail lists, phone lists, etc. This protocol can also be used when we want to compare an attributes value against another value.

An "LDAP-server" is referred to a server running software like Active Directory (Windows) or OpenLDAP (Linux) (supported list of LDAP softwares [8]), and these are servers that a "client" can authenticate users against.

3.2 PAM

3.2.1 What is PAM

PAM (Pluggable authentication module)[9] is a mechanism (framework) which makes it possible for applications to authenticate against an LDAP-server and authenticate related activities.

While LDAP is the protocol where the information goes forth and back, PAM is a library with all necessary code for an application to perform an authentication against an LDAP server.

> *"The core pieces of PAM are a library (libpam) and a collection of PAM modules, which are dynamically linked libraries (.so) files in the folder /lib/security."* | [1]

3.2.2 Advantages of PAM

> 1. *It provides a common authentication scheme that can be used with a wide variety of applications.*
> 2. *It allows great flexibility and control over authentication for both the system administrator and application developer.*
> 3. *It allows application developers to develop their program without implementing a particular authentication scheme. Instead, they can focus purely on the details of their program.*
>
> | [10]

3.2.3 PAM configurations

A service which wants to use PAM has to have its own PAM configuration in **/etc/pam.d/**, with the service name as the name of the configuration file. There are four different default modules which a service can include in its configuration file. The purpose of these modules are to avoid that every service create its own way to e.g. authenticate itself, but rather include a module. Short description of those four modules (taken from [1]):

- **common-auth:** This module is the one which validate a user with valid credentials.
- **common-account:** This decides if the user can get a valid account on the local machine or not.
- **common-session:** This adds all the necessary resources a user my be needing, e.g. displaying a message of the day or mounting the user into a homedirectory.
- **common-password:** This module is used when a user is updating or changing their own credentials, e.g. password.

A PAM configuration file consists of a list of rules (see Figure 1).

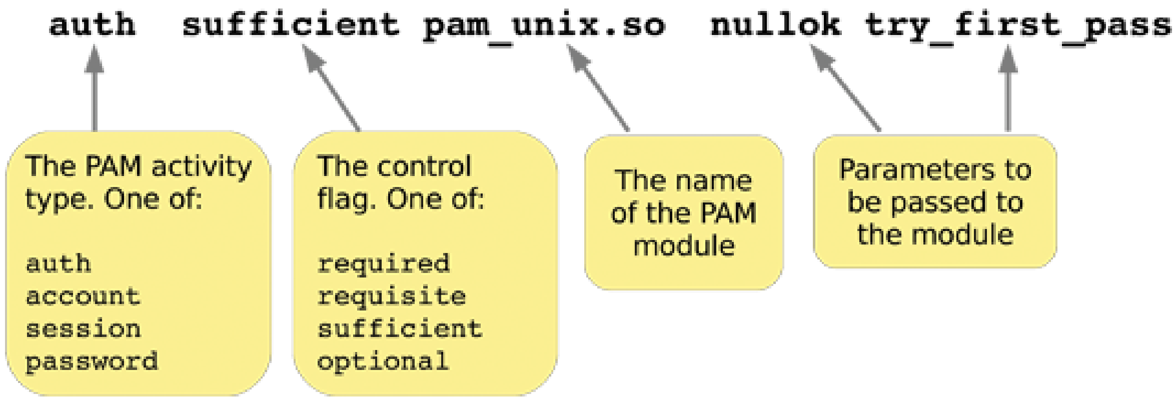

auth sufficient pam_unix.so nullok try_first_pass
The PAM activity type. One of:
auth
account
session
password
The control flag. One of:
required
requisite
sufficient
optional
The name of the PAM module
Parameters to be passed to the module

Figure 1: PAM configuration syntax [1]

The first token tells PAM what kind of authentication type it will use, supports 4 types. The second token is a control-flag which lets PAM choose what to do if the rule fails. It supports 4 different control types:

- **requisite:** If the authentication fails via this module, the whole PAM authentication process stops with an error.
- **required:** If the authentication fails via this module, PAM will return an error to the application, but it will simultaneously call the other modules in the stack.

- **sufficient:** If the authentication succeeds via this module, PAM will stop trying other modules and grant the authentication.
- **optional:** This is only important if it is defined one place while associated with a service type.

On the third place there is the name of the PAM module the rule is going to use. For example when a user is authenticating with a local password, then **pam_unix.so** is the standard PAM module used. The last token is the argument parameter itself which will be sent to the module.

3.3 Explanation of the package *libpam-ldapd*

3.3.1 What is *libpam-ldapd*

This package is a management tool for Unix systems that allows Unix to perform remote authentication and authorization via an LDAP-server.

libpam-ldapd is an updated version of *libpam-ldap* 3.4. This is basically an update of the old NSS module, but with some changes in the design structure.

The great thing with libpam-ldapd is that it uses daemons which caches and reuses the queries and data, which will reduce the overall network traffic and improved performance. Without those daemons, every service needs to set up its own LDAP connection and tear it back down.

3.3.2 Packet's content

libpam-ldapd package consists of several other packages and modules which let a Unix system to perform remote authentication and get its identity via an LDAP-server. The package consist of:

1. **ldap-utils:** Contains all the client programs required to access an LDAP server. The most common is *ldapsearch*, which is used to search and display entries (for a overview of all tools available visit [11]).
2. **libnss-ldapd:**Contains NSS (Name Service Switch) module for using LDAP as a naming service. This means that the LDAP-server can be used to retrieve the same information about user account, group, host name, alias or netgroup which can be find in /etc/ (must be flat or NIS files).
3. **libpam-ldapd:** This is a PAM module which provides password management, authorization and authentication based on credentials stored in an LDAP-server [12].
4. **nscd:** Is a Name Service Cache Deamon which handles passwd, group and host lookups. This deamon caches this information and uses it in other queries.
5. **nslcd:** Is a deamon for NSS and PAM lookups when used against an LDAP-server.

3.3.3 nslcd.conf

All the necessary configurations for *nslcd daemon* is configured in **/etc/nslcd.conf**. The **nslcd.conf** also consist of what information the **nslcd daemon** should retrieve from the LDAP-server.

Basic configuration and explanation

There are six configuration attributes in the configuration file which must be configured for the deamon to work (see figure: 4.3):

Listing 3.1: nslcd.conf basic

```
uri ldap://128.39.140.10
base ou=student,dc=hig,dc=no
binddn cn=120683,ou=12HBWUA, ou=student,dc=hig,dc=no
bindpw "PASSWORD"
scope sub
ldap_version 3
```

Attribute explanation:

uri ldap:// tells LDAP where the LDAP-server is located. Preferably with an IP-address as shown in 4.3.

baseDN is where the search will start. In this example, the search will start in the folder *student* and search through everything inside this folder.

bindDN is the DN of our user account in the HiG's LDAP directory (where the user is located). **OU** stands for *Organization Unit* and in practice is like a folder. The **CN** stands for *Common Name* and is usually the "end user" or the last piece of the search string. We can look at the DN as a tree with branches. Here we can see that this user **120683** is located first from the root directory in the folder named **student**, then in another folder named **12HBWUA**.

bindpw is the password that is needed to authenticate the bindDN.

scope means how the data is structured inside the LDAP-database and how the ldapsearch will perform the search. **sub** stands for *subtree* and indicate searching of all entries at all levels under and including the specified baseDN[2] (see the different scopes in Figure 2).

ldap_version 3 is needed to tell the LDAP-server that the client want to use the newest version of LDAP with all its new futures (a list over what's new in version 3 [13]).

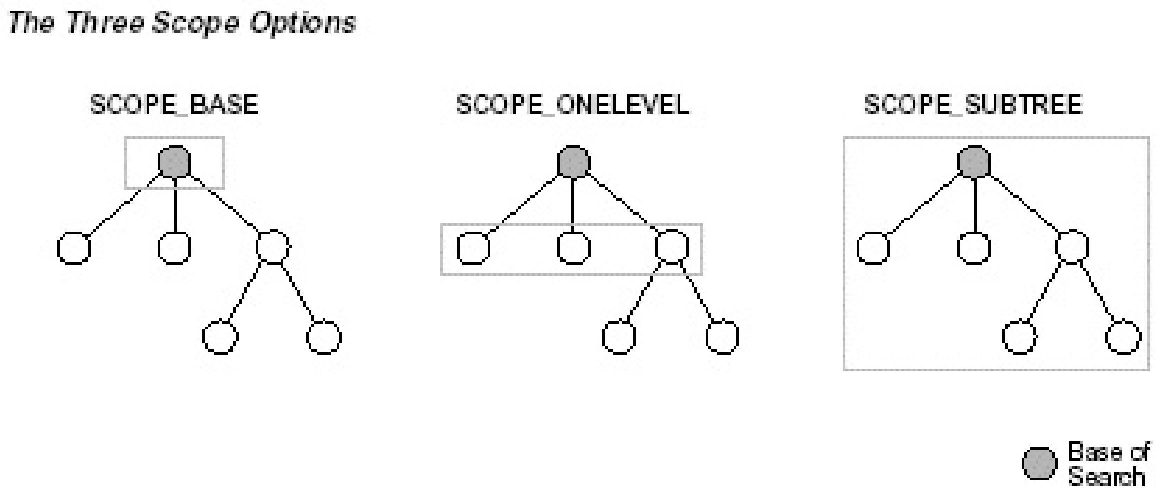


Figure 2: Example of scopes | [2]

nslcd.conf mapping

This is how a mapping in Ubuntu may look like against an active directory server.

Listing 3.2: Mapping in nslcd.conf

```
1  filter passwd (&(objectClass=user))
2  map passwd uid              sAMAccountName
3  map passwd gidNumber        primaryGroupID
4  map passwd homeDirectory    homeDirectory
5  map passwd gecos            description
6  map passwd loginShell       "/bin/bash"
7  map passwd uidNumber        msSFU30UidNumber
8  filter shadow (&(objectClass=user))
9  map shadow uid              sAMAccountName
10 map shadow shadowLastChange pwdLastSet
```

Line 1, 8: is a command for setting a search filter for a specific map. This sets the *passwd/shadow* filter attribute to **user** (default this is set to *postixAccount*).

Line 2, 9: **uid** is the attribute which tells the username of the user.

Line 3: **gidNumber** is the ID of the group which the user will be a member of.

Line 4: **homeDirectory** is where the homedirectory will be mounted.

Line 5: **gecos** contains general information. Usually the name of the account owner.

Line 6: loginShell is what kind of shell the user will be prompted.

Line 7: uidNumber is the UID number of the user.

Line 10: shadowLastShange is the attribute which tels when the user's password was last changed.

3.4 Explanation of the package *libpam-ldap*

This is the old version of libpam-ldapd with several inconveniences. For this project there is one major inconvenience which is important to understand:

- This package has all its necessary LDAP configuration stored in **/etc/ldap.conf** which is a static file. This means that there is no possibility to add or assign variables dynamically.

3.4.1 Packet's content

This package includes:

- **auth-client-config:** A helping script which modify nsswitch and PAM configuration with predefined configurations. This is meant to help and make it easier to configure the authentication and authorization parts.
- **ldap-auth-client:** A meta package for LDAP authentication, dependent of other packages.
- **ldap-auth-config:** This is the configuration package for LDAP authentication, dependent on the meta package.
- **libnss-ldap:** This package is in general the same as libnss-ldapd, but has some issues with lookups when booting and serving host information. Also there are some issues with setuid programs (sudo, su) when using LDAP with SSL.
- **libpam-ldap:** This package is in general the same as libpam-ldapd, but has some unimportant differences for this project.

3.4.2 ldap.conf

What is ldap.conf

The main configuration file for LDAP is located in **/etc/ldap.conf** containing all the necessary parameters for making a successful connection to an LDAP-server 4.3. It's important to notice that using libpam-ldapd, the configuration file is nslcd.conf containing the same attributes as in ldap.conf, only that the syntax is a little bit different when mapping with AD (see Listing 3.2.

Mapping with AD

AD and Unix has roughly the same attributes, but uses different names, therefore the mapping process is needed.

In the ldap.conf file there are 10 commented default lines which is meant to help when setting the mapping against AD. Those lines are:

Listing 3.3: Correct mapping

```
# RFC 2307 (AD) mappings
#nss_map_objectclass posixAccount user
#nss_map_objectclass shadowAccount user
#nss_map_attribute uid sAMAccountName
#nss_map_attribute homeDirectory unixHomeDirectory
#nss_map_attribute shadowLastChange pwdLastSet
#nss_map_objectclass posixGroup group
#nss_map_attribute uniqueMember member
#pam_login_attribute sAMAccountName
#pam_filter objectclass=User
#pam_password ad
```

The syntax starts first with the mapping command followed by the Unix's attribute name, and then the AD's name of the attribute. Mapping explanation:

- **nss_map_objectclass:** Maps an objectClass which is a collection of attributes.
- **nss_map_attribute:** Maps the attribute. An attribute contain data.
- **pam_login_attribute:** The user ID attribute.
- **pam_filter:** Filters PAM for user information.
- **pam_password:** Creating unicode password and updating unicode password attribute.

3.5 nsswitch.conf

This is the configuration file located in /etc/ which is a standard file in Debian where the configuration for NSS (Name Service Switch) should be defined. It tells the operating system where information such as password, shadow, group, etc. should be gathered from.

3.5.1 Line definition

The focus here will only be on lines 3-5 (see Listing 3.4). As default, those three lines ends with **compat**, which means that the information for password, group and shadow will only be gathered from the local files in /etc/(password | group | shadow)[14].

3.5.2 Example

Listing 3.4: Default nsswitch.conf

```
# /etc/nsswitch.conf

passwd:         compat
group:          compat
shadow:         compat

hosts:     files mdns4_minimal [NOTFOUND=return] dns
networks:       files

protocols:      db files
services:       db files
ethers:         db files
rpc:            db files

netgroup:       nis
```

3.6 Unity-greeter

LightDM is a display manager and starts the sessions and the greeter (which is the login screen) [15]. This section is going to explain how the greeter works in LightDM. Ubuntu uses LightDM as its display manager and LightDM starts the unity-greeter and looks like this [4]:

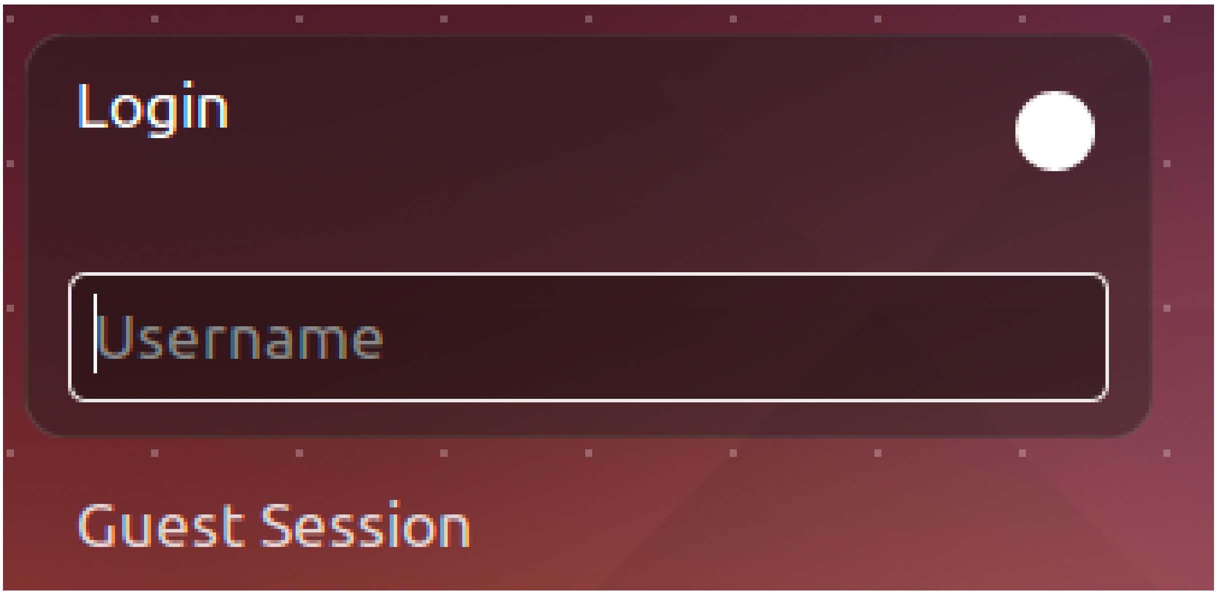


Figure 3: Unity greeter.

This greeter can be configured to look differently, but will for the most part do the same thing, which is logging in. The system configuration is in the path **/usr/share/lightdm/lightdm.conf.d/*.conf**, but to override these files a system administrator has to edit **/etc/lightdm/lightdm.conf.d/*.conf**[4].

3.6.1 Different commands

Listing 3.5: lightDM.conf | [4]

```
1 [SeatDefaults]
2 allow-guest=true/false
3 greeter-hide-users=true/false
4 greeter-show-manual-login=true/false
5 autologin-user=username
6 autologin-user-timeout=delay
7 user-session=name
8 greeter-session=name
```

Definition of 3.5:

Line 2: This allows a guest user to login.

Line 3: This will hide the user list if there are different user accounts on the host machine.

Line 4: If this is set to true, the user has to enter username and password.

Line 5: Autologin the username specified.

Line 6: Line 4 needs to be set and this will then delay the login so that the greeter will be show for that many seconds before logging in.

Line 7: Changes the default session. To change the default session you also need a .desktop file in **/usr/share/xsession/*.desktop**, where "*" is the name of the desktop session.

Line 8: Changes the default greeter, which usually is Unity in Ubuntu

3.7 Fluxbox

3.7.1 What is flux box?

Fluxbox is a window manager. It is a graphical handler for the windows generated by applications on a host. It can either be run within a desktop environment or standalone. [16][17]

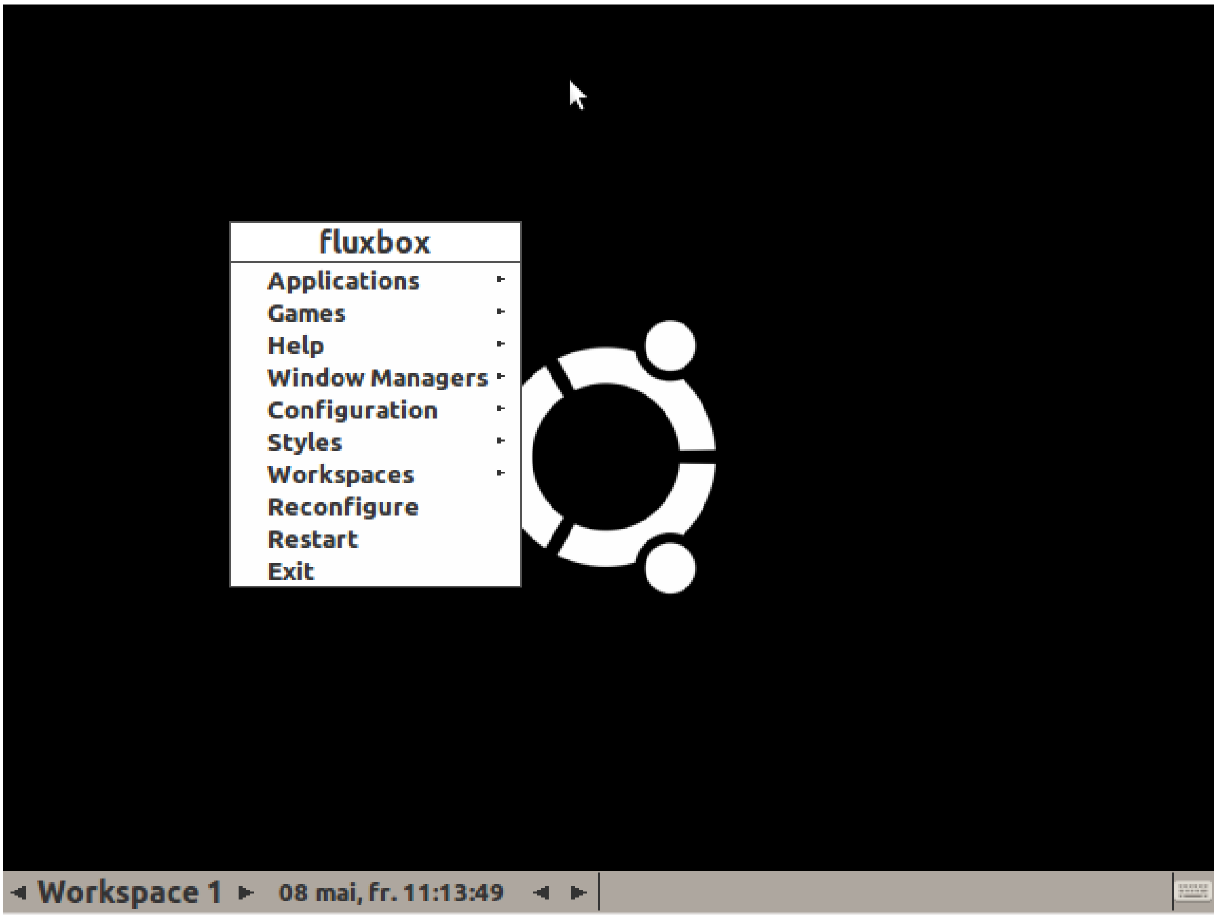


Figure 4: Fluxbox as desktop.

Fluxbox as a window manager offers a lot of functionality, not only graphical handling of the windows. Fluxbox has the possibility for different workspaces, a root menu and shortcut keys. Both the root menu and shortcut keys are highly configurable and can run very specific commands.

3.7.2 Configuration files

There are two ways of configuring fluxbox either in **/etc/X11/fluxbox** or **/home/"username"/.fluxbox**. The difference between them is that **/etc/X11/fluxbox** is for the whole system, so every new user who uses fluxbox will get this configuration, and the new user may tailor his own style and functions in fluxbox in **/home/"username"/.fluxbox**.

Contents in /etc/X11/fluxbox

1. apps
2. **fluxbox-menu**
3. fluxbox.menu-user
4. **keys**

5. menudefs.hook
6. overlay
7. system.fluxbox-menu
8. window.menu

Fluxbox-menu

In the fluxbox menu file, it is possible to configure how the standard menu should look like. In figure 4 you can see a menu that has been configured.

Listing 3.6: Fluxbox-menu syntax | [5]

```
[begin]  (name of the menu)
  [submenu] (name of the submenu)
    [exec] (name) {code}
  [end]
[end]
```

Keys

The keys file is where an administrator can edit what keys shortcut the user shall have, such as to open a terminal or even start an application of his choice.

Contents in /home/"username"/.fluxbox

This folder is very much like the one in **/etc/X11/fluxbox**, that is because the files only include the necessary files from **/etc/X11/fluxbox**, so "menu" would include "fluxbox-menu". Here is it where an administrator can make the configuration even more tailored to the specific user.

1. apps
2. **keys**
3. backgrounds (folder)
4. init
5. **menu**
6. lastwallpaper
7. overlay
8. pixmaps (folder)
9. slitlist
10. **startup**
11. styles (folder)
12. windowmeu

The **startup** file is whats start fluxbox, this is also the file where you can start applications at startup of the machine. The startup file is generated by the **/usr/bin/startfluxbox**[18].

3.8 Hypervisor

3.8.1 What is an hypervisor?

A hypervisor is a virtual machine manager which manage the host hardware to allow different operating systems on a host machine to share the same hardware [19].

There are a lot of different hypervisors on the market, just to mention some:

1. KVM
2. Xen
3. VMware
4. Virtualbox

There are also different types of hypervisors, there are type-1 hypervisor also known as bare-metal hypervisor. A type-1 hypervisor is running on the hardware itself where resources are provided by the hypervisor, while type-2 is running on the host operating systems. [20]

3.8.2 KVM

KVM (Kernel-based Virtual Machene) is a virtualization solution that turns Linux into a hypervisor. KVM itself is a hypervisor which doesn't perform any emulation, but what it does is that it provide near native performance to the guest operating system. For making this hypervisor to work with full power, processors with hardware virtualization extension is required. If the processor doesn't have the full virtualization support, KVM can still be used as a hypervisor, but then the QEMU will be required. What QEMU does is to perform as an emulator which binary translates the encoding between the hardware and the KVM. This will not let KVM to perform at full power and will speed it down, but this is a workaround.

Since KVM is a pure hypervisor, the need of an API as a management tool is important. The most widely used management tool for interacting with a hypervisor such as KVM is **libvirt**. For a user to be able to interact with the hypervisor, a user interface is needed. There are many different interfaces on the market, but the most common graphical interface used with libvirt is **Virtual Machine Manager** (best known as **virt-manager**), while the most popular command line interface is **virsh**.

KVM support different guest operating systems, such as Linux, BSD, Solaris, Windows, Haiku, ReactOS, Plan 9, AROS Research Operating system and OS X.

An explanation of used commands in this project:

- **virt-install:** Create a new container with defined attributes:
  -n: A given name to the virtual machine.
  –vcpu: How many virtual CPUs the guest OS will be able to use.
  -r: How much memory will be allocated to the guest OS.
  –disk: The path where the new .img container will be created.

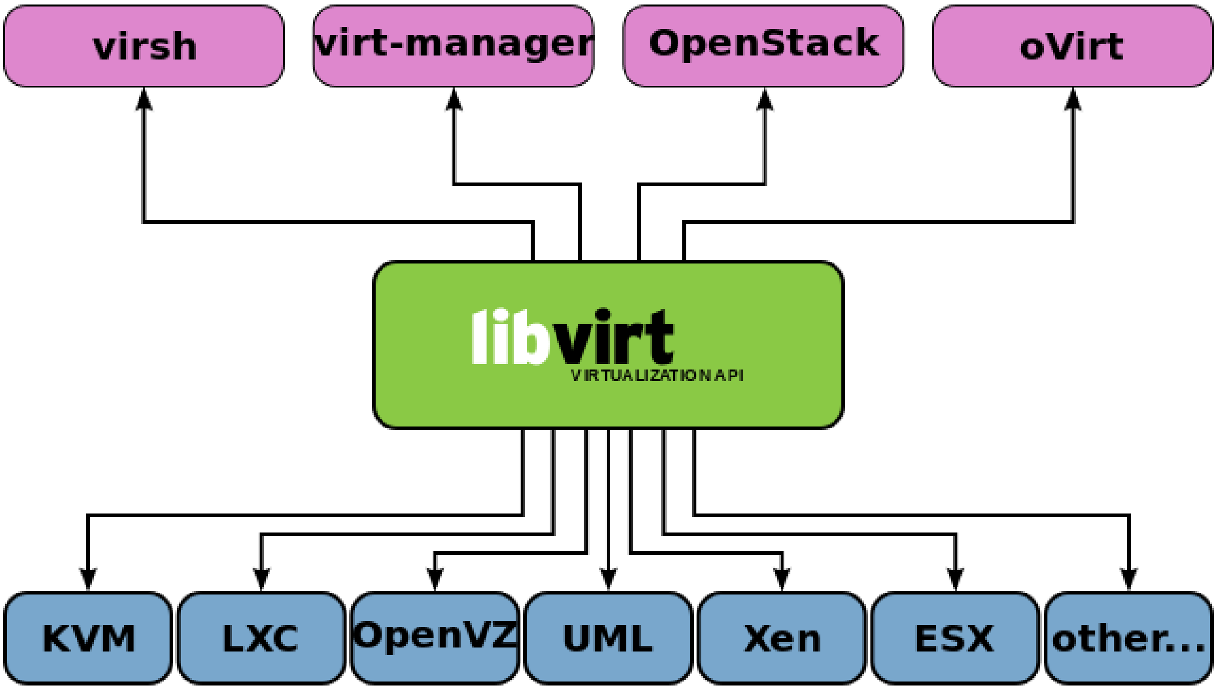


Figure 5: Hypervisor - Management tool - User interface [3]

–cdrom: What are going to be mounted in the CD-rom. Often a path to a *.iso*.
–accelerate: Use kernel acceleration capabilities.
–import: Used when an existing disk image is already made.

- **qemu-img create:** Is used when a new image is created (e.g. create a qcow2 image of a .img):
  -f: Is used to specify the format of the source image.
  -b: Defining the source image and the destination image (-b "path/to/source.img" "path/-to/destination.qcow2").

3.8.3 Virtualbox

With Virtualbox the user get a user interface where users can create virtual machines, which is easy to use and the administrator gets the possibility to make virtual machines trough CLI (VBoxManage). The administrator also has the opportunity to restrict certain options for the user interface with the CLI[21].

VBoxManage

Listing 3.7: VBoxManage commands | [6]

```
VBoxManage createvm
VBoxManage modifyvm
VBoxManage storagectl
VBoxManage storageattach
```

createvm creates a virtual machine, so it is possible to see it in the GUI.

modifyvm an administrator can do a lot of different things to modify the virtual machines that has been created.

storagectl attaches, removes or adds a storage controller.

storageattach attaches, removes or adds a media to the storage controller.

There are alot more information in the manual which is not mentioned here but these commands are essential to this bachelor thesis.

3.9 Profile.d

/etc/profile.d is a collection of scripts which runs as a user logs in. The scripts in **/etc/profile.d** helps to initialize and set up the environment [22].

3.10 Image

There are different types of images that can be used in virtualization, but in this project **.vdi** and **.img** is used. **.vdi** is an abbreviation for *Virtual Disk Image* and is usually a standard image type made by VirtualBox when a new image is created. **.img** is an image file created by KVM when a new image is created. There are also some attributes which can be set on the images, the immutable and multiattach.

3.10.1 Immutable

An immutable image acts like a normal image, but instead of storing its data on the original .vdi it makes a differencing file which it stores the data on. This differencing file gets deleted every time the virtual machine gets booted up. With this method a user can not do any damage to the original image since it stores the difference in the differencing file. [23]

3.10.2 Multi-attach

The multi-attach method works the same as immutable (section 3.10.1), but it does not delete the differencing file on boot up, so the users configuration is still the same next time the user logs back on.

3.10.3 Copy-On-Write

.qcow2 is an abbreviation for *QEMU Copy-On-Write* and is a storage created in our case by KVM(QEMU)..*qcow2* is created of images and is a storage container where all changes performed on the image will be saved. This way the original image file (.vdi) will not be changed. The *.qcow2* container can be set to immutable or multi-attach.

3.10.4 Summary

By using any of these methods mentioned in 3.10 it is possible to run multiple virtual machines on a host without taking up much space. By doing this, the only space that is used is the original images and their differencing file.

4 Implementation

4.1 Authentication

4.1.1 Prework

A connection between the host and the HiG's LDAP-server needed to be set up. We had a slow start, since we tried to authenticate against the LDAP-server **ldap.hig.no** (128.39.41.128). That server was an LDAP-server that only supported anonymous requests for student names, pictures, e-mails and some other basic attributes, but stored no sensitive information. Later, the IT-department gave us the needed parameters for establishing a connection to the right LDAP-server (recap and definitions of the attributes below, see subsection 3.3.3):

Listing 4.1: Note from the IT-department

```
hig1.hig.no (IP:128.39.140.7)
carol.hig.no (IP:128.39.140.10)
rootDN: dc=hig,dc=no
bindDN: cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no
baseDN: ou=student,dc=hig,dc=no
scope: sub
```

HiG has 3 LDAP-servers of two types that a client can authenticate against. One of them is a Linux server running OpenLDAP and the other two are Windows Servers running Active Directory (AD). From that point, we used carol.hig.no.

The first successful connection to the AD-server was made with the package *libpam-ldap* by this ldapsearch request entered in the CLI:

Listing 4.2: ldapsearch

```
~$ ldapsearch -x -h 128.39.140.10 -b ou=student,dc=hig,dc=no \
  -p 389 -D cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no \
  -W samaccountname=120683

# 120683, 12HBWUA, Student, hig.no
dn: CN=120683,OU=12HBWUA,OU=Student,DC=hig,DC=no
objectClass: top
objectClass: person
```

```
objectClass: organizationalPerson
objectClass: user
cn: 120683
sn: Maluchev
description: Stepan Maluchev
givenName: Stepan
distinguishedName: CN=120683,OU=12HBWUA,OU=Student,DC=hig,DC=no
instanceType: 4
whenCreated: 20120807083100.0Z
whenChanged: 20150511070720.0Z
displayName: Stepan Maluchev
uSNCreated: 12497131
memberOf: CN=hp_v2015_imt-in_ourbox,OU=Hovedprosjekter,OU=Prosjekter,DC=hig,DC
 =no
memberOf: CN=SPK-BDR,OU=Studieprogrammer,OU=Student,DC=hig,DC=no
memberOf: CN=Hovedprosjekt,OU=Hovedprosjekter,OU=Prosjekter,DC=hig,DC=no
memberOf: CN=12HBDRA,OU=12HBDRA,OU=Student,DC=hig,DC=no
memberOf: CN=12HBWUA,OU=12HBWUA,OU=Student,DC=hig,DC=no
memberOf: CN=Student,OU=Student,DC=hig,DC=no
uSNChanged: 36325840
name: 120683
objectGUID:: M6Ooih6+lUSCOEAbG1OR0w==
userAccountControl: 512
badPwdCount: 0
codePage: 0
countryCode: 0
homeDirectory: \\moa.stud.hig.no\home\120683
homeDrive: H:
badPasswordTime: 130758073772686499
lastLogoff: 0
lastLogon: 130758074239599687
pwdLastSet: 130715710708939558
primaryGroupID: 513
profilePath: \\moa.stud.hig.no\profile$\120683
objectSid:: AQUAAAAAAAUVAAAAtLfNIlKqyGgH5TsrKG4AAA==
accountExpires: 9223372036854775807
logonCount: 30
sAMAccountName: 120683
sAMAccountType: 805306368
userPrincipalName: 120683@hig.no
lockoutTime: 0
objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=inu,DC=no
```

```
51 lastLogonTimestamp: 130758016306697411
52 mail: 120683@hig.no
53 mobile: 93217041
54 msSFU30UidNumber: 48157
55 msSFU30HomeDirectory: /srv/stud/moa/home/120683
```

Definition of the ldapsearch command (see line 1-3 in listing 4.2):

- **-x:** Use the basic authentication method, no SSL or certificate involved.
- **-h:** The LDAP server.
- **-b:** The baseDN (the base node where the search will start).
- **-p:** Portnumber (default 389).
- **-D:** The user (bindDN) that are going to perform the lookup, since the AD-server doesn't support unauthorized requests.
- **-W:** Prompt the password instead of adding it in planetext with the search.
- **samaccountname:** search for the student number.

The response from the AD-server returned many attributes followed by a value. Those attributes can now be used for new queries when we start with mapping.

4.1.2 Configuring LDAP - Basics

The next step was to configure LDAP to automatically establish a connection to the AD-server and authenticate the users. LDAP's configuration file is located at **/etc/ldap.conf**, and this is how the first configuration was (explanation of the configuration, see section 3.3.3):

Listing 4.3: ldap.conf V.1

```
uri ldap://128.39.140.10
base ou=student,dc=hig,dc=no
binddn cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no
bindpw "PASSWORD"
scope sub
ldap_version 3
```

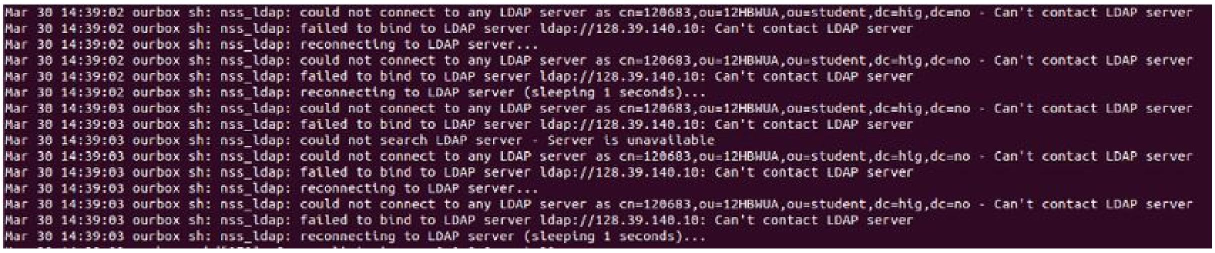
Before the LDAP/PAM could work properly, the OS needed to know that authentication would not only be done through the local files, but could also be done through the AD-server. That was done in the **/etc/nsswitch.conf** by adding "**ldap**" at the end of the 3 lines which started with *password*, *group* and *shadow* (explanation of the nsswitch.conf file, see section 3.5):

Listing 4.4: nsswitch.conf

```
# /etc/nsswitch.conf
```

```

passwd:         compat ldap
group:          compat ldap
shadow:         compat ldap

hosts:      files mdns4_minimal [NOTFOUND=return] dns
networks:       files

protocols:      db files
services:       db files
ethers:         db files
rpc:            db files

netgroup:       nis
```

Now that the LDAP was configured with the basics and the OS knew that if the user cannot be found locally, an search through the AD-directory would also be preformed. The way we checked if the OS grants a user from the AD-server was through a TTY. When e.g. the username *120683* and the password was entered, the login prompted that this user or password was wrong. To understand and locate what went wrong, we used the authentication log which is to be find in **/var/log/auth.log**:



```
Mar 30 14:39:02 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:02 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:02 ourbox sh: nss_ldap: reconnecting to LDAP server...
Mar 30 14:39:02 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:02 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:02 ourbox sh: nss_ldap: reconnecting to LDAP server (sleeping 1 seconds)...
Mar 30 14:39:03 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: could not search LDAP server - Server is unavailable
Mar 30 14:39:03 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: reconnecting to LDAP server...
Mar 30 14:39:03 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: reconnecting to LDAP server (sleeping 1 seconds)...
```

Figure 6: Binding error

The error log showed it couldn’t bind to the LDAP-server and that the bindDN didn’t work either. This was very confusing to understand why the server couldn’t be reached, cause the bindDN and password was correct, and an ldapsearch worked fine in the CLI.

After some extra research we managed to find some interesting parameters that haven’t been added to the /etc/ldap.conf file. One of the parameters was **bind_policy soft** and was referring to how PAM connects to the LDAP-server. As default, the value of **bind_policy** is set to **hard**, which means that if it fails, it will retry connecting to the LDAP-server with a wait in between with the same credentials multiple times. This would in practice not work for us. The reason was that when it tried to bind to the server the first time, it used root's credentials and failed. Since *hard* cannot unbind from the first credentials and then try with another credentials, PAM never got to use our bindDN credential, even if the error message said that it tried our bindDN. With this value set to **soft**, the host have the ability to unbind from the credentials it failed with and try again with another credentials. This solved the binding problem, but still the user couldn't be granted permission to login.

4.1.3 Configuring LDAP - Mapping

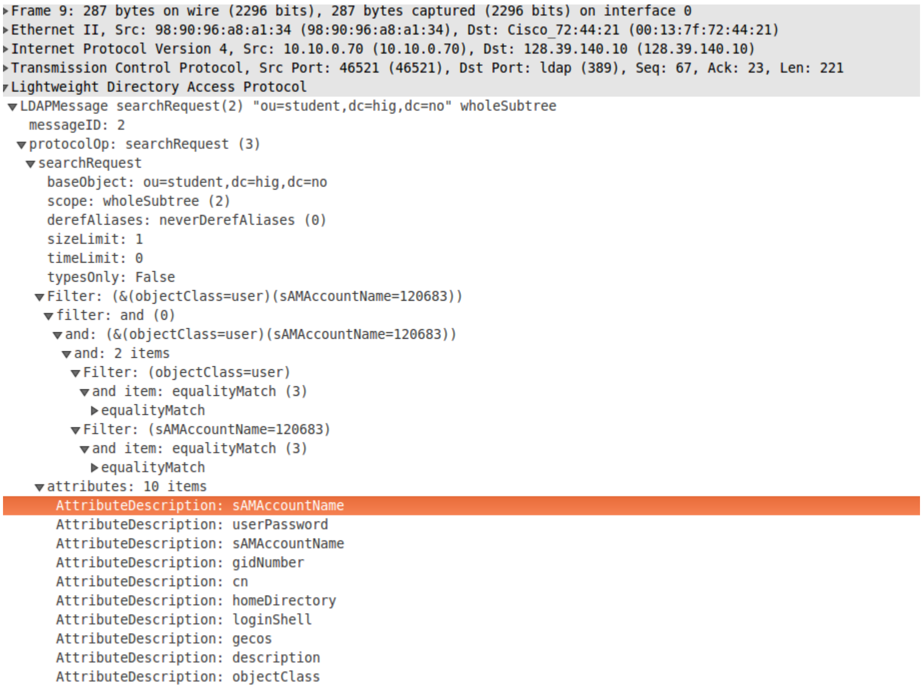
For mapping attributes between Unix and AD (see definition of the attributes in section 3.4.2). If the mapping is done correctly and LDAP is configured correct, all users in the AD will be displayed via this command (see Listings 4.5).

Listing 4.5: getent command

```
getent passwd
```

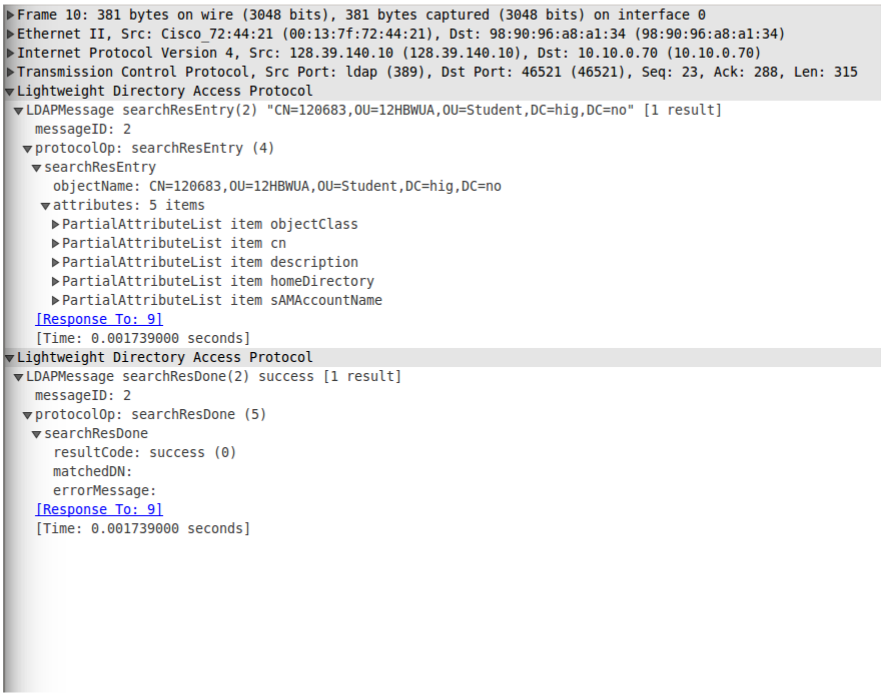
The syntax of the output will first display the users from the local /etc/passwd and then it search through AD and displays those users. We used Wireshark for identifying packets and checking what attributes our host asks for and what the AD-server returns, this way we would get a better understanding of what really happens and what attributes is needed to be correctly mapped.

In Figure 7 we can see that the host asked the AD-server for 10 attributes, but only got 5 of them(see Figure 8).



```
▸Frame 9: 287 bytes on wire (2296 bits), 287 bytes captured (2296 bits) on interface 0
▸Ethernet II, Src: 98:90:96:a8:a1:34 (98:90:96:a8:a1:34), Dst: Cisco_72:44:21 (00:13:7f:72:44:21)
▸Internet Protocol Version 4, Src: 10.10.0.70 (10.10.0.70), Dst: 128.39.140.10 (128.39.140.10)
▸Transmission Control Protocol, Src Port: 46521 (46521), Dst Port: ldap (389), Seq: 67, Ack: 23, Len: 221
▾Lightweight Directory Access Protocol
 ▾LDAPMessage searchRequest(2) "ou=student,dc=hig,dc=no" wholeSubtree
    messageID: 2
  ▾protocolOp: searchRequest (3)
   ▾searchRequest
      baseObject: ou=student,dc=hig,dc=no
      scope: wholeSubtree (2)
      derefAliases: neverDerefAliases (0)
      sizeLimit: 1
      timeLimit: 0
      typesOnly: False
    ▾Filter: (&(objectClass=user)(sAMAccountName=120683))
     ▾filter: and (0)
      ▾and: (&(objectClass=user)(sAMAccountName=120683))
       ▾and: 2 items
        ▾Filter: (objectClass=user)
         ▾and item: equalityMatch (3)
          ▸equalityMatch
        ▾Filter: (sAMAccountName=120683)
         ▾and item: equalityMatch (3)
          ▸equalityMatch
    ▾attributes: 10 items
       AttributeDescription: sAMAccountName
       AttributeDescription: userPassword
       AttributeDescription: sAMAccountName
       AttributeDescription: gidNumber
       AttributeDescription: cn
       AttributeDescription: homeDirectory
       AttributeDescription: loginShell
       AttributeDescription: gecos
       AttributeDescription: description
       AttributeDescription: objectClass
```

Figure 7: Wireshark, Host sent



```
▸Frame 10: 381 bytes on wire (3048 bits), 381 bytes captured (3048 bits) on interface 0
▸Ethernet II, Src: Cisco_72:44:21 (00:13:7f:72:44:21), Dst: 98:90:96:a8:a1:34 (98:90:96:a8:a1:34)
▸Internet Protocol Version 4, Src: 128.39.140.10 (128.39.140.10), Dst: 10.10.0.70 (10.10.0.70)
▸Transmission Control Protocol, Src Port: ldap (389), Dst Port: 46521 (46521), Seq: 23, Ack: 288, Len: 315
▾Lightweight Directory Access Protocol
 ▾LDAPMessage searchResEntry(2) "CN=120683,OU=12HBWUA,OU=Student,DC=hig,DC=no" [1 result]
    messageID: 2
  ▾protocolOp: searchResEntry (4)
   ▾searchResEntry
      objectName: CN=120683,OU=12HBWUA,OU=Student,DC=hig,DC=no
    ▾attributes: 5 items
     ▸PartialAttributeList item objectClass
     ▸PartialAttributeList item cn
     ▸PartialAttributeList item description
     ▸PartialAttributeList item homeDirectory
     ▸PartialAttributeList item sAMAccountName
    [Response To: 9]
    [Time: 0.001739000 seconds]
▾Lightweight Directory Access Protocol
 ▾LDAPMessage searchResDone(2) success [1 result]
    messageID: 2
  ▾protocolOp: searchResDone (5)
   ▾searchResDone
      resultCode: success (0)
      matchedDN:
      errorMessage:
    [Response To: 9]
    [Time: 0.001739000 seconds]
```

Figure 8: Wireshark, Host received

Through analyzing the Wireshark we solved the mapping issue with this (see Listings 4.6):

Listing 4.6: Mapping with AD- ldap.conf

```
nss_map_attribute gidNumber primaryGroupID
nss_map_attribute uidNumber sAMAccountName
nss_override_attribute_value loginShell /bin/bash
nss_map_attribute gecos description
nss_map_attribute homeDirectory unixHomeDirectory
nss_map_objectclass posixAccount user
nss_map_attribute uid sAMAccountName
nss_map_attribute shadowLastChange pwdLastSet
nss_map_objectclass posixGroup group
nss_map_attribute uniqueMember member
pam_login_attribute sAMAccountName
pam_filter objectclass=User
pam_password ad
```

Finally the *getent passwd* returned not only the local users, but also all the students in the AD. For example a search for *121088* returned first the UID,*,the groups, name, mounting point and shell:

Listing 4.7: getent passwd 121088

```
121088:*:121088:513:Gavin Thomas Garrad:\\moa.stud.hig.no\home\121088:/bin/
    bash
```

4.1.4 Configuring LDAP - Mapping mounting directory

Now that the loginshell has been specified to be **/bin/bash**, which is the default CLI, the user got an empty shell containing nothing inside. The user must be mounted in the local **/home/** directory with its own username and not the remote place as AD-server says. Therefor the home directory attribute needed to be changed. First we tried overriding the attribute like this:

Listing 4.8: getent passwd 121088

```
nss_override_attribute_value homeDirectory /home/121088
```

This gave the user a shell in the specified directory, but if another user tried to log in, they too will be granted to the same directory.

What we needed was to have a dynamic configuration file where we could use a variable to store the username. This way every user would get its own home directory. Since the package **libpam-ldap** (see section 3.4) had its main configuration file for LDAP in a non-dynamic file, we upgraded the package to **libpam-ldapd** (see section 3.3) . This package used **/etc/nslcd.conf** as its main configuration file.

Listing 4.9: Mapping with AD- nslcd.conf

```
1  filter passwd (&(objectClass=user))
2  map passwd uid              sAMAccountName
3  map passwd gidNumber        "125"
4  map passwd homeDirectory    "/home/$sAMAccountName"
5  map passwd gecos            description
6  map passwd loginShell       "/bin/bash"
7  map passwd uidNumber        msSFU30UidNumber
8  filter shadow (&(objectClass=user))
9  map shadow uid              sAMAccountName
10 map shadow shadowLastChange pwdLastSet
```

Definition of attributes used in Listing 4.9 is defined in section 3.3.3:

Line 1, 8: This sets the *passwd/shadow* filter attribute to **user** (default this is set to *postixAccount*).

Line 2, 9: Sets the uid which is named in the AD as **sAMAccountName**.

Line 3: Sets the gidNumber to be **125** which is the GID for *vboxusers*. To be able to use the extension pack on Virtualbox, the user must be a member of this group (see section 4.4.2).

Line 4: Sets the home directory to be **/home/"studentnumber"**.

Line 5: The AD has an attribute named **description** which contains the full name of the user.

Line 6: There is no loginShell attribute on the AD, so we need to set it here.

Line 7: AD has an attribute named **msSFU30UidNumber** which is the user number.

Line 10: The AD's name of this attribute is **pwdLastSet**.

The syntax of the mapping was just a little bit different from the ldap.conf (see Listing **??**), but this solved the mounting problem.

4.1.5 Home directory with a desktop environment

The last problem here is when a user logs in, they will be prompted to its home directory, but the directory contains absolutely nothing. The solution was to create a new desktop environment which contains a *Desktop* folder, *Downloads* folder etc. for every new user. This could be done in the **/etc/pam.d/common-session** where we had to add this line:

Listing 4.10: Make a home directory

```
session required pam_mkhomedir.so skel=/etc/skel/ umask=0077
```

Since the definition for the *common-session* is:

> *"Allocates the resources that a user might need during a login session, for example, mounting the user's home directory, setting resource usage limits, printing a message of the day, etc."* | [1]

We require to use the **pam_mkhomedir.so** module which is a module that gives users new default home directory. The **/etc/skel** is a directory containing all the necessary files and directories that will be copied automatically over to the new user's home directory. The **umask=0077** sets the privileges for the users, which in this case will prevent users from entering others home directories (use 0022 for giving the users more access).

4.2 Desktop Environment

4.2.1 Unity-greeter

The original greeter will show a list of users that has already logged in. What we want is when a user turns on a machine the user must be able to enter his/hers student number and password and not see any list of other users. How we did this is by adding a file in **/etc/lightdm/**, **lightdm.conf** and insert the lines:

Listing 4.11: lightdm.conf

```
[SeatDefaults]
greeter-hide-users=true
greeter-show-manual-login=true
allow-guest=false
```

4.2.2 Window manager

Profile.d

We tried to launch Fluxbox in **/etc/profile.d** (see section 3.9) inside the script which setup the virtual machines for Virtualbox. Which was a partial success, since it was possible to get the window manger up and running, but it was not able to have any function at all, but the virtual machines worked.

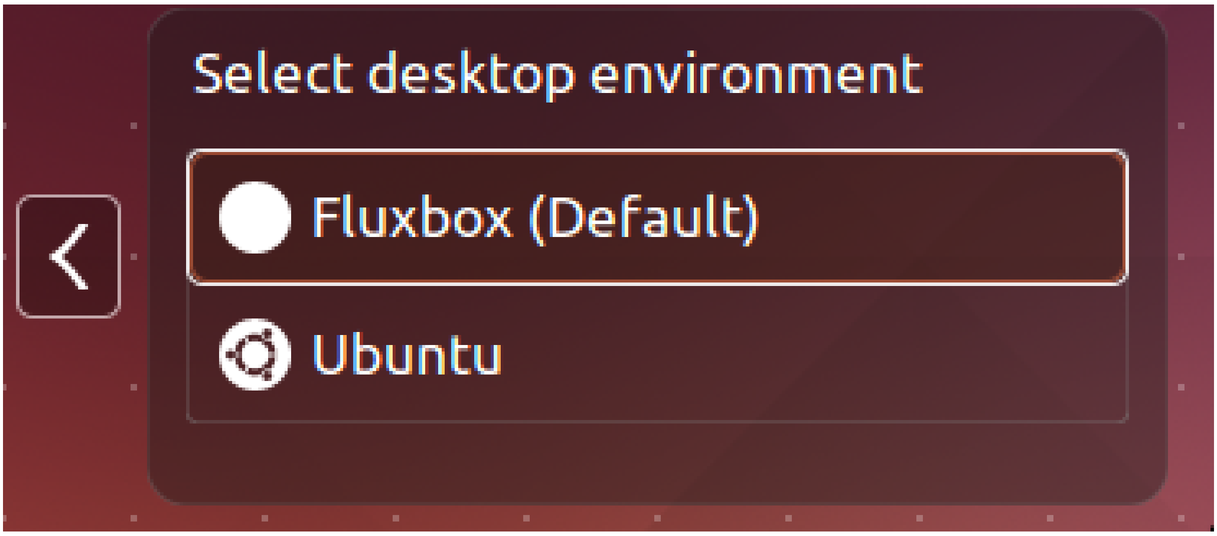
Sessions

Since the script inside **/etc/profile.d** didn't fully work, we had to take a look at sessions (see Listing 3.5). We found out that we could change the default sessions after we had installed Fluxbox. To get this to work, we inserted:

Listing 4.12: Default session

```
[SeatDefaults]
user-session=fluxbox
```

in **/usr/share/lightdm/lightdm.conf.d/50-ubuntu.conf**, the result of this is Figure 9.



Select desktop environment

Fluxbox (Default)

Ubuntu

Figure 9: Changed the default session.

Configuration of Fluxbox

We needed to restrict what the user could do (see section 3.7.2), the way of doing that was to change the config in **/etc/X11/fluxbox**:

Listing 4.13: Fluxbox changed

```
# This is an automatically generated file.
# Please see <file:/usr/share/doc/menu/README> for
    information.

# to use your own menu, copy this to ~/.fluxbox/menu, then
    edit
# ~/.fluxbox/init and change the session.menuFile path to ~/.
    fluxbox/menu

[begin] (Fluxbox)

# Automatically generated file. Do not edit (see /usr/share/
    doc/menu/html/index.html)

   [submenu] (Applications) {}
```

```
      [submenu] (VirtualBox) {}
         [exec] (VirtualBox) {/usr/bin/virtualbox} </usr/share/pixmaps/virtualbox.xpm>
      [end]
   [end]
   [restart] (Restart)
   [exit] (Exit)

[end]
```

We also had to change the keys file, where we removed the option for opening a terminal and a dialog for running other programs.

Listing 4.14: Removed code in Keys

```
# open a terminal
Mod1 F1 :Exec x-terminal-emulator
# open a dialog to run programs
Mod1 F2 :Exec fbrun
```

The administrators Fluxbox includes the same files as a ordinary user, but there are two more lines in the fluxbox-menu of the administrator **/home/"username"/.fluxbox**:

Listing 4.15: Fluxbox administrator

```
1 [begin] (fluxbox)
2 [include] (/etc/X11/fluxbox/fluxbox-menu)
3 [submenu] (Switch Environment)
4         [exec] (Unity) {gnome-terminal -x sudo  mv /usr/share/ubuntu.desktop /usr/share/xsessions/ }
5         [exec] (Fluxbox) {gnome-terminal -x sudo mv /usr/share/xsessions/ubuntu.desktop /usr/share/ }
6 [end]
7 [end]
```

Line 4 Moves ubuntu.desktop back to its original folder(**/usr/share/xsessions**). What this means is that the administrator has the opportunity to go back to regular Ubuntu (Unity), so that the administrator gets a familiar desktop to work with when he needs it.

Line 5 What this does is move the **ubuntu.desktop** from the **/usr/share/xsessions/ubuntu.desktop** to another location, what this does is that when a user logs onto the machine the user will not have a choice between the two different desktop environments (see Figure9).

Startup file

As default Virtualbox also needs to start when the user logs into the system. That is where the startup file comes in, as mentioned in section 3.7.2 that the startup file is generated by the **/usr/bin/startfluxbox**. We had to edit this file to start Virtualbox in our installation script:

Listing 4.16: Insert into startfluxbox

```
sed  -i 's/exec fluxbox/ exec virtualbox\&\n exec fluxbox/g'
   /usr/bin/startfluxbox;
```

This replaces the line in startfluxbox where it says **exec fluxbox** with **exec virtualbox& exec fluxbox**. Where the "&" means, run in the background.

4.3 TTY

TTY consoles or Virtual consoles, needs to be removed from the **/etc/init/** folder so that the user cannot enter any CLI. We accomplished this by moving the **/etc/init/tty*.conf** files with our installation script by adding:

Listing 4.17: Move TTY consoles

```
mv /etc/init/tty1.conf /home/"username"/Desktop
mv /etc/init/tty2.conf /home/"username"/Desktop
mv /etc/init/tty3.conf /home/"username"/Desktop
mv /etc/init/tty4.conf /home/"username"/Desktop
mv /etc/init/tty5.conf /home/"username"/Desktop
mv /etc/init/tty6.conf /home/"username"/Desktop
```

Where **"Username"** is the name of the administrator on the host machine.

4.4 Hypervisor

Our employer wanted it to cost as little as possible, so we where recommended by our tutor and others to look at KVM and Virtualbox.

4.4.1 KVM implementation

First we installed the **KVM** package, **libvirt** and the desktop user interface **virt-manager**:

Listing 4.18: Install KVM

```
sudo apt-get install kvm libvirt-bin virt-manager
```

Next we created an .img from a .iso file. We set the **rx** permission to the .img file because users must be able to execute and read the file, to be able to create a personal .qcow2 image.

Listing 4.19: Create a .img

```
virt-install -n ubuntu --vcpu=1 -r 1024   --disk path=/home/ourbox/Desktop/libvirt/ubuntu.img,size=8 --cdrom /home/ourbox/Desktop/os/ubuntu-14.04.2-desktop-amd64.iso  --accelerate
sudo chmod 755 /home/ourbox/Desktop/libvirt/ubuntu.img
```

Now a Ubuntu image was created and was read and executable of everyone, but only ourbox (admin) could write to it. In the /etc/profile.d/ we created a script that created 5 .qcow2 images:

Listing 4.20: defaultLibvirtFile.sh

```
 1 #!/bin/bash
 2
 3 me=$(whoami);
 4 user="";
 5
 6 if [[ $me != $user ]]; then
 7   if [[ ! -d ~/VirtualMachines ]]; then
 8     mkdir ~/VirtualMachines;
 9   fi
10   for i in {1..5}; do
11     if [[ ! -f ~/VirtualMachines/ubuntu$i.qcow2 ]]; then
12       qemu-img create -f qcow2 -b /home/ourbox/Desktop/libvirt/ubuntu.img ~/VirtualMachines/ubuntu$i.qcow2;
13       if [[ $i <= 1 ]]; then
14         virt-install -n ubuntu$i --connect qemu:///session --vcpu=1 -r 1024 --disk path=~/VirtualMachines/ubuntu$i.qcow2,format=qcow2 --import --accelerate;
15         virsh --connect qemu:///session destroy ubuntu$i;
16       fi
17     fi
18   done
19 fi
```

Line definition:

Line 3 This script is executed as the user (not as root), therefore the username is stored in a variable.

Line 4 The administrator username is stored in its own variable.

Line 6 If the user is anyone else besides the ourbox (admin), then enter the if-statement.

Line 7 Checks if the default virtual machines folder exists. The virtual machines will be stored in this folder.

Line 8 If the folder doesn't exists, then create a new one in the user's home directory.

Line 10 In this example we create a loop which will be executed 5 times and create virtual machine containers.

Line 11 Checks if the .qcow2 image exists, if not then enter the statement.

Line 12 Creating a .qcow2 image of the original *ubuntu.img* which is found in the ourbox's home directory. The loop will create 5 .qcow images named *ubuntu1.qcow2*, *ubuntu2.qcow2* ,etc.

Line 13 Enter the statement only the first time the for-loop runs. This statement only creates one default mapped VM that will automatically be displayed when the user logs in.

Line 14 Create a virtual image with the specific attributes. This image will be visible for the user in the graphical user interface virt-manager. Since this is a user which isn't running this command as root, the user needs to start user session with the qemu/hypervisor.

Line 15 Line 14 created a new virtual image and automatically started the machine. If many virtual machines will start simultaneously, the host will run slow. Therefor we force powering the virtual image off.

The step above is also possible to do in the virt-manager (GUI). The problem is that this is not straightforward. (This is not a problem if the qcow2 file is in the folder /VirtualMachines). To make a user be able to do this, we can follow this step by step guide [24]:
Step 1: Create a new VM from [Import existing disk image] option.
Step 2: Choose your qcow2 image.
Step 3: Select [Customize configuration before install] before you go forward (note: without this step, you will got a non-bootable message from the virtual machine).
Step 4: Identify the qcow2 format to the image and begin installation [Disk1] – [Advanced options] –[Storage format]: qcow2 – [Begin Installation]

4.4.2 Virtualbox implementation

Installation of Virtualbox

There are several versions of Virtualbox and extension packs which gives Virtualbox different features. We needed to be able to bridge the USB-ports from the host machine to the virtual machine. To accomplish this we had to install an extension pack on the host machine. Every version of Virtualbox has its own extension pack, whereof the newest releases has a delay on the extension pack. This means that we had to install a specific version of Virtualbox and the correct extension pack to it. The result is the following script:

Listing 4.21: Installation of Virtualbox-4.3

```
1 #!/bin/bash
2 admin="";
3 sudo echo 'deb http://download.virtualbox.org/virtualbox/debian trusty
      contrib' >> /etc/apt/sources.list;
4 sudo wget -q https://www.virtualbox.org/download/oracle_vbox.asc -O- | sudo
       apt-key add -;
5 sudo apt-get update;
6 sudo apt-get install -y virtualbox-4.3;
7 wget -O /tmp/extention.vbox-extpack http://dlc-cdn.sun.com/virtualbox
      /4.3.26/Oracle_VM_VirtualBox_Extension_Pack-4.3.26-98988.vbox-extpack;
8 sudo VBoxManage extpack install /tmp/extention.vbox-extpack;
9 sudo adduser $admin vboxusers;
```

Line definition:

Line 2: Variable which tells the name of the administrator user. This will get a value when the script is being executed.

Line 3: Get access to the repository where Virtualbox-4.3 will be downloaded from.

Line 4: Download the Oracle public key for apt-secure and automatically add the key.

Line 5: Update the host with the new key.

Line 6: Install Virtualbox 4.3.

Line 7: Download the extension pack for Virtualbox 4.3.26 into /tmp/ (after a reboot the downloaded extension file will be removed automatically).

Line 8: Make Virtualbox install the extension pack.

Line 9: The user must be added to the "vboxusers" group to get full use of the pack.

Configuration

We made a script which went into the profile.d folder, which is meant to be run every time a user logs on to the system. Where it made a virtual machine, from the .vdi image that was on the administrators desktop. The problem we encountered was the restrictions of the .vdi image, what was done to fix the problem was **chmod 755** the .vdi file and add the .vdi image to our test group.

Listing 4.22: DefaultVMs.sh first edition

```
1  #!/bin/bash
2  os='Ubuntu';
3  user=$(whoami);
4  DIRECTORY=~/VirtualBox\ VMs/$os;
5  if [ $user != 'ourbox']; then
6    osRegEx ='^\'${os}\'';
7    knownVMs=$(VBoxManage list vms | grep -e '$osRegEx'| awk '{print $1}' | cut -d '\'' -f 2);
8    if [[ $knownVMs != $os ]]; then
9      if [[ -d $DIRECTORY ]];then
10       rm -r 'DIRECTORY';
11     fi
12     VBoxManage create --name $os --ostype Ubuntu_64 --register 2>&1;
13     VBoxManage modifyvm $os --memory 1024 2>&1;
14     VBoxManage modifyvm $os --vram 256 2>&1;
15     VBoxManage modifyvm $os --cpus 2 2>&1;
16     VBoxManage storagectl $os --name sata1 --add sata  2>&1;
17     VBoxManage storageattach $os --storagectl sata1 --port 0 --device 0 --type hdd --medium /home/ourbox/Desktop/Ubuntu.vdi --mtype immutable 2>&1;
18    fi
19 fi
```

Definition of 4.22:

Line 1-4: These are different variables that were used in the script. "DIRECTORY" is for the differential files.

Line 5: If the user who logs in isn't outbox then it makes the virtual machines

Line 6: This is a regular expression for getting the name "Ubuntu" out of the VBoxManage list.

Line 7: Gets the known VM which in this case will be "Ubuntu" as stated in line 2.

Line 8: Then it checks if the "knownVMs" is the same as "os" if they are not, then line 10-11 happens.

Line 9-10: Here it checks if there is a directory to the "os", since it may have been deleted in Virtualbox, but the user might not have deleted everything, so the script guarantee that everything is deleted.

Line 12: Now that the virtual image is created, this can be seen in the GUI, but you will not yet be able to start the machine before line 18-19 has run.

Line 13 - 15: These lines modify the virtual machine which has just been made, and it tells how much ram, cpu and video ram it should have.

Line 16: This modifies the storages it should use.

Line 17: This attaches it to that storage which was just created.

4.5 Result

4.5.1 Conclusion of the KVM

In advance we have to create $n * .qcow2$ files in the " /VirtualMachines/" for each user. It is these .qcow2 files the user can create virtual machines. The qcow2 file must be created first, which is at minimum ca. 192K. The problem here is:

- We have to pre-define how many VMs a user can create.
- 192K * (numb. of the particular OS a user can create) * (numb. of different OSes) * (numb. of users on the system) = Too much unused and occupied space.
- When the user choose a virtual machine, e.g. Ubuntu1.qcow2, this file will now be occupied. Next time the user want to choose and create a new Ubuntu virtual machine, the Ubuntu1.qcow2 will still be displayed, but at the end, when the user presses âĂIJfinishâĂİ with the installation of the new virtual machine, an error will occur and say that this .qcow2 file is already used by another virtual machine. This can easily confuse the user, and he/she must remember and have an overview of what kind of files is used and not.

4.5.2 Conclusion of the Virtualbox

DefaultVM.sh final

As mentioned in section 4.4.2 it is needed to change the permissions on the .vdi images. To make it easier for the administrator, now all he as to do is to have a folder which is named "os" on the administrators desktop. Then Listing 4.23 will take care of the permissions of every image inside the folder. It will also change the permission for the "os" folder. The script runs only one time and that is at installation, or if there are new .vdi images.

Listing 4.23: verifyNewOSes.sh

```
#!/bin/bash

admin="";

chmod 755 /home/$admin/Desktop/os;

for file in $(ls -l /home/$admin/Desktop/os/*.vdi | awk ’{ print $9}’)
do
        chmod 755 $file;
done
```

Listing 4.24: DefaultVMScript.sh

```
#!/bin/bash

admin="";
me=$(whoami);
os="";

for oses in $(ls -l /home/$admin/Desktop/os/ | awk ’{print $9}’ | grep ".vdi" | cut -d "." -f 1)
do
  os=’$os $oses’;
done

if [[ $me != $admin ]];then

```

```
14   for eachOS in $os;do
15     osRegEx="^\"${eachOS}\"";
16     knownVMs=$(VBoxManage list vms | grep -e "$osRegEx" | awk
         ’{print $1}’ | cut -d "\’’" -f 2);
17     DIRECTORY=~/VirtualBox\ VMs/$eachOS;
18
19     if [[ $knownVMs != $eachOS ]];then
20       if [[ -d  $DIRECTORY ]]; then
21         rm -fr "$DIRECTORY";
22       fi
23       VBoxManage createvm --name $eachOS --ostype Ubuntu_64
           --register;
24       VBoxManage modifyvm $eachOS --memory 1024 --vram 256 --
           cpus 2 --nic1 bridged --bridgeadapter1 eth1;
25       VBoxManage storagectl $eachOS --name sata1 --add sata;
26       VBoxManage storageattach $eachOS --storagectl sata1 --
           port 0 --device 0 --type hdd --medium /home/$admin/
           Desktop/os/$eachOS.vdi  --mtype immutable;
27     fi
28   done
29 fi
```

This script (Listing 4.24) does the same as (Listing4.22), but with some modifications. The script (Listing 4.24) is a script which resides in the **profile.d** folder, and runs every time a user logs on the machine.

Line 7 This for loop gets all the names of the .vdi images in the folder "os" which is on the desktop of the administrator

Line 9 Puts the .vdi images name into an array which is later used in line 14.

Line 14 Goes through the array made in line 9, and makes a VM.

Line 24 Here it modify the virtual machine the same way it did in (Listing 4.22), but it adds the "–nic1 bridged –bridgeadapater1 eth1" which sets the NIC into bridge mode.

5 Ending

5.1 Requirements and their Results

Completed	**Requirement**	**Summery**
Partially	If possible no OS will be running on the hosts, only a hypervisor which boots up.	This is possible with KVM, since KVM turns Linux into a hypervisor. We also tested Virtualbox, what we found was that KVM was not that user friendly and that Virtualbox was a more widely known software for virtual machines. We had to make a choice which of them we where going to use and went for Virtualbox, but then we had to run it on an OS.
Yes	No access to the hypervisor by the users.	Since we went for an OS which runs Virtualbox, we used a Window manager (Fluxbox) and disabled all functions besides starting Virtualbox and exiting. We mainly do not want the user to have any access to a terminal, so we disabled the TTY consoles (or also known as Virtual consoles).
Yes	A list of virtual machines can be start by the users.	This was possible with both hypervisors we tested. Default when a user logs into the system, they will have a list of different virtual machines ready to be booted up. The user will also be able to create their own virtual machines.
Yes	The virtual machines must be able to have accounts/users.	This depends on the base image that is created. The ones we have created has only one user, which is also the administrator and will give the students administrator rights on the virtual machine.

Yes	Connect with HIG LDAP for authentication.	Since the host's OS is Ubuntu, we used LDAP/PAM to authenticate against HiG's AD-server. This was successful and gave the users the ability to be authenticated with their own user and password they use at HiG.
Yes	Authorised by a local entity (cisco lab).	The user will be able to enter their credentials on the local machine and access the system.
No	As a Guest user.	Not at this time, and it does not exist any guest session. This will have to be implemented in a later project.
Partially	Groups has to exists.	At the moment users will automatically be assigned to group *vboxusers* (GID=125) which will let them have advantages of the extension pack which is installed with Virtualbox.
No	Each group contains of one or more users.	This requirement was not implemented in the thesis.
No	They share one folder, where the VMs for the group will be stored.	This requirement was not implemented in the thesis.
No	Each user VM will be store in the group folder.	This requirement was not implemented in the thesis.
Yes	Administrators will have full permission to change configuration settings on the hypervisor.	There is only one local administrator on the host machine. The administrator will need to switch the desktop environment before configuration of either the hypervisor or the host.
Yes	Students should be able to choose which NIC the virtual machines is going to use. Virtual machines shall have access to the hardware.	Both hypervisors (KVM and VirtualBox) will be able to satisfy the ability of choosing the physical NIC. This is possible as default through settings of the virtual machine.
Partially	Students will be able to choose how much memory, etc. the virtual machines are going to use. Within defined limits.	Both hypervisors (KVM and VirtualBox) will be able to let the user choose how much memory, CPU, etc. each virtual machine will be able to use (we haven't looked at how the hypervisor can create limits for the users).

Yes	The student will not be able to make changes on the host, only make changes to running VM's	Student will only have administration rights on their own virtual machines, nothing else.
Yes	After a student has created a snapshot, the snapshot should be able to be stored on the host, so that later, the student can find that VM on the same host.	This works. The students are able to take snapshots on the immutable image. They are also able to delete the snapshots which makes them go back to the original immutable image.
Yes	Each virtual machines must be separated, no shared folders, etc. (SF shall be possible)	Default, both VirtualBox and KVM will have virtual machines separated from each other. It is not possible to share a folder between virtual machines.
Yes	Each host will run several virtual machines when the host is at full capacity.	This depends on what kind of specifications that are set on the virtual machine and that they do not go over the limits of the host machine. The specification of the host machine is: 32 GB memory, quad-core 3.30 GHz i5-4590 CPU and a 238 GB SSD.
No	The administrator will be able to change how many virtual machines the host can run when the host is at full capacity, but this will of course be restricted to the host's hardware itself.	The administrator is not able to set restrictions on the virtual machines.
Yes	Configuration will be possible through command line or GUI.	An Administrator will have the ability to configure the host operating system and hypervisor (VirtualBox) by changing the desktop environment to Ubuntu (Unity). From there you will have a standard desktop with a terminal.
No	A centralized management solution shall be possible.	The easiest way will be to have a management host, with Puppet installed (we have not tried to make this possible).

There are some requirements we did not implement since we did not have enough time. That is because we encountered a lot of problems with the authentication part, that took longer than expected.

With the time we had left we tried to fix the ability for a user to copy and paste from host to the VM and vica versa [25]. This seemed very unstable and did not work all the time.

Because the grouping was scheduled after the authentication part which took longer than expected, there were no time left to research and implement it. We do have a theory of how it could be implemented which is, every user gets their own group which they then administer and can add users into. We do not know if this is possible or not, but it would be ideal if it was. Since Thomas has mentioned he do not want to administer the grouping of the students. He wants a "project leader" to be able to add other students into his group and they have access to the same virtual machines.

5.2 Critic to the thesis

We tried to make the host machine as secure as we could with limiting the possibility for a user to tamper with the host machine, but since the users has access to the hardware there are limits to how secure we could make it. We have also mentioned to our employer that an authentication through LDAP might not be the best way, since this means that the host machine always needs to be connected to the network. As the user has access to the hardware this might not always be the case, but the only downside is then that they will not be able to log into the machine. We had set up a week for testing which we did not get to do since we used much of our time on the authentication.

5.3 Future work

The continuance of this project would be to make the management of system centralized and make it more secure. Our thought has been to develop a working prototype, there are still some work to do before it is finished. E.g. encrypting the data going between the host and the AD-server. This is very critical and must be improved, which in theory can be solved by adding a SSL-certificate to the host.

The "skeleton" of the system is done. An improvement might be that for the .vdi images you might want to have them on a server so that the virtual machines does a pxe boot when creating new virtual images.

Another thing will be to make it possible for guest users to use the system, since there are some problems with the guest user only being a temporary user. You will have to customize the guest session. Here might be a solution [26].

All the missing requirements in 5.1 is also something that should be considered future work.

5.4 Evaluation

5.4.1 Introduction

Working with this project has been challenging and interesting. This project was our first pick and we where lucky to get it. At first we where not sure what this task was about. We knew it had something to do with virtualization and Cisco, so that peaked our interest. As we asked Thomas more questions and learned more about what he wanted, we realized that we misunderstood what the task was about. We had big plans of what we where going to make. We thought at first he wanted to have everything centralized with a server that hosted all the virtual machines, but this was not the case. We also want to mention that there has been some issues with the language barrier and knowledge.

5.4.2 Organize

We used Google drive to store all of our work, so that if anything got lost from our PC's we would have it in the cloud. Why we choose Google drive over bitbucket is because we did not have a lot of code we where going to do. But rather documents through documentation. We wrote a log for each day that we worked.

5.4.3 Project as work flow and work distribution

We have had a fluid work distribution we have taken on responsibility for what we found interesting. We have also worked together if one of us got stuck. We have had weekly meetings with our tutor and employer, so that they know what we have done, and they could come with ideas of what we should look into next.

5.5 Conclusion

We feel that our goal has been reach with this thesis. We were able to make a prototype which works. The process of making the prototype has been an challenging task and we have learned a lot. We hope the product we made for Thomas Kimmerich will be used and satisfies most of his needs. Even though some of the requirements where not fully met. We feel that we have laid the ground work for other students to continue on the subject/thesis.

Bibliography

[1] http://www.tuxradar.com/. 2015. Getting to know pam. http://www.tuxradar.com/content/how-pam-works. [Online; accessed 14.05.15].

[2] Hunter, J. 2015. Ldap search - setting the scope parameter. http://www.idevelopment.info/data/LDAP/LDAP_Resources/SEARCH_Setting_the_SCOPE_Parameter.shtml. [Online; accessed 09.05.15].

[3] Wikipedia. 2015. Libvirt — wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Libvirt&oldid=660801299. [Online; accessed 12-May-2015].

[4] Neal-pippaluk. 2014. Lightdm. https://wiki.ubuntu.com/LightDM. [Online; accessed 27.04.15].

[5] Ramsay, J. 2015. fluxboxmenu. http://fluxbox.org/help/man-fluxbox-menu.php. [Online; accessed 29.04.15].

[6] Oracle. 2015. Vboxmanage manual. https://www.virtualbox.org/manual/ch08.html#vboxmanage-modifyvm. [Online; accessed 10.05.15].

[7] Wikipedia. 2015. 'lightweight directory access protocol. http://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol. [Online; accessed 25.04.15].

[8] Wikipedia. 2015. List of ldap softwares. http://en.wikipedia.org/wiki/List_of_LDAP_software. [Online; accessed 25.04.15].

[9] Wikipedia. 2015. Pluggable authentication module. http://en.wikipedia.org/wiki/Pluggable_authentication_module. [Online; accessed 27.04.15].

[10] Hat, R. 2015. Advantages of pam. http://www-uxsup.csx.cam.ac.uk/pub/doc/redhat/redhat8/rhl-rg-en-8.0/ch-pam.html. [Online; accessed 29.04.15].

[11] wolfric. 2015. ldap-utils. https://wiki.debian.org/LDAP/LDAPUtils. [Online; accessed 10.05.15].

[12] de Jong, A. 2015. Package: libpam-ldapd. https://packages.debian.org/sid/libpam-ldapd. [Online; accessed 10.05.15].

[13] microsoft.com. 2015. Differences between ldap 2 and ldap 3. https://msdn.microsoft.com/en-us/library/aa366099%28v=vs.85%29.aspx. [Online; accessed 29.04.15].

[14] and/or its affiliates, O. C. 2015. Format of the nsswitch.conf file. http://docs.oracle.com/cd/E19683-01/817-4843/a12swit-84565/index.html. [Online; accessed 28.04.15].

[15] Archlinux. 2015. Display manager. https://wiki.archlinux.org/index.php/Display_manager. [Online; accessed 08.05.15].

[16] Archlinux. 2015. Window manager. https://wiki.archlinux.org/index.php/Window_manager. [Online; accessed 08.05.15].

[17] Dunn, S. 2015. What is a window manger? https://www.media.mit.edu/wearables/mithril/anduin/window_manager.html. [Online; accessed 10.05.15].

[18] Boetes, H. 2015. startfluxbox(1) manual page. http://www.fluxbox.org/help/man-startfluxbox.php. [Online; accessed 01.05.15].

[19] Rouse, M. 2015. hypervisor. http://searchservervirtualization.techtarget.com/definition/hypervisor. [Online; accessed 11.05.15].

[20] Vanover, R. 2015. Type 1 and type 2 hypervisors explained. https://virtualizationreview.com/blogs/everyday-virtualization/2009/06/type-1-and-type-2-hypervisors-explained.aspx. [Online; accessed 05.05.15].

[21] Oracle. 2015. Oracle vm virtualbox. https://www.virtualbox.org/manual/UserManual.html. [Online; accessed 01.05.15].

[22] linuxfromscratch.org. 2015. The bash shell startup files. http://www.linuxfromscratch.org/blfs/view/6.3/postlfs/profile.html. [Online; accessed 13.05.2015].

[23] Oracle. 2015. Chapter 5.4: Special image write. https://www.virtualbox.org/manual/ch05.html. [Online; accessed 10.05.2015].

[24] Jing. 2015. How to import a qcow2 to virtual image storage pool (gui). https://docs.google.com/document/d/1X8TaBP1v_rh8e2QXDGFmkmzoPB13Pvp2V0FQ630VCsU/edit?pli=1. [Online; accessed 12-May-2015].

[25] Kaufman, L. 2014. How to copy and paste between a virtualbox host machine and a guest machine. http://www.howtogeek.com/187535/how-to-copy-and-paste-between-a-virtualbox-host-machine-and-a-guest-machine/. [Onlne; accessed 13-May-2015].

[26] gunnarhj. 2015. Customizeguestsession. https://help.ubuntu.com/community/CustomizeGuestSession. [Onlne; accessed 13-May-2015].

A Log

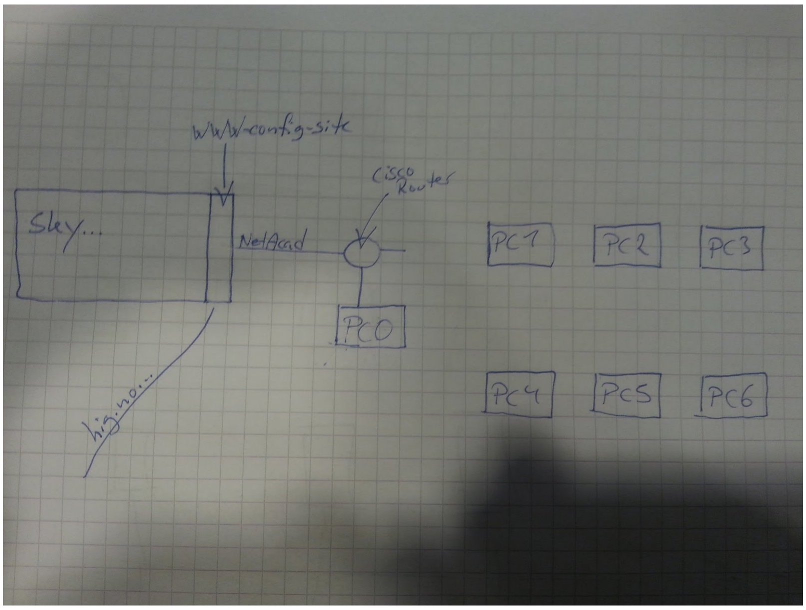
LOGG

12.01.2015:

- Vi har blitt enige om arbeids tider, noe form for utviklingsmodell(En blanding mellom inkrementell, fossefall med tidsfrister).
- Vi har sendt forespørsel til Thomas om møte
- Vi venter med å sette opp møte med Erik H, til etter at vi har vært i møte med Thomas.
- Vi skal bruke latex for selve oppgaven. Samt prosjektplan.
- Vi skal bruke Trello for en to-do list.
- Vi ønsker å bruke git bucket som sammarbeids metode.
- Vi kunne tenke oss ca 1 møte i uken, så vi har ca annenhver uke med veileder og annenhver uke med Oppdragsgiver. Hvor ett møte i måneden er mer likt ett større status møte. (Kan være at vi ønsker ett møte hver uke med veileder)
- Vi har begynt å sett på bruken av latex for å skrive prosjektplanen.
- Vi har bestemt oss for at prosjektlederen skal være Stepan.
- Vi har i “lekse” å lese pdf som er i fronter rommet ang Latex (The not so short introduction to latex) til i morgen.

13.01.2015

- Diskusjon om aktuell struktur løsninger.
- Usikkerhet om hva Thomas egentlig ønsker, men en konkret
- Vi har ikke startet med Prosjektplan enda, vi vil vite eksakt hva det er Thomas vil først før så vi begynner etter møtet med Thomas i morgen å jobbe med Prosjektplanen.
- Vi har fortsatt igjen å sette opp “Hjemmesiden våres”.



Spm:

1. Hva ønsker han av de 6 innkjøpte maskinene, er det noe elevene skal ha fysisk tilgang til.
2. Hvordan ønsker han at elevene skal ha tilgang til VMen (SSH, Remote Desktop,etc.).
3. Er det skyløsning han er ute etter?
4. Disse servicene som allerede skal være lagt opp, skal det være en virtuell maskin som da vil være inne i den evt skyløsningen?
5. Ønsker han å ha en deleløsning mellom PC0 og resten av “dummy-PCne”(PC1-6), som kommer til å foregå på en egen intern LAN?(2 forskjellige VLAN). Blir bare brukt for f.eks deling av filer (f.eks. oppgaver).

14.01.2015
Referat fra møtet med Thomas (09:30-10:15):

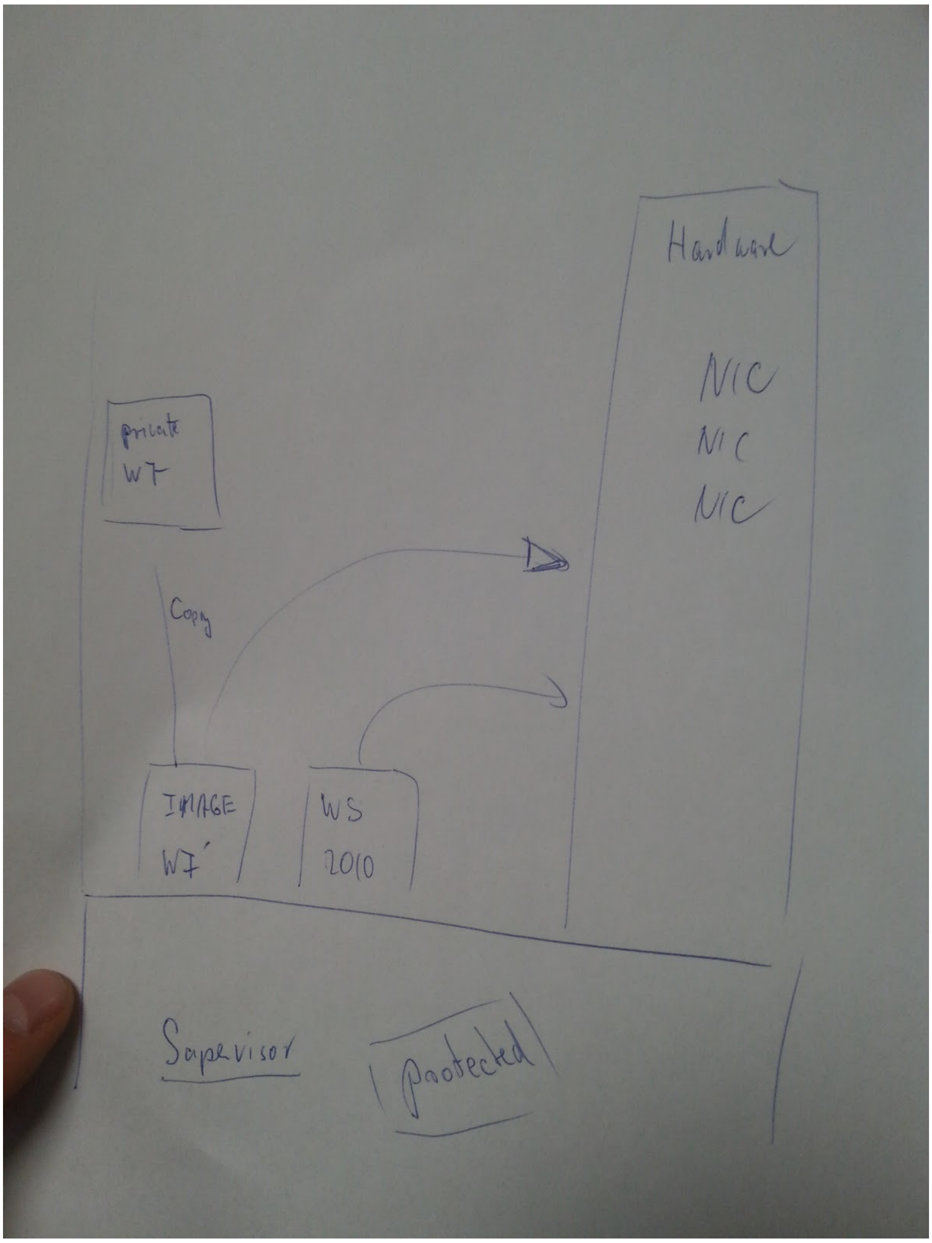
- Snapshot skal være mulig for alle elevene
- Hva elevene skal møte på nr PCen booter opp er ikke gitt, kom med forslag.
- Ikke nødvendig, men til slutt hadde det vært fint om det hadde vært muig å boote opp en VM via VPN.
- snakk med John, han har ideen bak dette. Mulighet med virtualbox.
- Alt handler om selve Supervisoren og begrensningene der.
- Søk nettet om fordeler og ulemper, tilfredstiller løsningen kravene?
- Til slutt skal det være mulig at læreren skal kunne sitte på et sted og fyre opp VMer fra en PC, dette er for å spare tid med å installere et VM fysisk på hver PC (tidsfordriv).
- Noen andre kan lage interfacet.
- Hva slags supervisor?
- hver gang eleven starter PCen, skal alt være base configurert?
- trenger ikke “host-eller-noe-slikt”, ved mindre det har større fordeler.
- På slutten av Bachelor oppgaven, skal det være alle de andre stegene som kreves for å fortsette for å få systemet 100% opp og fungerende.
- 4 february etter kl. 13 er det ikke mulig å komme inn.
- 9:30 onsdager blir det faste møter med Thomas. Uken før og etter påsken.
- skrive kravspesifisering til neste gang.

(Gavin søk: KVM ->
http://serverfault.com/questions/23738/run-virtual-machines-without-a-host)
KVM or Kernel-based Virtual Machine basically turns the Linux kernel into a hypervisor. It’s been around since 2007 and it’s awesome at using your hardwares virtualization extensions like Intel’s VT-x or AMD-V. It’s great for running multiple Operating Systems like OSX and FreeBSD and Linux and Windows all on top of this one VM infrastructure.

20.01.2015

- Worked with the specification of the system.

- We have discussed about how we are supposed to set up the hosts.
    - How the memory and cpu should be divided
        - Stepan thoughts:
            - The default value for minimum running VMs is set to 4, and f.eks. we have 100 resources. Before the first VM can be created, the resources will be divided into 4, so the new VM can at a maximum use 25% av the resources. When the first VM is created and is running, then there are 80% of the resources left. The next VM will then have access to 75/3 =25% of the resources, and so on.
            - This way, the first VM cannot be a large VM (use more then 25%), however, if the user would like to create one powerfull VM and 4 less powerfull, this can be done this way: First VM
            -
        - Gavin thoughts:
            - Have a max cpu usage and max ram usage. So that we can guarantee that you have atleast so many VM's running on the Hosts. e.g You have a bottle of soda that is 1.5 L and you want to regulate how many glass of soda you get. Lets say you want a minimum of three glasses of soda then you will get a 0.5 L glass.
            - So you can make more glass of soda but none of your soda is above 0.5 L but it can be anything between 0.1 L and 0.5 L.
            - The admin can ofcourse change this if he wants it to be 0.6L glass but then it wont be 3 glasses.
            - But the main thought is that we set the max limit for each VM so that we know how many VM we can run with full capacity on each VM.


Hardware
NIC
NIC
NIC
private
W7
Copy
IMAGE
W7´
WS
2010
Supervisor
protected

21.01.2015

Referat fra møtet med Thomas:

- Gå gjennom spec. krav og se hva slags programvarer som oppfyller hva slags krav.
- Send spec. krav til Thomas.
- Hvis noe KOSTER NOEN TUSEN KRON ER FO REN LISENS, SÅ GÅR DET GREIT. MEN ELLER NEI.
- 1-2 ukene: Lag pros/cons av programvare
  - lage timetable
  - Søke gjennom hva som finnes på markedet
  - start på sammenligning
- Vi må ta også hensyn til om servisen komme rtil å være
- slutten av april skal vi være ferdig med det fysiske, også skal vi bruke 2 uker til å skrive ferdg rapporten.
- Når vi skriver på en ting, men plutselig str vi fast, så skriv på noe annet, og fordele oppgaven.
- Før påske burde vi prøve og feil, så etter det må vi skrive en del og få ting til å funke i rapporten.
- VI må finne ut en løsning for hvordan vi skla a i bruk LDAP til authorisering.
- Authoriseringen hadde vært fint om det var mulig med med gjeste grupper, slik at folk som ikke er studenter her, får også tilgang.
- John sier (mener Thomas) at VirtualBox kan tilfredstille mange av kravene.
- I løpet av februar, så burde vi være ferdig med hvordan alt skal være, og klare til å starte med å jobbe fysisk.
- Thomas sier at vi minimum skla ha 2 NIC, men hvis flere trengs, så er ikke det et problem.
- 802.1q (Cisco standard ta no slag) se hva det innebærer, for han ønsker at NIC kommer til å ha støtte for det.

Logg:

- Vi har snakket med Erik ang møte dag, og han kan onsdager fra kl 10.15 så vi er nødt til å høre med Thomas om det er mulig for ham også. vi snakker med ham i morgen/sender mail.
- Erik nevnte at vi burde sjekk ut **Virsh** og **Libvirt** sammen med **KVM**. KVM er det som blir brukt i skyhig.
- Han mente også at denne autentiseringen og gruppering/brukere sammen med LDAP vil også bli vanskelig.
- Vi møtes i morgen for å skrive prosjektplan, vi vurderer også å jobbe i helga slik at vi får gitt den så tidlig som mulig til Erik.

22.01.2015

- Vi har laget ordentlige gruppe regler.

- Vi har begynt å skrive prosjektplan.
    - Gavin har begynt på punkt 1 av prosjektplanen “Mål og Rammer”, fortsetter med dette i morgen.
    - Stepan har begynt på punkt 2 av prosjektplanen “Omfang”, fortsetter med dette i morgen.
- Vi har skal skrive store deler av prosjektplanen i morgen.
- Vi har fått en ønsket oppdatering av Thomas på kravspesifikasjonene som vi fra før av hadde sendt til han. Han har sendt det videre til Tyskland.
- Thomas og Erik kommer nå til å ha et felles møte med oss onsdager kl. 10:15 på Thomas sitt kontor.

23.01.15

- Vi fortsatte der vi slapp i går.
    - Punkt 1, 2, 3 og 4 er snart ferdig.
- Vi har fått mail ang prosjektavtale og dette må vi få ordnet på møtet.
    - Vi må få underskrift av Thomas.

28.01.15

Referat fra møte:

- Angående 1.del innleveringen vår:
    - Målet med denne oppgaven er ikke å bygge store nettverk, men til en senere anledning skal det være mulig (Second step). Hoved poenget i dette stedet er så ha en hypervisor som er avgrenset med grupperettigheter.
    - taskdescription as bulletpoints. Frode skulle vist ha noen lynkurs i hvordan man laget kravspec.

- Det må være mulig å copy paste mellom de forskjellige VMene.
- 10.februar skal vi være ferdig med kravspec og hvordan det skal være.
- det enkleste er hvis vi bruker linux og KVM, local group policy, så kommer det policy rules bli kopiert videre til alle andre hostene.
- Erik foreslår at vi heller har egen policy grupper som blir gitt ut til elevene, passord og brukernavn.
- Erik foreslår at jo fortere jo bedre at vi prøver å sette opp det fysiske og ser hva som er mulig og ikke, og hvordan teknologien fungerer, eller hvor grensene går.
- Bridge mode mellom VMene.

30.01.15

- uname: **ourbox** pwd: **our555box**
- First we install Ubuntu 14.04.1 LTD with LXDE-Core with Virsh, via tutorial: http://www.howtogeek.com/117635/how-to-install-kvm-and-create-virtual-machines-on-ubuntu/
- Chack if the CPU is a 64 og 32 bit: ~$ uname -m
- https://help.ubuntu.com/community/KVM/Installation

- We started to surf the internett and we found out what we mainly want is a type-1 hypervisor aka bare-metal hypervisor (http://en.wikipedia.org/wiki/Hypervisor). We found one that is called Xen (http://wiki.xenproject.org/wiki/Xen_Overview#What_is_the_Xen_Project_Hypervisor.3F) Gavin started to read about this. https://www.youtube.com/watch?v=n3POmlMNzvw

- Problem:
    - When we booted up Ubuntu, we got a timeout message like this:
      [65.456035] mei_me 0000:00:03.0: reset: connect/disconnect timeout.
      Solution: Add "rmmod mei-me" into the startup fil: /etc/rc.local
      Source: https://bbs.archlinux.org/viewtopic.php?id=168403

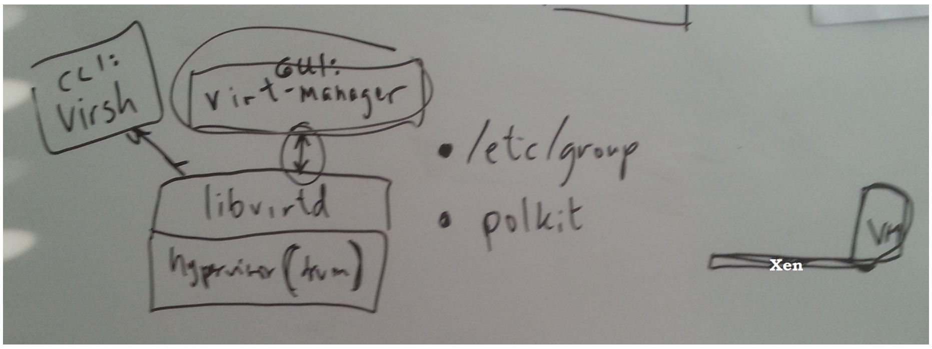
02.02.15

- Trying to find more information about different hypervisors. We have three hypervisors which we will take a closer look at, which are:
    - KVM
        - A lot of good definition around virtualization, KVM, libvirt and virsh: http://www.linuxnix.com/2013/02/kvm-get-hypervisor-and-guest-virtual-machine-details.html
        - Libvirt has several permission/authentication rules:
            - https://wiki.archlinux.org/index.php/libvirt#User_permissions
            - https://www.suse.com/documentation/sles11/book_kvm/data/sec_libvirt_connect_auth.html
            - A default install of libvirt will typically use polkit to authenticate the initial user connection to libvirtd. This is a very coarse grained check though, either allowing full read-write access to all APIs, or just read-only access. The polkit access control driver in libvirt builds on this **capability to allow for fine grained control over the operations a user may perform on an object.**
              https://libvirt.org/aclpolkit.html
    - VMware ESX
    - Xen

04.02.14

- Møte
    - Fortalt litt om kvm med libvirt til begge.

- Libvirt know alot of about how this work in general.
- libvirt.org/apps
- There should be a mechanisem in libvirt to differ Admin and Users.
- Snakke med Jon. ASAP the brenner på dass.
- Erik recommends Kvm
    - libvirt deamon as a service on the host
    - user loggs in and gets a interface from libvirt
        - virtmangeager talks to the libvirt deamon
        - user has to be added to the virt manager on.
        - Virsh CCi
    - The quem-kvm package is more like a binary translator, which will not let KVM to talk directly with the HW, and this will slow down the VMs. Not cool. http://wiki.qemu.org/Index.html
    - Kiosk mode er hva vi vil ha opp.
        - hvordan skal man stoppe fra en bruker fra å krysse ut bilde når kiosk mode er oppe.
    - Describe it when we test it. the test.
    - Dont think much about HPV men mer om hvordan løsningen med libvirt og virtmanges (GUI)
    - Do a sruvay on the managment managment part.
    - google : "local libvirt root group"
- Next week:
    - Jon alt solutions
    - end of week new pc ?
    - reaserched more on the manger



Møte med Jon:

- Han mener at vi kan engelt bruke virtual box.
    - I virtual box kan man lage en virtual maskin som er en clone av en virtual maskin.
    - Da kan vi lage en root VM som da kan bli clonet som kan brukes.
    - Dropp VM ware fra tankene.
    - VM vil skape mer problemer og en del mer vedlikehold.
- Imutable disk images.
- Vi stilte dette spørsmålet om Kiosk mode med virtualbox
    - Han tror det ikke egentlig det ikke finnes.
    - Han sier at det kan være mulig å hacke det til.
- Nevnes i forhold til sikkerhet at de har tilgang til HW så sikkerhet.
- Hvorfor skal man kunne måtte starte en VM for å tilgang til putty.
- Lillemyk skal være med så må vi ha virtualisering.
- Hvis host oset er en for for linux. tre valg av teknologier. Xen og kvm og en til han ikke vil nevne.
- Utfordringen der er at de forskjellige virt mangagerne er lite standardisert de trenger som oftest mye rettiheter.
- Virtualbox er det siste valget om pakker mangerer hypwerviser og etc i en pakke.
- Xen via xem full controll gjennom ccl, libvirt med xen mister du mye funksjonalitet.
- Virtualbox du får gui og kan bruke ccl og du kan scripte.
- En vanlig måte er å lage disk imge er å bruke quem (dette er for xen og kvm)
- hvis det er sånn som at vi gjøre det som vi er nå så finnes det vm ware lab manager som har tatt 12 år og utvikle.
- Logge på med forskjellige så må vi kopiere imges til deres hjemme områder.
- Det er en viktig ting å kunne bruke hosten som en workstation.
- hvis host os er windows så funger virtual box fungerer det veldig bra.

https://www.virtualbox.org/manual/UserManual.html

05.02.15

- More info about Libvirt:
    - A libvirt forum: https://www.redhat.com/archives/libvirt-users/index.html
    - Can use **SASL**, which is used to provide authentication. This will encrypt (MD5) all information going between the hypervisor and libvirt. After sasl is active, you will be prompted by libvirt to provide a user account and password each time an operation is performed. http://prefetch.net/blog/index.php/2009/06/16/create-sasl-accounts-for-libvirt/
        - It’s possible to use SASL authentication using **LDAP** db:
            - http://blog.toxa.de/archives/493

- ○ **WebVirtMgr** is a libvirt-based Web interface for managing virtual machines
    - ■ Screenshots: https://github.com/retspen/webvirtmgr/wiki/Screenshots
    - ■ http://retspen.github.io/
    - ■ https://github.com/retspen/webvirtmgr/
- ○ Usermanual: https://libvirt.org/html/

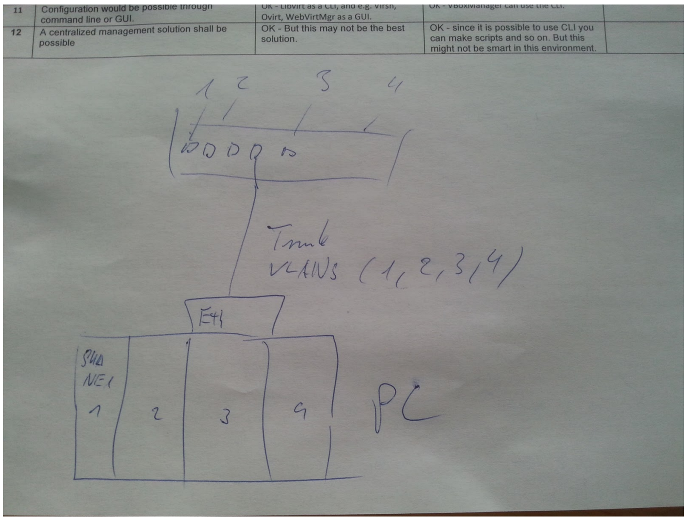
- VirtualBox
    - VBoxManage is the command-line interface to VirtualBox. https://www.virtualbox.org/manual/ch08.html

09.02.15

- VirtualBox:
    - You can absolutely choose what the user is available to do or not to do. with hideing different menu choices.
- Libvirt
    - Polkit reference manual: http://www.freedesktop.org/software/polkit/docs/latest/index.html
    - A brief guide to PolicyKit: http://scarygliders.net/2012/06/20/a-brief-guide-to-policykit/
        - Same guy as also made a basic GUI of polkit: https://github.com/scarygliders/Polkit-Explorer

11.02.15
Møte:

- Vi trenger ikke å tenke på at kabelen skal bli dratt ut, etc.
- Hver PC skal være tilkoblet til pod
- 12 hosts totalt sa thomas (2 per pod).
- Vi må tegne opp et nettverkskart
- Elevene skal kunne copy paste mellom VMene
- Viktig at John godkjenner nettstrukturen. Der alle PCene er koblet sammen til HIG-backbone
- Ikke at det er en del av oppgaven vår, men han ønsker at det skal være mulig til senere å configurere VMene remote.
- Hver PC må ha en trunk med VLANS (Forskjellige subnet):



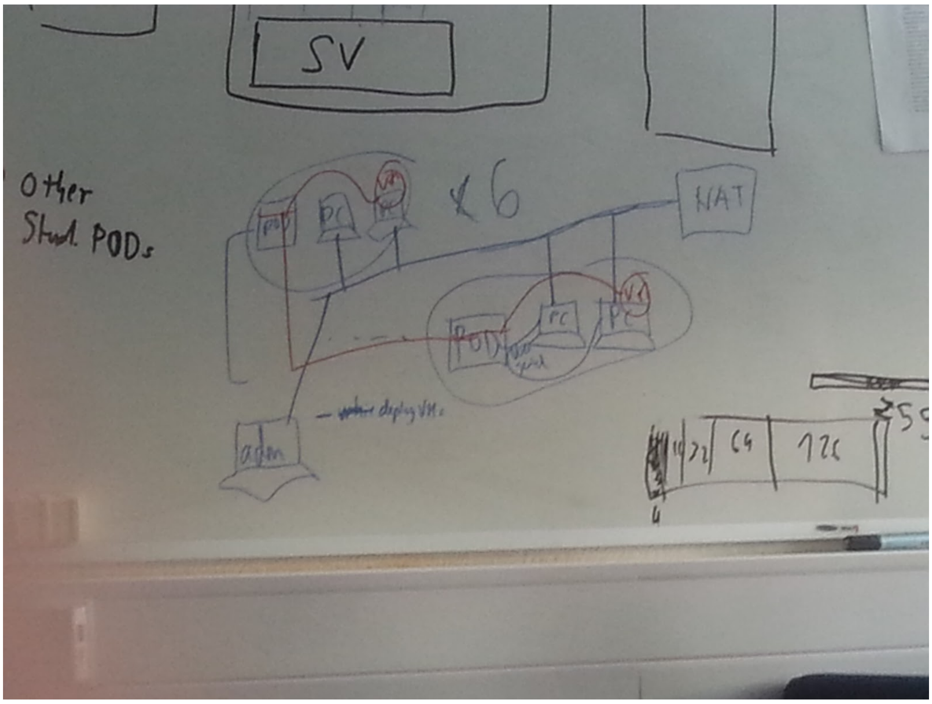
11	Configuration would be possible through command line or GUI.	OK - Libvirt as a CLI, and e.g. Virsh, Ovirt, WebVirtMgr as a GUI.	OK - VBoxManager can use the CLI.	
12	A centralized management solution shall be possible	OK - But this may not be the best solution.	OK - since it is possible to use CLI you can make scripts and so on. But this might not be smart in this environment.	

- Viktig at vi har med imutual-images.
- Erik sier at det er viktig å gjøre hostene selv maintable, så de kommer f.eks til å slette temp. filer etter en vis tid, og updaterer seg selv, etc.
- PCene kommer til å ha 32GB ram. De har 15 000kr per PC å bruke
- Erik påstår at vi "burde" ha en SSD!
  - Erik synes det kan være også mulig å heller bestille PCer via komplett, men da har de ikke 3 års garanti avtaler med DELL.
- Det skal burde være forskjellige images fr de forskjellige service.
- Erik sier at det er bedre å bruke libvirt enn VirtualBox, fordi det kan være lettere å sette grenser og teste når du kan bruke commandline og prøve å teste via libvirt, istedenfor en hel applikasjon.
- Vi burde også fokusere på åbruke CLI når vi skal konfiguere og prøve KVM. Da lærer man mer og får en bedre oversikt over hva som er mulig og ikke.

- Til neste uke skal vi lage et nettverkskart
  - Snakke med john og få det godkjent
  - Finn ut hvordan vi via CLI kan foreta configurasjon på KVM.
  - Finn John og spør han hvorfor han foretrekker VirtualBox. For erik ser ikke hvorfor det er bedre enn å bruke libvirt og KVM.
  - Viktig at i oppgaven å sette klare linjer for hvorfor vi gjør og velger det vi gjør.
  - Vi må få installert KVM og Libvirt. Så skal en root-bruker opprette en VM. Deretter skal en user kunne logge seg på og starte opp den VMen, men alle endringene skal bli lagret i en egen fil. Så nestegang når en annen bruker logger seg inn, skal de kunne starte VMen, men da fra sin egen scratch, og

endringene skal bli lagret til en egen fil. Slik vil alle brukere kunne starte og jobbe med et base-OS, med sine egene endringer på.

- Erik tegnet et oversikt hvordan nettverket kan se ut:



Fikk mail av Erik:

- Jeg har snakka med Jon, og er enig med han i at vi prøver virtualbox først.

16.02.15

- VirtualBox:
    - Testet dette med VirtualBox og immutable images.
    - Samt om vi får bootet det samme image med en annen bruker
        - Det fikk vi til, men den andre brukeren sitt image var ikke immutable.
    - Men på "admin" brukeren våres så måtte vi sette imaget til en gruppe og gi read and write muligheter for at vi skulle få det til at en annen bruker kunne bruke samme image.
    - Vi fikk til at den ene brukeren kunne ha normal og gjøre endringer som da den som hadde immutable kunne se.

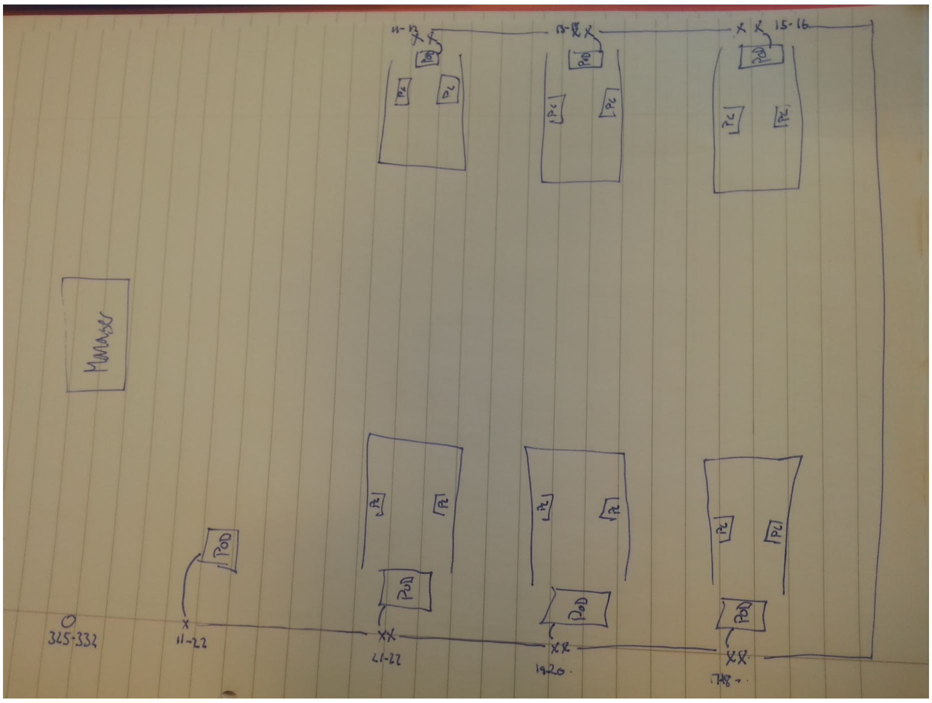
- Vi Gikk mye frem og tilbake for å teste forskjellige måter. Men det gikk ikke så bra med ihvertfall kun Read rettigheter.
- Vi måtte også sette oss litt inn i group policy for å få det til å funke med det første.
- Vi har en annen vi også kan se på og det er multi attach mode.

17.02.15

- Ble nødt til å installere alt på nytt igjen på PCen. Knota litt med å få installert virtualbox-4.3.22.
    - Oppretter en admin bruker “ourbox” og en standard bruker “me”.
    - La begge til i gruppen “aaa” **usermod -a -G aaa ourbox; usermod -a -G aaa me**
    - Oppretter en VM som (Ubuntu 14.04.1 Desktop) på ourbox brukeren. Den plasserte vi i **/home/ourbox/Desktop/ub.vdi**
    - Gjorde den read and executeable av grupper: **chmod 755 ub.vdi**
    - La filen til gruppen: **chggrp aaa ub.vd**i

    - Kommandoer for å opprette en ny VM i student/me bruker:
        - http://www.electricmonk.nl/log/2011/09/24/multiple-virtualbox-vms-using-one-base-image-copy-on-write/
        - VBoxManage create --name Ubuntu --ostype **Ubuntu_64** --register
        - VBoxManage modifyvm Ubuntu --memory **1024**
        - VBoxManage modifyvm Ubuntu --vram **256**
        - VBoxManage modifyvm Ubuntu --cpus **2**
        - VBoxManage storagectl Ubuntu --name **sata1** --add **sata**
            - Setter storage til å bruke sata.
        - VBoxManage storageattach Ubuntu --storagectl sata1 --port **0** --device **0** --type **hdd** --medium **/home/ourbox/Desktop/Ubuntu.vdi** --mtype **immutabl**
            - Legger til originale VMen og setter den til immutable.
    - Kommandoene over har vi lagt inn i en executable bash fil. Tanken er å la den bli kjørt hver gang kontoen blir besøkt.
    - Multiattach mode fungerer veldig likt som Immutable, den største forskjellen er at når du restarter maskinen så blir ikke differansen slettet. Dette medfører man kan kjøre to VM av samme image men den lager seg en snapshot som da inneholder forskjellen.
    - Vi kan også begrense hva som skal være tilgjengelig for brukere av management toolsen, mer info: https://www.virtualbox.org/manual/ch09.html#guitweaks
- Problemer:
    - Hvis en student har tatt snapshot eller lagret endringene sine, og admin må gjøre en endring på originalen, så vil dette føre til at alle lagringene hos brukerne vil bli ødelagt. I teorien. Dette må vi teste ut.

- Oversikt over kabling i CISCO labben per dags dato:



18.02.15
Referat fra møtet:

- Til neste gang skal vi ferdiglage en liten presentasjon for å demonstrere hvordan immutable, osv. fungerer med VirtualBox og KVM.
- Johns årsak til å bruke VirtualBox:
  - Virtualbox client is known interface for students
  - har funksjonalitet for immutable
- Gjør en “**lsattr**” på immutable disken til virtualbox
  - **chattr +i** -> make the file imutable
  - ----i---
- Om to uker skal vi ha klar:
  - Noen eks. av hvordan nettverkskjema kan/ser ut i cisco rommet
    - Bruk fancy program, slik at vi har en original under, og vi kan tegne endringe rog forslag med farger oppå.
    - Google this for inspiration: “Network lab diagram”

- Sett opp kapittler og lag struktur for hele bacheloroppgaven.
  - lag bulletpoints hvor skal hva, og sett inn diagrammer
- Hvis erik hadde satt opp systemet, ville han ha satt en manager med Forman
  - Vi kan ta en tur til terje og få visning om hvordan forman fungerer
- Thomas sa at hvis elevene selv setter opp grupperinger på PCene selv, så vil ikke det være verden undergang heller.

20.02.15

- FIkk tips om en side ang. ldap: http://www.stud.hig.no/~131284/it/hig.html
- A startup script have been set up, so everytime a user (not ourbox user) logs in, the script for creating a default Ubuntu vm will automatically be run. All scripts which ends with **.sh** in /etc/profile.d/ will be executed every time a user logs in. Therefore we created a script there named **defaultVMs.sh** with the content:

```
#!/bin/bash
os="Ubuntu";
user=$(whoami);
DIRECTORY=~/VirtualBox\ VMs/$os;
if [ $user != “ourbox” ]; then
        osRegEx =”^\”${os}\””;
        knownVMs=$(VBoxManage list vms | grep -e “$osRegEx“| awk ‘{print $1}’ | cut
                    -d “\”” -f 2);
        if [[ $knownVMs != $os ]]; then
                if [[ -d $DIRECTORY ]];then
                        rm -r “DIRECTORY”;
                fi

                VBoxManage create --name $os --ostype Ubuntu_64 --register 2>&1;
                VBoxManage modifyvm $os --memory 1024 2>&1;
                VBoxManage modifyvm $os --vram 256 2>&1;
                VBoxManage modifyvm $os --cpus 2 2>&1;
                VBoxManage storagectl $os --name sata1 --add sata  2>&1;
                VBoxManage storageattach $os --storagectl sata1 --port 0 --device 0
                         --type hdd --medium /home/ourbox/Desktop/Ubuntu.vdi
                        --mtype immutable 2>&1;
        fi
fi
```

  - Cause when the VM is already there, the script will try to create a new VM, but p
- We have been set up a IF to make the script more reliable, and now we are trying to make the script flexible by getting regex to use variables but it seems we can not make it work yet.

- Now we got it to work, how we do not know, but now when every user logs on the machine will create 1 default VM in virtualbox, and if it allready got it, it will not create a new VM.

23.02.15

- http://kashyapc.com/2012/09/14/externaland-live-snapshots-with-libvirt/
- http://serverfault.com/questions/240701/can-kvm-roll-back-changes-to-virtual-disks-automatically
- http://linux.die.net/man/1/qemu-img
- https://www.suse.com/documentation/sles11/book_kvm/data/cha_qemu_guest_inst_qemu-img.html
- http://en.wikibooks.org/wiki/QEMU/Images
- We followed this guide to install QEMU-KVM:
  - http://www.howtogeek.com/117635/how-to-install-kvm-and-create-virtual-machines-on-ubuntu/
  - We also had to install qemu-system package
    - apt-get install qemu-system
- We searched the internet for a smart solution with this Immutable image:
  - http://kashyapc.com/2012/09/14/externaland-live-snapshots-with-libvirt/
  - http://serverfault.com/questions/240701/can-kvm-roll-back-changes-to-virtual-disks-automatically
    - We found a possible solution with
      - http://linux.die.net/man/1/qemu-img
    - Which means that we need to make a script which then takes a snapshot of an image and then we need to delete that snapshot after it has been used.
      - we did not go for this solution
  - We found COW(Copy on write) which we then thought was better
    - http://en.wikibooks.org/wiki/QEMU/Images
    - We found out that COW works like this, that you have a base image which then you make an image from, which we make it possible to make as Copy on write mode, so that the base image doesnt get touch but every change we make will be done to the copy on write img. But it seems like then we have to make a COW image for every user that is going to have a VM, and we are not sure yet on how we are going to make it “immutable like” it seems like we still need to either take a snapshot of the image and then delete the snapshot or we need to delete the COW img and make a new one every time.
    - Creating a new image with all changes made from the original image:
      - `qemu-img create -f qcow2 -b Ubuntu.img test.qcow2`
  - Now we need to create and map this test.img to virsh-manager:
    - We can do that with virt-install:
      -> virt-install -n test --vcpu=1 -r 1024 --disk path=/var/lib/libvirt/images/test.qcow2 --import --accelerate

        - https://acidborg.wordpress.com/2010/02/18/how-to-create-virtual-machines-using-kvm-kernel-based-virtual-machine/
    - Listing up all linked VMs: **virt list --all**

24.02.15

- We got the admin user (ourbox) to run a VM with the qcow2, but when we tried to use the same ubuntuu.img to make a new qcow2 as another user, we where allowed to do so. But it seems like it doesnt have any disk space which is a problem
    - We did some digging and we found this:
        - http://unix.stackexchange.com/questions/159069/how-can-i-create-a-kvm-guest-100-as-a-non-root-user
    - Not sure if this is true or not but just putting it here.
- Found an alternative to starting virtualbox:
    - http://askubuntu.com/questions/310671/start-ubuntu-without-a-desktop-environment-but-start-an-x-application
- We also found out that when we run the virtualbox in our defaultVMs.sh in profile.d we run the application without a window manager, so we need to found out how to start a window manager with virtualbox.
    - is it possible to do this though xsessions instead?

02.03.15

- We have created two groups, **immutable** and **multiattach**:

  -> groupadd immutable

04.03.15
Referat fra møtet:

- Finn ut hvorfor vi ikke velger libvirt/kvm, men heller velger VirtualBox.
- Virtualbox har en mer “mature” brukergrensesnitt og er mer kjent for studentene. KVM er mye bedre og moden når det komme rtil hypervisor, men i vårt tilfelle, så vil virtualbox være mer egnet.
- Lag linux server, mailserver,(windows) og ha det tilgjengelig når Virtualbox starter.
- FInn ut om virtualbox tilfredsstiller alle kravene våre. Hvis ikke, så evt. fortelle hvorfor ikke, og hva som kan gjøres for å få det til.
- 12.april, til da burde vi ha en viss utgave av bacheloroppgaven.
- 2-3 uker før innlevering skal vi også ha en fornyet utgave som erik skal se på.
- Skaff automatic spell check for TexWorks.

09.03.15

- Virtualbox, var rar. Den ville ikke la guest få stort bilde. Den var bare 600x400 eller noe. Så jeg prøver å installere fra internett.
    - Virtualbox vil ikke la guest få bilde i sitt egent vindu.
        - http://www.ubuntugeek.com/virtualbox-and-ubuntu-14-04-display-issue.html

        - Prøvde dette, og det fungerte, vet ikke helt om det var det som gjørde at den ikke viste bilde men.
            - **“Devices -> Insert Guest Additions CD image”**
            - This also mad the desktop window bigger and possible to go in fullscreen mode.
        - How to make a user see libvirt (next paragraph).
        - Looking at the solution of makin “kiosk mode” out of virtualbox.
            - https://forums.virtualbox.org/viewtopic.php?f=7&t=52974
            - Seems to old, it was made in 2012 and it seems like it hasn’t been updated since.
            - Gonna try it to see what it does.
        -

- Libvirt:
    - Install ubuntu to an img file:
      ```
      virt-install -n ubuntu --vcpu=1 -r 1024   --disk
      path=/home/ourbox/Desktop/libvirt/ubuntu.img,size=8 --cdrom
      /home/ourbox/Desktop/os/ubuntu-14.04.2-desktop-amd64.iso  --accelerate
      ```
    - Create an qcow2 image:
      ```
      qemu-img create -f qcow2 -b ubuntu.img test.qcow2
      ```
    - Take in use the qcow2 image (important to specify that this is a qcow2 formatted file):
      ```
      virt-install -n test --vcpu=1 -r 1024 --disk
      path=/home/ourbox/Desktop/libvirt/test.qcow2,size=8,format=qcow2
      --import --accelerate
      ```
    - Create a student (“multiattach”) and a guest (“immutable”) group:
      ```
      groupadd student
      groupadd guest
      ```
    - Create a student user:
      ```
      useradd -d /home/student1 -m student1
      ```
    - Add a user to a group:
      ```
      usermod -G student  -a student1
      ```
    - Add the ubuntu.img file to group student ourbox
      ```
      setfacl -m g:student:rx ubuntu.img
      setfacl -m g:ourbox:rwx ubuntu.img
      ```
    - NOTES:
        - Libvirt service is called:
            - In RHEL or CentOS: **libvirtd**.
            - In Ubuntu: **libvirt-bin**
        - Compact info about how to use libvirt + virsh:
            - https://wiki.archlinux.org/index.php/Libvirt
    - On the user **student1**, we first create a file a qcow2 image and put it on the desktop:

    - **qemu-img create -f qcow2 -b /home/ourbox/Desktop/libvirt/ubuntu.img ~/Desktop/test.qcow2**
  - Now we can create (map) this ubuntu as a (multiattached). Since the student isn't a member of the **libvirtd** group, no root access to the hypervisor is granted. Therefor, we make the student start a session to the hypervisor: **virt-install -n test --vcpu=1 -r 1024 --accelerate --disk path=~/Desktop/test.img,format=qcow2 --import**
  - The step above is also possible to do in the virt-manager (GUI). The problem is that this is not straightforward. (This is not a problem if the qcow2 file is in the /VirtualMachines) To make a student be able to do this, we need to add a step by step guid (source: https://docs.google.com/document/d/1X8TaBP1v_rh8e2QXDGFmkmzoPB13Pvp2V0FQ630VCsU/edit?pli=1):
    - **Step** 1: Create a new VM from [Import existing disk image] option.
    - **Step** 2: Choose your qcow2 image.
    - **Step** 3: Select [Customize configuration before install] before you go forward
      - **note**: without this step, you will got a non-bootable message from the virtual machine.
    - **Step** 4: Identify the qcow2 format to the image and begin installation [Disk1] -> [Advanced options] ->[Storage format]: qcow2 [Begin Installation]
  - CONCLUSION:
    In forehand, you have to create X qcow2 files in the ~/VirtualMachines/. This way, we can create and take in use copy-on-write VMs. Otherwise, we cannot create VMs, the qcow2 file must be created first, which is at minimum ca. 192K. The problem here is:
    - We have to pre-define how many VMs a user can create.
    - 192K * numb. of the particular OS a user can create * numb. of different OSes * numb. of users on the system = Too much unused and occupied space.
    - When the user are going to choose a VM, e.g. Ubuntu1.qcow2, this file will be occupied. Next time the user want to choose and create a new ubuntu VM, the Ubuntu1.qcow2 will still be displayed, but at the end, when the user presses “finish” with the installation of the new VM, an error will occur and say that this .qcow2 file is already used by another VM. This can easily confuse the user, and he/she must remember and has an overview of what kind of files is used and not.

10.03.15

- Virtualbox: ‘
  - Fluxbox: https://help.ubuntu.com/community/Fluxbox

- Tried fluxbox again, but we had a tip from Magnus the student assistent, instead of:
  - fluxbox&;
  - virtualbox;
- he said that we should try:
  - fluxbox& virtualbox;
- on the same line which then works.
- The next we need to do is make sure that you cannot access anything through fluxbox and the virtual consoles.
- How to deactive the tty (virtual consoles)
  - http://ubuntuforums.org/showthread.php?t=1400893
  - https://help.ubuntu.com/community/RemoveTTY
  - **sudo mv tty1.conf /home/ourbox/Desktop**
    - It says you can either remove them or just move them from the "**/etc/init**" folder.
    - This should work when i restart the pc.
    - **It WORKS**
    - Just move/remove all the tty*.conf files from /etc/init folder and restart the comp.
  - Tried this: http://linuxpoison.blogspot.no/2008/08/how-to-disable-virtual-consoles-altf1.html
    - /etc/inittab doesnt exists on ubnuntu
- Custumize fluxbox:
  - https://help.ubuntu.com/community/Fluxbox/customize
  - It seems like it make a menu for each user, so i need find the file that custumize for all users.
    - http://fluxbox.org/help/man-fluxbox-menu.php
      - it seems to be around **@pkgdatadir@/menu**
        - how to find this type "**fluxbox -i"**
    - **nano /etc/X11/fluxbox/fluxbox-menu**
      - I just delete everythin execpt restart and exit choices in the menu.
        - How to set up a menu:
          - [submenu] (Application) {}
                    [submenu] (Accesibiluty) {}
                            [exec] (Xmag)
            {xmag} <>   //this should be a line up.
            end
        - [restart] (Restart)
      - we have a problem that it seems like it wont execute commands, because we are trying to

make a command that shutsdown virtualbox so that you log ut, but fluxbox wont run that command

- **pkill VirtualBox**

○ i tried to remove som of the componets in the toolbar in etc/X11/fluxbox/init file, but i doesnt seem to work.

11.03.15
REFERAT:

- En innlevering til erik 7.april (rett etter påsken), 1-2 before handin (rødpen)
- lage en plan for 9 ukene framover, så vi dekker alt vi skal ha med,
- Spør erik hvorfor vi har probemer med fluxbox når vi starter det i profile.d/, vi får ikke kjørt eller eksekvert noen programmer.
- Finn ut om vi trenger home directory, eller ikke. Ta hensyn til så ikke vilene blir opprettet i homedirectory på loke sin servere, men på lokale PC
- **powerbroker** for administrasjonskontrol!!!!!
- stay away of mounting the homefolder inside the cp
- Kanskje ha mulighet så en VM / mappe kan ha flere brukere, og ikke bare en (lage grupperinger på en måte).
- Husk at hvis det vil være nødvendig for at en bruker skal kunne sette grupper selv, så vil det være godkjent
- lag et skript som vil ta inn parametere, som lager og gjør at vmene kommer til å bli delt mellom de og de brukerne. Det eneste ved siden av virtualbox vinduet som skal være tilgjengelig for brukeren.
- **umask** command: setting the default access mode. Kan være til hjelp.
- Finne en måte så en bruker kan ha mulighet til å sette grupperinger selv. For egentlig så må enn være root bruker for å gjøre det.
- Brukere kan ha tilgang og setter en fil i en shared directory, der filen inneholder brukernavn som de ønsker gruppering på. Så er det crontab som kjører og sjekker hvert 5 min og exekverer filene
- MAP, enkleste måte med LDAP er “PADL software pty ltd” (**PAM_LDAP**). Spør John og få han til å gi oss LDAP mulighet. Hvis ikke, så må vi gjøre det lokalt
  - Hver PC som starter får
- Lag en presentasjon og visning til thomas om hva som fungerer og tilfredstiller kravene våre, og hva ikke, (13. april og ut burde vi ha det lagt opp klar for visning)
- Erik 21.mars - 10.april 26.mars tilgjengelig på email)
- Uke 12 er det møte.

LOGG:

- Vi snakket med IT-avdelingen (ikke John) om å få lov til å koble oss opp mot LDAP på skolen, og det fikk vi lov til.

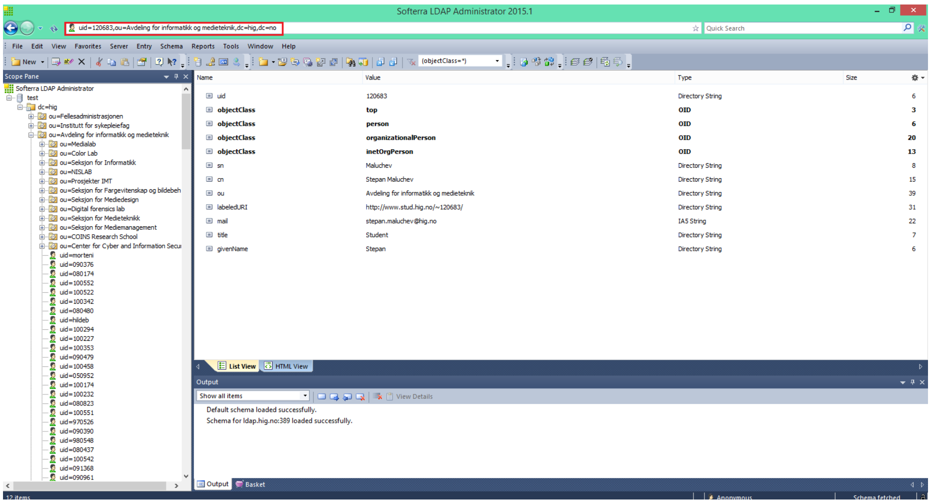
### Authentication, etc.

12.03.15

- How to setup + basic information about LDAP/PAM:
  - https://wiki.debian.org/LDAP/PAM
  - http://www.padl.com/OSS/pam_ldap.html
- **sudo apt-get install libpam-lpdap**
  - skriver inn: ldap://ldap.hig.no som addresse
  - ou og dc ble ciscolab og local.
  - I removed it again.
- Started to work on why fluxbox wont run commands.
  - http://gotoanswer.stanford.edu/?q=Fluxbox+Menu+and+root+commands

13.03.15

- Windows tool for LDAP authentication **LDAP Administrator 2015.1** (can find info about users, but cannot verify passwords or users. Looks like we need a admin account):
  - http://www.ldapbrowser.com/download.htm
    - My DN in the ldap.hig.no: uid=120683,ou=Avdeling for informatikk og medieteknik,dc=hig,dc=no



uid=120683,ou=Avdeling for informatikk og medieteknik,dc=hig,dc=no

```
Softerra LDAP Administrator
  test
    dc=hig
      ou=Fellesadministrasjonen
      ou=Institutt for sykepleiefag
      ou=Avdeling for informatikk og medieteknik
        ou=Medialab
        ou=Color Lab
        ou=Seksjon for Informatikk
        ou=NISLAB
        ou=Prosjekter IMT
        ou=Seksjon for Fargevitenskap og bildebeh
        ou=Seksjon for Mediedesign
        ou=Digital forensics lab
        ou=Seksjon for Medieteknikk
        ou=Seksjon for Mediemanagement
        ou=COINS Research School
        ou=Center for Cyber and Information Secu
        uid=morteni
        uid=090376
        uid=080174
        uid=100552
        uid=100522
        uid=100342
        uid=080480
        uid=hildeb
        uid=100294
        uid=100227
        uid=100353
        uid=090479
        uid=100458
        uid=050952
        uid=100174
        uid=100232
        uid=080823
        uid=100551
        uid=970526
        uid=090390
        uid=980548
        uid=080437
        uid=100542
        uid=091368
        uid=090961
```

Name	Value	Type	Size
uid	120683	Directory String	6
objectClass	**top**	**OID**	**3**
objectClass	**person**	**OID**	**6**
objectClass	**organizationalPerson**	**OID**	**20**
objectClass	**inetOrgPerson**	**OID**	**13**
sn	Maluchev	Directory String	8
cn	Stepan Maluchev	Directory String	15
ou	Avdeling for informatikk og medieteknik	Directory String	39
labeledURI	http://www.stud.hig.no/~120683/	Directory String	31
mail	stepan.maluchev@hig.no	IA5 String	22
title	Student	Directory String	7
givenName	Stepan	Directory String	6

```
Default schema loaded successfully.
Schema for ldap.hig.no:389 loaded successfully.
```

- OpenLDAP setup:
  - https://wiki.archlinux.org/index.php/OpenLDAP
  - https://wiki.archlinux.org/index.php/LDAP_authentication

16.03.15

- **http://www.linuxquestions.org/questions/slackware-14/how-do-i-start-fluxbox-52159/**
  - Sett på det å få startet fluxbox kanskje fra et annet sted, årøver å finne litt ut om hvordan startx fungerer helt og holdent.
- https://wiki.ubuntu.com/CustomXSession
- **LDAP:**
  - Find who you are, as a response from the LDAP-DB:
    ldapwhoami -h 128.39.41.128 -x -D "" // Return “Anonymous”
  - If we want to check a username and retrieve some basic information about the user, we do this with this syntax (source for the script syntax):
    ldapsearch -x -h 128.39.41.128 -b dc=hig,dc=no -p 389 '(uid=120683)'

    ```
    > # extended LDIF
    > #
    > # LDAPv3
    > # base <dc=hig,dc=no> with scope subtree
    > # filter: (uid=120683)
    > # requesting: ALL
    > #
    > # 120683, Avdeling for informatikk og medieteknik, hig.no
    > dn: uid=120683,ou=Avdeling for informatikk og medieteknik,dc=hig,dc=no
    > uid: 120683
    > objectClass: top
    > objectClass: person
    > objectClass: organizationalPerson
    > objectClass: inetOrgPerson
    > sn: Maluchev
    > cn: Stepan Maluchev
    > ou: Avdeling for informatikk og medietekniklabeled
    > URI: http://www.stud.hig.no/~120683/
    > mail: stepan.maluchev@hig.no
    > title: Studentgiven
    > Name: Stepan
    > # search result
    > search: 2
    > result: 0 Success
    > # numResponses: 2
    > # numEntries: 1
    ```

        - If we want also to add another search parameter, e.g. we want to search for a username under "Avdeling for informatikk og medieteknik", we would to this:
          ldapsearch -x -h 128.39.41.128 -b dc=hig,dc=no -p 389 '(&(uid=120683)(ou=Avdeling for informatikk og medieteknik))'

18.03.15
Møtereferat:

- Vi må ha testing på planen.
- Følger det gant skjemaet opp kravspecc? (Spør Erik).
- Les om user acceptance før vi gjør testing og før vi skriver om det.
    - Det burde være i henhold til kravspec.
        - Bruk Thomas, de assistentene og 2 andre random. (Maks 5)
        - 15 -20. April. burde vi gjør det.
- Vi kan ikke nå 27 mars - 10 april. Thomas.
- Møte med Erik den 10 april og møte med Thomas den 15 april. hvor vi snakker om hvordan vi skal teste.

20.03.15

- Started working a little bit on the report.
- LDAP:
    - If we want to check a user with uid **120683** and compare the username **Maluchev,** we can compare this by first specifying the full DN, and then the sn:
      ldapcompare -x -h 128.39.41.128 'uid=120683,ou=Avdeling for informatikk og medieteknik,dc=hig,dc=no' sn:Maluchev
      > TRUE
      This will return true,false or an error message.
    - If we have an account in the ldap directory, we can try this link: http://www.unixmen.com/configure-linux-clients-authenticate-using-openldap/

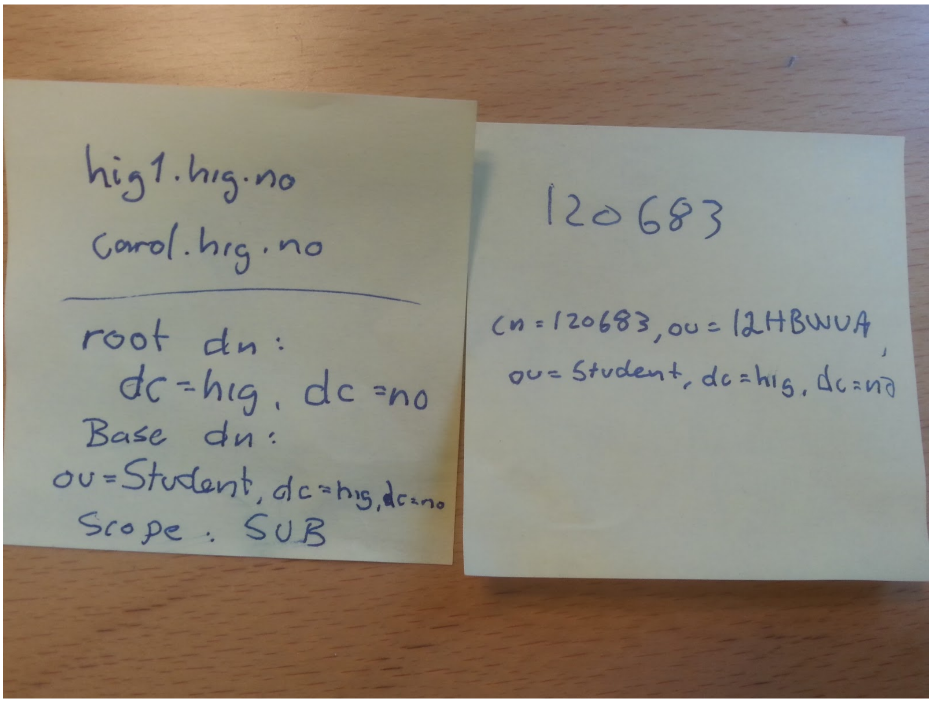
23.03.15

- Some nice general information around PAM:
    - https://www.digitalocean.com/community/tutorials/how-to-use-pam-to-configure-authentication-on-an-ubuntu-12-04-vps
- Some nice information on how to authenticate with ldap using libpam_ldap:
    - https://www.digitalocean.com/community/tutorials/how-to-authenticate-client-computers-using-ldap-on-an-ubuntu-12-04-vps

REFERAT FRA MØTET MED JOHN:

- **Fant ut at ldap.hig.no er en server for enkle oppslag om brukere, som email, bilde og navn. Så vi har hittil prøvd å vertifisere passordet mot feil server de siste ukene. Årsaken var at etter et besøk hos IT-avdelingen, fikk vi beskjed om at vi bare skulle prøve oss fram. Dermed har vi klart å bruke opp 2 uker på lite fornuftig framdrift.**

- Fikk to lapper:
    - 1.note:
      Server we should connect to: **hig1.hig.no** (IP:128.39.140.7) or **carol.hig.no** (IP:128.39.140.10)
      My user account we will authenticate with: 120683
      bindDN: **cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no**
    - Han sa det var to muligheter å autentisere seg, enten mot den vi fikk som gjør at vi skal bruke PAM mot Active Directory, dette betyr at vi må skreddersy PAM mye mer enn om vi da skulle ha brukt den andre servern som vi kunne koble oss i mot. Den bruker da Poisix.
    - Vi kan også få mulighet til å gjøre dette med ansatte også, men da må vi filtere mer.
        - Vi fikk da gitt "treet" for studenter, så om vi skal gjøre det med ansatte også så er vi nødt til å gå tilbake til jon.



hig1.hig.no
carol.hig.no

root dn:
dc=hig, dc=no
Base dn:
ou=Student, dc=hig, dc=no
Scope : SUB

120683

cn=120683, ou=12HBWUA,
ou=Student, dc=hig, dc=no

LOG:

- After our meeting with Jon Langseth, we were told to configure pam to talk with AD, here is a link which is going to be read:
    - https://technet.microsoft.com/en-us/magazine/2008.12.linux.aspx
-

24.03.15

- Tried this tutorial for ldap with AD:
    - http://thejoyofstick.com/blog/2012/03/31/authenticating-linux-users-against-microsoft-active-directory/

    -
      ```
      host carol.hig.no
      base ou=student,dc=hig,dc=no
      ldap version 3
      binddn cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no
      bindpw "MY PWD"
      pam_login_attribute sAMAccountName
      ```
        - THE TUTORIAL ABOVE DID NOT WORK!
- https://wiki.samba.org/index.php/Local_user_management_and_authentication/nslcd

27.03.15

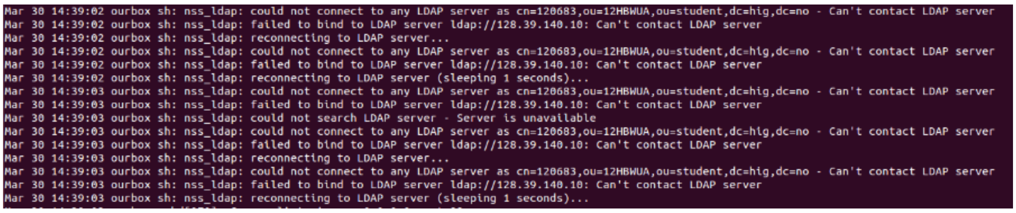
- Tryed to follow this link: https://help.ubuntu.com/community/LDAPClientAuthentication But something went wrong with the authentication of my account. Will work with this later on.
    - **Couldn't get this to work either…**

30.03.15

- Tried this video: https://www.youtube.com/watch?v=kSCx3tzC0cA
    - **Couldn't get this to work either…**
- Trying this tutorial: http://naidutrk.blogspot.de/2012/03/setting-up-ldap-client-authentication.html
    - **Couldn't get this to work either… Nothing happens besides the usual login prompt to ourbox local user.**
- NOTE: To check the log when we cannot log in with an account, visit **/var/log/auth.log**.

  I have authentication problems with my own connection to LDAP, with error message:
  ```
  >nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
  >nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP serourever
  ```

When we boot up the host, the host tries to set up an connection with the LDAP server,
but fails:



```
Mar 30 14:39:02 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:02 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:02 ourbox sh: nss_ldap: reconnecting to LDAP server...
Mar 30 14:39:02 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:02 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:02 ourbox sh: nss_ldap: reconnecting to LDAP server (sleeping 1 seconds)...
Mar 30 14:39:03 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: could not search LDAP server - Server is unavailable
Mar 30 14:39:03 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: reconnecting to LDAP server...
Mar 30 14:39:03 ourbox sh: nss_ldap: could not connect to any LDAP server as cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no - Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: failed to bind to LDAP server ldap://128.39.140.10: Can't contact LDAP server
Mar 30 14:39:03 ourbox sh: nss_ldap: reconnecting to LDAP server (sleeping 1 seconds)...
```

  - ○ First I tried to make an ldapsearch with the same binddn, and I got an connection:
    ldapsearch -x -h 128.39.140.10 -b ou=student,dc=hig,dc=no -p 389 -D cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no -W samaccountname=120683
    - ■ How to make an ldapsearch: http://blogs.splunk.com/2009/07/30/ldapsearch-is-your-friend/
- NOTE: file: /etc/nsswitch.conf let us set where the credentials like password, groups, shadow will be gathered from. The default is **compat** or **files**, which means that the information is to be find on the local machine. In our case, we must set **ldap** after the **files**. This way, if the user is not found on the local machine, it will try to authenticate through the ldap-server. For now, we cannot get the connection with the LDAP-server with our username.
  - ○ More info about nsswitch.conf: http://searchitchannel.techtarget.com/feature/Using-nsswitchconf-to-find-Linux-system-information
- People say that we should use **sssd** rather than nss_ldap, pam_ldap or nscd (http://serverfault.com/questions/626527/client-authentication-invalid-credentials-ldap)
  - ○ When installing sssd on ubuntu server 14.04, there are no default sssd.conf file in /etc/sssd/, but there are one located in /usr/share/doc/sssd-common/examples/sssd-example.conf. Now we need to copy that one into /etc/sssd/sssd.conf:
    cp /usr/share/doc/sssd-common/examples/sssd-example.conf /etc/sssd/sssd.conf
    - ■ http://askubuntu.com/questions/247763/why-is-my-sssd-conf-file-missing-after-installing-sssd
  - ○ Trying to follow this tutorial: http://linuxrackers.com/doku.php?id=ubuntu_13.04_-_set_up_ldap_client_using_sssd_for_auth_and_identity
    - ■ Noop, didn't do anything new as I can see…
- Trying to see what this does for us:
  - ○ http://people.skolelinux.org/pere/blog/Caching_password__user_and_group_on_a_roaming_Debian_laptop.html

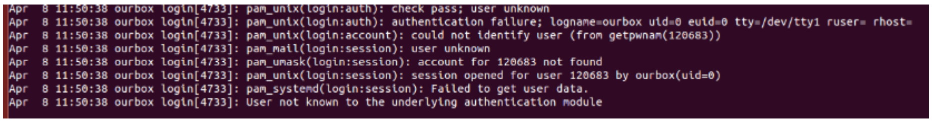
07.04.15

- Writing on the report and talking with John for help with the authentication part.
- John found out our problem in the config file /etc/ldap.conf, he adde:

  - **bind_policy soft**
    As default, it’s **bind_policy hard**, which means that if the host tries to connect to the AD server but fails the first time, then the whole LDAP-authentication fails. With this set to **soft**, the host have the ability to reestablish the connection several times. (Not sure why this fails the first time).
  - The next thing we haven’t looked at, was how the host maps to the AD server. As default, all of the below lines was commented out, while John enabled every of them besides the homedirectory line (we won’t mount the user’s homedirectory at our host anyway):
    # RFC 2307 (AD) mappings
    nss_map_objectclass posixAccount user
    nss_map_objectclass shadowAccount user
    nss_map_attribute uid sAMAccountName
    #nss_map_attribute homeDirectory unixHomeDirectory
    nss_map_attribute shadowLastChange pwdLastSet
    nss_map_objectclass posixGroup group
    nss_map_attribute uniqueMember member
    pam_login_attribute sAMAccountName
    pam_filter objectclass=User
    pam_password ad
  - Jon mentioned maybe why the user logs out instantly is because it doesnt have a homedirectory and it doesnt know what shell the user is using.

08.04.15

- Writing on the analyse part on the report.
- Now when we log in with our student number and right password, we will be authenticated through the AD, but since the host don’t know where the home directory is, we will automatically log out. In the log file it looks like this:

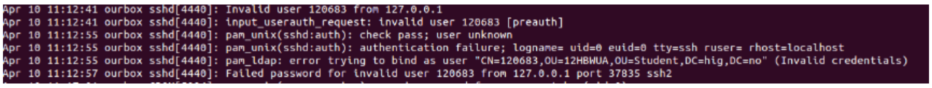


```
Apr  8 11:50:38 ourbox login[4733]: pam_unix(login:auth): check pass; user unknown
Apr  8 11:50:38 ourbox login[4733]: pam_unix(login:auth): authentication failure; logname=ourbox uid=0 euid=0 tty=/dev/tty1 ruser= rhost=
Apr  8 11:50:38 ourbox login[4733]: pam_unix(login:account): could not identify user (from getpwnam(120683))
Apr  8 11:50:38 ourbox login[4733]: pam_mail(login:session): user unknown
Apr  8 11:50:38 ourbox login[4733]: pam_umask(login:session): account for 120683 not found
Apr  8 11:50:38 ourbox login[4733]: pam_unix(login:session): session opened for user 120683 by ourbox(uid=0)
Apr  8 11:50:38 ourbox login[4733]: pam_systemd(login:session): Failed to get user data.
Apr  8 11:50:38 ourbox login[4733]: User not known to the underlying authentication module
```

  The log says that the user cannot be found, but it opens a session for the user. Not sure why we get this error message, but John said that the error came because a missing knowledge of what homedirectory to mount.
- Trying to understand how we can use the UID from the AD to create a user on the host with a new local homedirectory.
  - Cannot figure out this...
- TIPS: If we need to reinstall the OS, we only need to install those package to setup ldap:
  sudo apt-get install libnss-ldap ldap-utils
  - Update the pam files: **pam-auth-update**

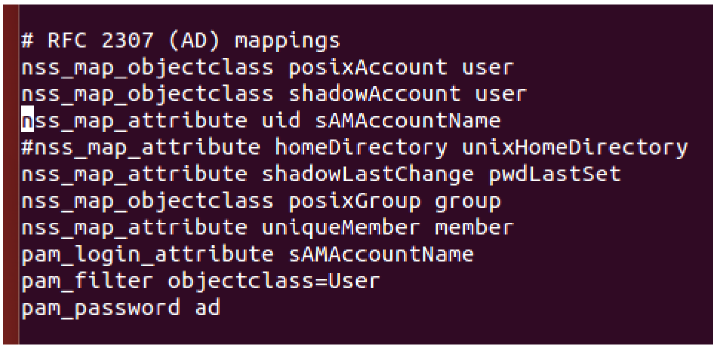
10.04.15

- Reading this, which is the RFC for AD mappings in ldap.
    - ○ http://www.rfc-base.org/txt/rfc-2307.txt
    - ○ http://manpages.ubuntu.com/manpages/quantal/man8/pam_mkhomedir.8.html
    - ○ https://cdc.iseage.org/tutorial-pam-ldap-authentication-active-directory-debianubuntu/
- When we try to SSH into the host, we get this error msg:



```
Apr 10 11:12:41 ourbox sshd[4440]: Invalid user 120683 from 127.0.0.1
Apr 10 11:12:41 ourbox sshd[4440]: input_userauth_request: invalid user 120683 [preauth]
Apr 10 11:12:55 ourbox sshd[4440]: pam_unix(sshd:auth): check pass; user unknown
Apr 10 11:12:55 ourbox sshd[4440]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=localhost
Apr 10 11:12:55 ourbox sshd[4440]: pam_ldap: error trying to bind as user "CN=120683,OU=12HBWUA,OU=Student,DC=hig,DC=no" (Invalid credentials)
Apr 10 11:12:57 ourbox sshd[4440]: Failed password for invalid user 120683 from 127.0.0.1 port 37835 ssh2
```

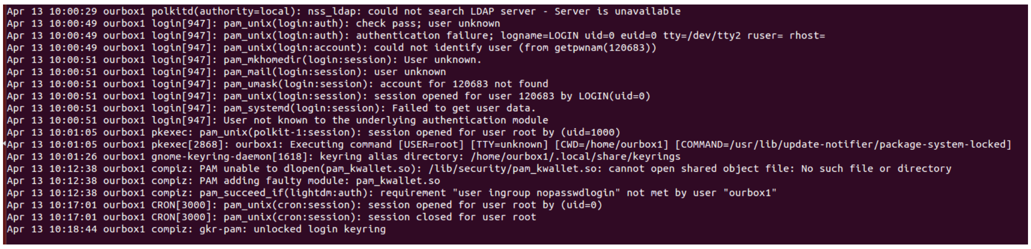
13.04.15

- Meeting with Erik. We need help with LDAP mounting:
    - ○ /etc/pam.d/
        - ■ we added line for making home directory, we tried differen files. common-auth, common-account, common-session and login file
            - **session required pam_mkhomedir.so skel=/etc/skel/ umask=0077**
    - ○ etc/ldap.conf
        - ■ Changed:
            - Base ou=student, dc=hig,dc=no
            - uri ldap
            - ldap version 3
            - binddn
            - scope
            - bind policy
            - RFC 2307 (AD) mappings
                - ○ We just removed the hashtag in front of everyone. except home directory.
                - ○ 

```
# RFC 2307 (AD) mappings
nss_map_objectclass posixAccount user
nss_map_objectclass shadowAccount user
nss_map_attribute uid sAMAccountName
#nss_map_attribute homeDirectory unixHomeDirectory
nss_map_attribute shadowLastChange pwdLastSet
nss_map_objectclass posixGroup group
nss_map_attribute uniqueMember member
pam_login_attribute sAMAccountName
pam_filter objectclass=User
pam_password ad
```

- ○ Our problem is that when you login you just log back out.

        - Jon said this had something to do with that you don't have a home directory and it doesn't know your shell.
        - And he also mentioned something about UID from the AD server.
    - This is our error:

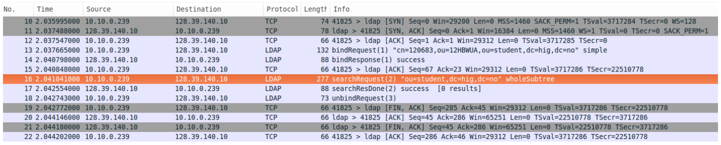


```
Apr 13 10:00:29 ourbox1 polkitd(authority=local): nss_ldap: could not search LDAP server - Server is unavailable
Apr 13 10:00:49 ourbox1 login[947]: pam_unix(login:auth): check pass; user unknown
Apr 13 10:00:49 ourbox1 login[947]: pam_unix(login:auth): authentication failure; logname=LOGIN uid=0 euid=0 tty=/dev/tty2 ruser= rhost=
Apr 13 10:00:49 ourbox1 login[947]: pam_unix(login:account): could not identify user (from getpwnam(120683))
Apr 13 10:00:51 ourbox1 login[947]: pam_mkhomedir(login:session): User unknown.
Apr 13 10:00:51 ourbox1 login[947]: pam_mail(login:session): user unknown
Apr 13 10:00:51 ourbox1 login[947]: pam_umask(login:session): account for 120683 not found
Apr 13 10:00:51 ourbox1 login[947]: pam_unix(login:session): session opened for user 120683 by LOGIN(uid=0)
Apr 13 10:00:51 ourbox1 login[947]: pam_systemd(login:session): Failed to get user data.
Apr 13 10:00:51 ourbox1 login[947]: User not known to the underlying authentication module
Apr 13 10:01:05 ourbox1 pkexec: pam_unix(polkit-1:session): session opened for user root by (uid=1000)
Apr 13 10:01:05 ourbox1 pkexec[2868]: ourbox1: Executing command [USER=root] [TTY=unknown] [CWD=/home/ourbox1] [COMMAND=/usr/lib/update-notifier/package-system-locked]
Apr 13 10:01:26 ourbox1 gnome-keyring-daemon[1618]: keyring alias directory: /home/ourbox1/.local/share/keyrings
Apr 13 10:12:38 ourbox1 compiz: PAM unable to dlopen(pam_kwallet.so): /lib/security/pam_kwallet.so: cannot open shared object file: No such file or directory
Apr 13 10:12:38 ourbox1 compiz: PAM adding faulty module: pam_kwallet.so
Apr 13 10:12:38 ourbox1 compiz: pam_succeed_if(lightdm:auth): requirement "user ingroup nopasswdlogin" not met by user "ourbox1"
Apr 13 10:17:01 ourbox1 CRON[3000]: pam_unix(cron:session): session opened for user root by (uid=0)
Apr 13 10:17:01 ourbox1 CRON[3000]: pam_unix(cron:session): session closed for user root
Apr 13 10:18:44 ourbox1 compiz: gkr-pam: unlocked login keyring
```

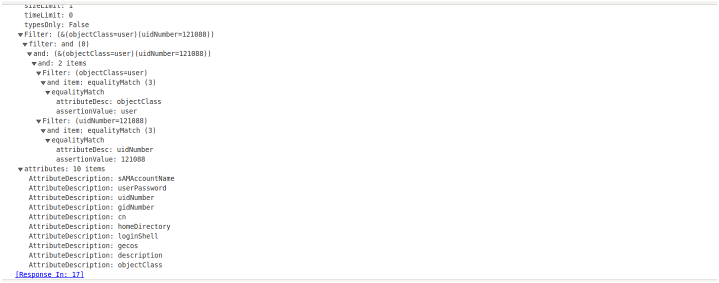
- pam_winbindd vedlikeholder enlokal db for UID. (kanskje nss_ldap ikke vil fungere riktig mot AD)
- getent
- A**fter the meeting with Erik:**
    - We found out that we have configure most of it right.
    - We can contact the AD server and we get authenticated.
    - What our problem is:
        - When you log in you don't stay logged in, Erik thinks the problem is with how we do the mappings. Since we don't find the user in "Getent" and we need to find out what is getting sent over the network.
          Also the local host cant fin the user in PASSWD, which we think is a result of that we find the SAMACOUNTNAME on the ad but that isnt the one we want (?).
          And we need to check what pam_mkhomedir.so excactly does.
          Since when we log in we get unkown user, when it tries different .so files such as pam_mkhomedir.so.
          Since "getent passwd" will contain the user entry in the passwd file with the useranme, user id and home dir.
          But our user on the AD is not in this file, so he thinks that it can't catalog/find the user
    - We are going to check what is being sendt over the network with wireshark.
    - We also might have to install another package, winbind. Check pam_ldap winbind and nss winbind.
- Wireshark: **(ip.src==10.10.0.239 || ip.dst==10.10.0.239) && tcp.port==389**
- If we enter: **getent passwd 121088**, we get no results. the search is for **uidNumber=121088**. There are no attributes named uidNumber in the AD, but there is one named **msSFU30UidNumber**. If we only could get passed to search for this **sAMAccountName** instead.

-
-
-

  Wireshark:



```
No.     Time         Source          Destination     Protocol Length Info
     10 2.035995000  10.10.0.239     128.39.140.10   TCP       74 41825 > ldap [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3717284 TSecr=0 WS=128
     11 2.037488000  128.39.140.10   10.10.0.239     TCP       78 ldap > 41825 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 WS=1 TSval=0 TSecr=0 SACK_PERM=1
     12 2.037547000  10.10.0.239     128.39.140.10   TCP       66 41825 > ldap [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3717285 TSecr=0
     13 2.037665000  10.10.0.239     128.39.140.10   LDAP     132 bindRequest(1) "cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no" simple
     14 2.040798000  128.39.140.10   10.10.0.239     LDAP      88 bindResponse(1) success
     15 2.040848000  10.10.0.239     128.39.140.10   TCP       66 41825 > ldap [ACK] Seq=67 Ack=23 Win=29312 Len=0 TSval=3717286 TSecr=22510778
     16 2.041041000  10.10.0.239     128.39.140.10   LDAP     277 searchRequest(2) "ou=student,dc=hig,dc=no" wholeSubtree
     17 2.042554000  128.39.140.10   10.10.0.239     LDAP      88 searchResDone(2) success  [0 results]
     18 2.042743000  10.10.0.239     128.39.140.10   LDAP      73 unbindRequest(3)
     19 2.042772000  10.10.0.239     128.39.140.10   TCP       66 41825 > ldap [FIN, ACK] Seq=285 Ack=45 Win=29312 Len=0 TSval=3717286 TSecr=22510778
     20 2.044146000  128.39.140.10   10.10.0.239     TCP       66 ldap > 41825 [ACK] Seq=45 Ack=286 Win=65251 Len=0 TSval=22510778 TSecr=3717286
     21 2.044180000  128.39.140.10   10.10.0.239     TCP       66 ldap > 41825 [FIN, ACK] Seq=45 Ack=286 Win=65251 Len=0 TSval=22510778 TSecr=3717286
     22 2.044202000  10.10.0.239     128.39.140.10   TCP       66 41825 > ldap [ACK] Seq=286 Ack=46 Win=29312 Len=0 TSval=3717286 TSecr=22510778
```

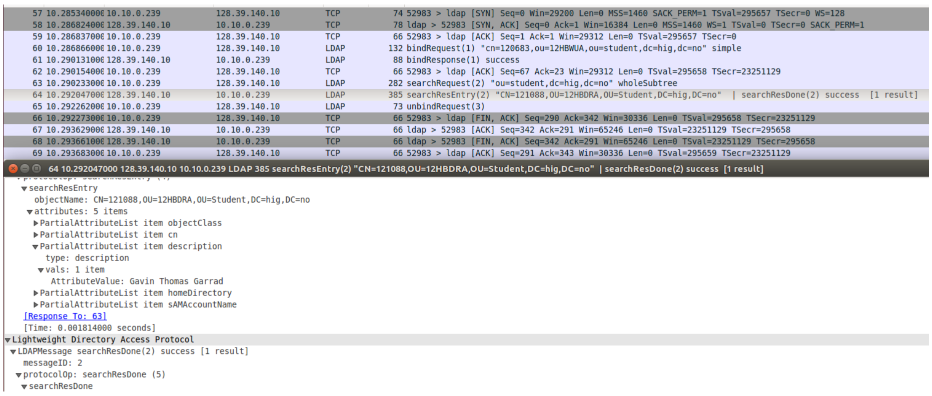


```
    sizeLimit: 1
    timeLimit: 0
    typesOnly: False
  ▼Filter: (&(objectClass=user)(uidNumber=121088))
   ▼filter: and (0)
    ▼and: (&(objectClass=user)(uidNumber=121088))
     ▼and: 2 items
      ▼Filter: (objectClass=user)
       ▼and item: equalityMatch (3)
        ▼equalityMatch
           attributeDesc: objectClass
           assertionValue: user
      ▼Filter: (uidNumber=121088)
       ▼and item: equalityMatch (3)
        ▼equalityMatch
           attributeDesc: uidNumber
           assertionValue: 121088
  ▼attributes: 10 items
     AttributeDescription: sAMAccountName
     AttributeDescription: userPassword
     AttributeDescription: uidNumber
     AttributeDescription: gidNumber
     AttributeDescription: cn
     AttributeDescription: homeDirectory
     AttributeDescription: loginShell
     AttributeDescription: gecos
     AttributeDescription: description
     AttributeDescription: objectClass
 [Response In: 17]
```

- But when we enter: **getent shadow 121088**, we get 1 result. the search is for sAMAccountName=121088:
  CLI returns: **121088:*:15578::99999::::0**
  Wireshark:

14.04.15

- Interesting: When we enter **id 121088**, the PC takes a LDAP search **sAMAccountName=121088**, and the LDAP server response is successful with alot of information about this user. However, the prompt returns that there are no users with **id=121088**.



```
57 10.28534000 10.10.0.239    128.39.140.10  TCP    74 52983 > ldap [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=295657 TSecr=0 WS=128
58 10.28682400 128.39.140.10  10.10.0.239    TCP    78 ldap > 52983 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 WS=1 TSval=0 TSecr=0 SACK_PERM=1
59 10.28683700 10.10.0.239    128.39.140.10  TCP    66 52983 > ldap [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=295657 TSecr=0
60 10.28686600 10.10.0.239    128.39.140.10  LDAP  132 bindRequest(1) "cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no" simple
61 10.29013100 128.39.140.10  10.10.0.239    LDAP   88 bindResponse(1) success
62 10.29015400 10.10.0.239    128.39.140.10  TCP    66 52983 > ldap [ACK] Seq=67 Ack=23 Win=29312 Len=0 TSval=295658 TSecr=23251129
63 10.29023300 10.10.0.239    128.39.140.10  LDAP  282 searchRequest(2) "ou=student,dc=hig,dc=no" wholeSubtree
64 10.29204700 128.39.140.10  10.10.0.239    LDAP  385 searchResEntry(2) "CN=121088,OU=12HBDRA,OU=Student,DC=hig,DC=no"  | searchResDone(2) success  [1 result]
65 10.29226200 10.10.0.239    128.39.140.10  LDAP   73 unbindRequest(3)
66 10.29227300 10.10.0.239    128.39.140.10  TCP    66 52983 > ldap [FIN, ACK] Seq=290 Ack=342 Win=30336 Len=0 TSval=295658 TSecr=23251129
67 10.29362900 128.39.140.10  10.10.0.239    TCP    66 ldap > 52983 [ACK] Seq=342 Ack=291 Win=65246 Len=0 TSval=23251129 TSecr=295658
68 10.29366100 128.39.140.10  10.10.0.239    TCP    66 ldap > 52983 [FIN, ACK] Seq=342 Ack=291 Win=65246 Len=0 TSval=23251129 TSecr=295658
69 10.29368300 10.10.0.239    128.39.140.10  TCP    66 52983 > ldap [ACK] Seq=291 Ack=343 Win=30336 Len=0 TSval=295659 TSecr=23251129
```

64 10.292047000 128.39.140.10 10.10.0.239 LDAP 385 searchResEntry(2) "CN=121088,OU=12HBDRA,OU=Student,DC=hig,DC=no" | searchResDone(2) success [1 result]

```
   ▼searchResEntry
      objectName: CN=121088,OU=12HBDRA,OU=Student,DC=hig,DC=no
     ▼attributes: 5 items
       ▶PartialAttributeList item objectClass
       ▶PartialAttributeList item cn
       ▼PartialAttributeList item description
          type: description
         ▼vals: 1 item
            AttributeValue: Gavin Thomas Garrad
       ▶PartialAttributeList item homeDirectory
       ▶PartialAttributeList item sAMAccountName
    [Response To: 63]
    [Time: 0.001814000 seconds]
▼Lightweight Directory Access Protocol
 ▼LDAPMessage searchResDone(2) success [1 result]
    messageID: 2
  ▼protocolOp: searchResDone (5)
   ▼searchResDone
```

- https://cdc.iseage.org/tutorial-pam-ldap-authentication-active-directory-freebsd/
  - override
  - *#Uncomment the following line to override the default login shell*

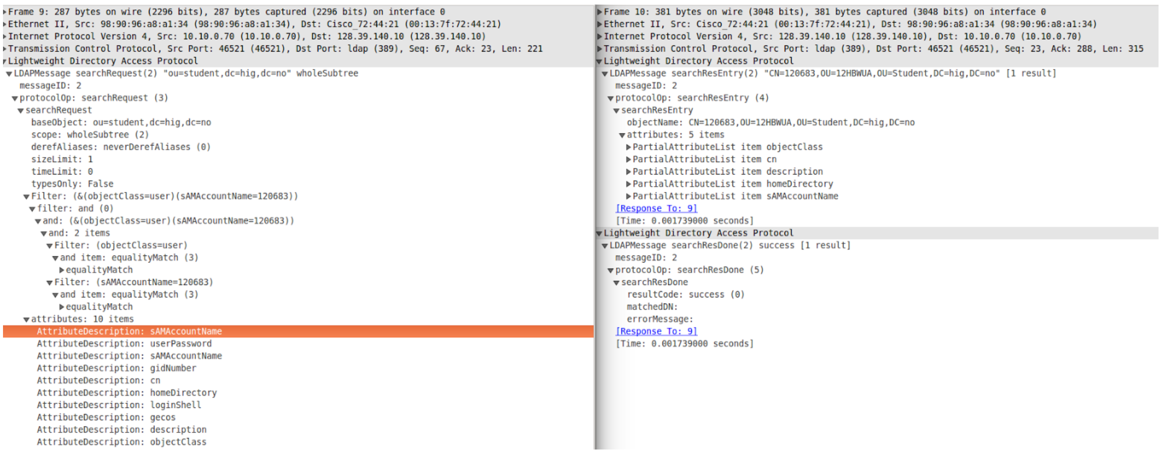
    #nss_override_attribute_value loginShell /usr/local/bin/bash
- John said that we need to figure out how we can map the uidNumber on the host to get it's value from **msSFU30UidNumber** in the AD.
  - This could be done in the ldap.conf file.
  - Also we need to override the loginShell:
- Repport:
  - http://en.wikipedia.org/wiki/Virtualization Research.

15.04.15

- How PAM perform an authentication and the steps alongside: http://en.wikipedia.org/wiki/Linux_PAM

16.04.15

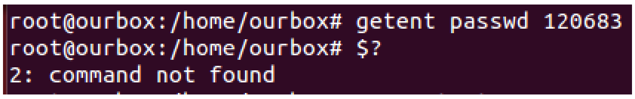
- Trying to map the right attributes in ldap.conf when using **getent passwd 120683**
  - Interesting, when we write **getent passwd ourbox**, then it searches locally for the username, but when we search in the LDAP directory, then this attribute is named **uidNumber**.
  - In the ldap.conf we added this line **nss_map_attribute uidNumber sAMAccountName** and now using Wireshark, we get success with the lookup! However, the command returns nothing. The request we send with the **getent passwd 120683** command is asking the LDAP server for 10 items, will we only receive 5 attributes:



```
▸Frame 9: 287 bytes on wire (2296 bits), 287 bytes captured (2296 bits) on interface 0
▸Ethernet II, Src: 98:90:96:a8:a1:34 (98:90:96:a8:a1:34), Dst: Cisco_72:44:21 (00:13:7f:72:44:21)
▸Internet Protocol Version 4, Src: 10.10.0.70 (10.10.0.70), Dst: 128.39.140.10 (128.39.140.10)
▸Transmission Control Protocol, Src Port: 46521 (46521), Dst Port: ldap (389), Seq: 67, Ack: 23, Len: 221
▾Lightweight Directory Access Protocol
 ▾LDAPMessage searchRequest(2) "ou=student,dc=hig,dc=no" wholeSubtree
    messageID: 2
  ▾protocolOp: searchRequest (3)
   ▾searchRequest
      baseObject: ou=student,dc=hig,dc=no
      scope: wholeSubtree (2)
      derefAliases: neverDerefAliases (0)
      sizeLimit: 1
      timeLimit: 0
      typesOnly: False
    ▾Filter: (&(objectClass=user)(sAMAccountName=120683))
     ▾filter: and (0)
      ▾and: (&(objectClass=user)(sAMAccountName=120683))
       ▾and: 2 items
        ▾Filter: (objectClass=user)
         ▾and item: equalityMatch (3)
          ▸equalityMatch
        ▾Filter: (sAMAccountName=120683)
         ▾and item: equalityMatch (3)
          ▸equalityMatch
    ▾attributes: 10 items
       AttributeDescription: sAMAccountName
       AttributeDescription: userPassword
       AttributeDescription: sAMAccountName
       AttributeDescription: gidNumber
       AttributeDescription: cn
       AttributeDescription: homeDirectory
       AttributeDescription: loginShell
       AttributeDescription: gecos
       AttributeDescription: description
       AttributeDescription: objectClass
```

```
▸Frame 10: 381 bytes on wire (3048 bits), 381 bytes captured (3048 bits) on interface 0
▸Ethernet II, Src: Cisco_72:44:21 (00:13:7f:72:44:21), Dst: 98:90:96:a8:a1:34 (98:90:96:a8:a1:34)
▸Internet Protocol Version 4, Src: 128.39.140.10 (128.39.140.10), Dst: 10.10.0.70 (10.10.0.70)
▸Transmission Control Protocol, Src Port: ldap (389), Dst Port: 46521 (46521), Seq: 23, Ack: 288, Len: 315
▾Lightweight Directory Access Protocol
 ▾LDAPMessage searchResEntry(2) "CN=120683,OU=12HBWUA,OU=Student,DC=hig,DC=no" [1 result]
    messageID: 2
  ▾protocolOp: searchResEntry (4)
   ▾searchResEntry
      objectName: CN=120683,OU=12HBWUA,OU=Student,DC=hig,DC=no
     ▾attributes: 5 items
      ▸PartialAttributeList item objectClass
      ▸PartialAttributeList item cn
      ▸PartialAttributeList item description
      ▸PartialAttributeList item homeDirectory
      ▸PartialAttributeList item sAMAccountName
    [Response To: 9]
    [Time: 0.001739000 seconds]
▾Lightweight Directory Access Protocol
 ▾LDAPMessage searchResDone(2) success [1 result]
    messageID: 2
  ▾protocolOp: searchResDone (5)
   ▾searchResDone
      resultCode: success (0)
      matchedDN:
      errorMessage:
    [Response To: 9]
    [Time: 0.001739000 seconds]
```

  - So there are some attributes the command needs that it’s not getting. We can check if a command was run successful or had any errors by entering **$?** in the CLI.



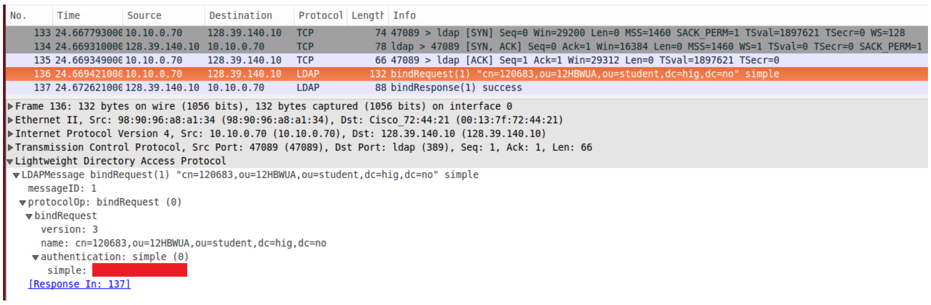
```
root@ourbox:/home/ourbox# getent passwd 120683
root@ourbox:/home/ourbox# $?
2: command not found
```

In the man page for error code for getent, the code 2 indicates:

“2 - One or more supplied key could not be found in the database.”

- ○ http://www.davidpashley.com/articles/ldap-basics/
    - ■ an article about ldap basics, this is where we read that **“Another common use for LDAP is authentication of user accounts. For this, we can use theposixAccount class. This is an auxiliary class and adds cn, uid, uidNumber, gidNumber andhomeDirectory mandatory attributes and userPassword, loginShell, gecos and description as optional attributes. Because posixAccount is auxiliary, we can add it to our person object for people we want to be able to authenticate.”**

- ● Password: When we use no encryption when we authenticating a user against the LDAP server, the password is sent in plaintext, on hexadecimal form.



```
No.   Time         Source         Destination    Protocol Length Info
  133 24.66779300( 10.10.0.70     128.39.140.10  TCP          74 47089 > ldap [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1897621 TSecr=0 WS=128
  134 24.66931000( 128.39.140.10  10.10.0.70     TCP          78 ldap > 47089 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 WS=1 TSval=0 TSecr=0 SACK_PERM=1
  135 24.66934900( 10.10.0.70     128.39.140.10  TCP          66 47089 > ldap [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=1897621 TSecr=0
  136 24.66942100( 10.10.0.70     128.39.140.10  LDAP        132 bindRequest(1) "cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no" simple
  137 24.67262100( 128.39.140.10  10.10.0.70     LDAP         88 bindResponse(1) success

▶Frame 136: 132 bytes on wire (1056 bits), 132 bytes captured (1056 bits) on interface 0
▶Ethernet II, Src: 98:90:96:a8:a1:34 (98:90:96:a8:a1:34), Dst: Cisco_72:44:21 (00:13:7f:72:44:21)
▶Internet Protocol Version 4, Src: 10.10.0.70 (10.10.0.70), Dst: 128.39.140.10 (128.39.140.10)
▶Transmission Control Protocol, Src Port: 47089 (47089), Dst Port: ldap (389), Seq: 1, Ack: 1, Len: 66
▼Lightweight Directory Access Protocol
 ▼LDAPMessage bindRequest(1) "cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no" simple
    messageID: 1
  ▼protocolOp: bindRequest (0)
   ▼bindRequest
      version: 3
      name: cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no
    ▼authentication: simple (0)
       simple: [redacted]
    [Response In: 137]
```

- ● We got getent to return some values:
    - ○ we also commented something out such as:
        - ■ homeDirectory
        - ■ shadowLastChange
        - ■ posixGroup

        - uniqueMember
    - what we put in was what getent was asking for in our RFC 2307 (AD) mappings
        - **nss_map_attribute gidNumber primaryGroupID**
        - **nss_map_attribute gecos description**
        - **nss_override_attribute_value loginShell /bin/bash**
    - when we did our getent passwd 121088 we got back:
        - **121088:*:121088:513:Gavin Thomas Garrad:\\moa.stud.hig.no\home\121088:/usr/local/bin/bash**
    - When we added the nss_map_attribute we got the host to create the "\\moa.stud.hig.no\home\121088" folder in / folder.

17.04.15

- If we must reinstall the OS:

```
# RFC 2307 (AD) mappings
nss_map_attribute gidNumber primaryGroupID
nss_map_attribute uidNumber sAMAccountName
nss_override_attribute_value loginShell /bin/bash
nss_map_attribute gecos description

nss_override_attribute_value homeDirectory /home/

nss_map_objectclass posixAccount user
#nss_map_objectclass shadowAccount user
nss_map_attribute uid sAMAccountName
#nss_map_attribute homeDirectory unixHomeDirectory
nss_map_attribute shadowLastChange pwdLastSet
nss_map_objectclass posixGroup group
nss_map_attribute uniqueMember member
pam_login_attribute sAMAccountName
pam_filter objectclass=User
pam_password ad
```

- To get the login screen (lightdm) to not show a list of users, which it will do as a standard for ubuntu, we need to add some customization:
    - We need to add a file in /etc/lightdm/
        - just add lightdm.conf there.
            - Then put in the lines:

        - **[SeatDefaults]**
          greeter-hide-users=true
          greeter-show-manual-login=true
    - greeter-hide-users=true
        - This makes so that you don't get a list of users as you usually do.
    - **greeter-show-manual-login=true**
        - this makes it possible to enter a username and a password.
    - Sources:
        - http://www.tejasbarot.com/2014/04/25/hide-users-login-as-other-user-from-login-screen-ubuntu-14-04-lts-trusty-tahr/#axzz3XYz25ual
        - https://wiki.ubuntu.com/LightDM
- Trying to reconfigure everything with **libpam-ldapd**
    - **which contains packages ldap-utils libnss-ldapd libpam-ldapd nscd nslcd**
        - **http://arthurdejong.org/nss-pam-ldapd/nslcd.conf.5**
        - **http://linux.web.cern.ch/linux/docs/account-mgmt.shtml**
        -

20.04.15

- Got NSLCD to work(This is the /etc/nslcd.conf.conf), with:
    - **# /etc/nslcd.conf**
    - **# nslcd configuration file. See nslcd.conf(5)**
    - **# for details.**
    -
    - **# The user and group nslcd should run as.**
    - **uid nslcd**
    - **gid nslcd**
    -
    - **# The location at which the LDAP server(s) should be reachable.**
    - **uri ldap://128.39.140.10/**
    -
    - **# The search base that will be used for all queries.**
    - **base ou=student,dc=hig,dc=no**
    - **# The LDAP protocol version to use.**

```
○ #ldap_version 3
○
○ # The DN to bind with for normal lookups.
○ binddn cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no
○ bindpw Ourbox92
○
○ # The DN used for password modifications by root.
○ #rootpwmoddn cn=admin,dc=example,dc=com
○
○ # SSL options
○ ssl off
○ #tls_reqcert never
○
○ # The search scope.
○ scope sub
○
○ # Mappings for Active Directory
○ pagesize 1000
○
○
○ filter passwd (&(objectClass=user))
○ map    passwd uid              sAMAccountName
○ map    passwd gidNumber            primaryGroupID
○ map    passwd homeDirectory       "/home/$sAMAccountName"
○ map    passwd gecos          description
○ map    passwd loginShell   "/bin/bash"
○ map    passwd uidNumber            msSFU30UidNumber
○ filter shadow (&(objectClass=user))
```

- **map shadow uid sAMAccountName**
- **map shadow shadowLastChange pwdLastSet**

- Changed **mappasswd uidNumber msSFU30UidNumber**
  - so that i takes sAMAccountName so that we also get the getent.
- If we want to let LDAP execute our own bash scripts and sending the username as a parameter, we can use e.g. **auth require pam_exec.so "/script.sh" $PAM_USER**
- Create a group with group ID 513 and name 513:

  groupadd -g 513 513
- Trying to find another way to start fluxbox, since when you start fluxbox from the login menu and choose it the fuctions work. But what i want is that fluxbox window manager starts when you log in.
  - https://wiki.ubuntu.com/CustomXSession
  - http://www.fluxbox.org/help/man-fluxbox.php
- Been checking those sides, and it says something about changing the .xinitrc file, or make it, but it did not work.
  - http://vsido.org/index.php?topic=852.0
    - this might be an intresting read.
- i found out how to make fluxbox the default display manger, with this
  - https://wiki.ubuntu.com/LightDM
    - [SeatDefaults]
      user-session=name
    - There is something to do with this also:
    - update-alternatives --install "/usr/bin/x-session-manager" "x-window-manager" "/usr/binstartfluxbox" 2
    - update-alternatives --config x-window-manager

21.04.15

- Removed some functonality in fluxbox, in the folder /etc/X11/fluxbox and in files “keys” and fluxbox-menu.
    - in keys i removed the functionality to open terminal in “keys”
        - # open a terminal
        - Mod1 F1 :Exec x-terminal-emulator
        - # open a dialog to run programs
        - Mod1 F2 :Exec fbrun
    - in fluxbox-menu i removed most of the applications so that nothing other than virtualbox can be opened.
- How to change the startup script is in /usr/bin/startfluxbox
- remove unity:
    - sudo apt-get remove unity-lens-music unity-lens-applications unity-greeter unity-common unity-asset-pool unity-2d-launcher unity-2d libunity-misc4 libunity-2d-private0 gir1.2-unity-4.0
        - Not sure if this totaly works yet.
            - http://www.geek.com/chips/dont-uninstall-ubuntu-just-change-the-interface-1542514/2/
        - This did not fully work
    - trying this also:
      sudo apt-get remove unity unity-asset-pool unity-control-center unity-control-center-signon unity-gtk-module-common unity-lens* unity-services unity-settings-daemon unity-webapps* unity-voice-service
        -

##
#

- Install everything:
  - echo "ourbox ALL=(ALL) NOPASSWD:ALL" >> /etc/sudoers;
  - apt-get install -y **virtualbox fluxbox**; apt-get upgrade -y; apt-get install -y **libpam-ldapd;**
    - Full in the needed param + copy the **/etc/nslcd.conf** conf file from below:

```
# /etc/nslcd.conf
# nslcd configuration file. See nslcd.conf(5)
# for details.

# The user and group nslcd should run as.
uid nslcd
gid nslcd

# The location at which the LDAP server(s) should be reachable.
uri ldap://128.39.140.10/

# The search base that will be used for all queries.
base ou=student,dc=hig,dc=no
# The LDAP protocol version to use.
#ldap_version 3

# The DN to bind with for normal lookups.
binddn cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no
bindpw Ourbox92

# The DN used for password modifications by root.
#rootpwmoddn cn=admin,dc=example,dc=com

# SSL options
ssl off
#tls_reqcert never

# The search scope.
scope sub

# Mappings for Active Directory
pagesize 1000


filter passwd (&(objectClass=user))
map         passwd uid              sAMAccountName
map         passwd gidNumber        primaryGroupID
map         passwd homeDirectory    "/home/$sAMAccountName"
map         passwd gecos            description
map         passwd loginShell       "/bin/bash"
map           passwd uidNumber          msSFU30UidNumber
filter shadow (&(objectClass=user))
map           shadow uid                sAMAccountName
map           shadow shadowLastChange pwdLastSet
```

  - echo "**session required pam_mkhomedir.so skel=/etc/skel/ umask=0077**" >> /etc/pam.d/common-session
  - echo -e **"[SeatDefaults]\n greeter-hide-users=true\n greeter-show-manual-login=true**" > /etc/lightdm/lightdm.conf
  - Create a group with group ID 513 and name 513: **groupadd -g 513 513**
  - Install a ubuntu.vdi and make the group 513 RX permissions:

    setfacl -m g:513t:rx ubuntu.vdi
  - Download the defaultstartupscript and put it in /etc/profile.d/

22.04.15

- **Creating a demo!**
- The OS files in Desktop/os/ folder must not contain any white spaces. Cause when we search for every OS.vdi file, we search for the 9th column (a new column would be created for every white space):
  - Get the fullname of the file:

    ls -l /home/ourbox/Desktop/os/*.vdi | awk '{print $9}'
  - Only get the name of the file:

    ls -l /home/ourbox/Desktop/os/ | awk '{print $9}' | grep ".vdi"
  - Only display the image name without .vdi:

    ls -l /home/ourbox/Desktop/os/ | awk '{print $9}' | grep ".vdi" | cut -d "." -f 1
- Get input from the user in bash script:

  admin=$(zenity --entry --text="What is the hostname of the administrator?");
- Using sed to change the value of the admin variable in a file:

  admin=$(zenity --entry --text="What is the hostname of the administrator?");

  sed -i "/admin=/c\admin=\"$admin\";" verifyNewOSes.sh

23.04.15

- First we had the script which goes inside the /etc/profil.d/ inside the runMe.sh file, but for making it more easyer to read (we need alot of “\” when the script is in the runMe.sh for escaping variables), we puted this code inside a separate file named defaultVMScript.sh. When we copy this file over into /etc/profile.d/, the script will not be run (has nothing to do with making it executable). It’s like the file is corrupt in some way, but I tried to create another file and pass the code into that one, but it just won’t work.
    - After the installation of the runMe.sh script, if i take **cat /etc/profile.d/defaultVMScript.sh > /etc/profile.d/defaultVMs.sh**, then it works. It’s like there are some format error or a corrupt file(but I can perform rw), no idea what’s wrong…
    - **SOLUTION:** It doesn’t work to use “cp $path/defaultVMScript.sh /etc/profile.d/;” We must use:
      cat $path/defaultVMScript.sh > /etc/profile.d/defaultVMScript.sh;
- What we need to do:
    - Let every created VM be created with NIC configured to **bridged mode.**
    - Take in use **Kerberos** and make LDAP authentication encrypted.
- Bridge - mode:
    - List a lot of detailed information about the VM’s configuration:
      VBoxManage showvminfo “name”
    - https://forums.virtualbox.org/viewtopic.php?f=7&t=45911
        - Her sto det noe som kanskje kan være nyttig.
            - Jepp, det gjorde det! :D

24.04.15

- Let every created VM be created with NIC configured to **bridged mode.**
    - This is done by adding this line:
      VBoxManage modifyvm “NAME --nic1 bridged --bridgeadapter1 eth1

        - Notice that when configuring the VM to bridged mode, we must use **eth1** or higher, eth0 will not work.
- Installing VirtualBox-4.3 version with extentson pack. A script has been made. Source:
    - Install VirtuallBox: https://www.virtualbox.org/wiki/Linux_Downloads
    - Install extension pack: https://www.howtoforge.com/vboxheadless-running-virtual-machines-with-virtualbox-4.3-on-a-headless-ubuntu-14.04-lts-server
- VBoxManage modifyvm “navnpåvm” --draganddrop bidirectional
- VBoxManage modifyvm “navnpåvm” --clipboard bidirectional
  clipboard needs guest additions installed on the vm.

27.04.15

- Continuing writing the report.
    - **Stepan**: Authentication.
    - **Gavin**: Desktop Environment
- http://texblog.org/2011/06/11/latex-syntax-highlighting-examples/ for code. in the latex.

29.04.15

- Continuing writing on the report
    - **Stepan:** Authentication:Understanding LDAP files
    - **Gavin:** Desktop Environment, Hypervisor

30.04.15

- Continuing writing on the report
    - **Stepan:** Authentication: Understanding LDAP files
    - **Gavin:** Desktop Environment, Hypervisor

01.05.15

- Continuing writing on the report
    - **Stepan:** Authentication: Understanding LDAP files

- **Gavin:** Desktop Environment, hypervisor.

A.1 Meetings

14.01.2015

Referat fra møtet med Thomas (09:30-10:15):

- Snapshot skal være mulig for alle elevene
- Hva elevene skal møte på nr PCen booter opp er ikke gitt, kom med forslag.
- Ikke nødvendig, men til slutt hadde det vært fint om det hadde vært muig å boote opp en VM via VPN.
- snakk med John, han har ideen bak dette. Mulighet med virtualbox.
- Alt handler om selve Supervisoren og begrensningene der.
- Søk nettet om fordeler og ulemper, tilfredstiller løsningen kravene?
- Til slutt skal det være mulig at læreren skal kunne sitte på et sted og fyre opp VMer fra en PC, dette er for å spare tid med å installere et VM fysisk på hver PC (tidsfordriv).
- Noen andre kan lage interfacet.
- Hva slags supervisor?
- hver gang eleven starter PCen, skal alt være base configurert?
- trenger ikke "host-eller-noe-slikt", ved mindre det har større fordeler.
- På slutten av Bachelor oppgaven, skal det være alle de andre stegene som kreves for å fortsette for å få systemet 100% opp og fungerende.
- 4 february etter kl. 13 er det ikke mulig å komme inn.
- 9:30 onsdager blir det faste møter med Thomas. Uken før og etter påsken.
- skrive kravspesifisering til neste gang.

21.01.2015

<u>Referat fra møtet med Thomas:</u>

- Gå gjennom spec. krav og se hva slags programvarer som oppfyller hva slags krav.
- Send spec. krav til Thomas.
- Hvis noe KOSTER NOEN TUSEN KRON ER FO REN LISENS, SÅ GÅR DET GREIT. MEN ELLER NEI.
- 1-2 ukene: Lag pros/cons av programvare
  - lage timetable
  - Søke gjennom hva som finnes på markedet
  - start på sammenligning
- Vi må ta også hensyn til om servisen komme rtil å være
- slutten av april skal vi være ferdig med det fysiske, også skal vi bruke 2 uker til å skrive ferdg rapporten.
- Når vi skriver på en ting, men plutselig str vi fast, så skriv på noe annet, og fordele oppgaven.
- Før påske burde vi prøve og feil, så etter det må vi skrive en del og få ting til å funke i rapporten.
- VI må finne ut en løsning for hvordan vi skla a i bruk LDAP til authorisering.
- Authoriseringen hadde vært fint om det var mulig med med gjeste grupper, slik at folk som ikke er studenter her, får også tilgang.
- John sier (mener Thomas) at VirtualBox kan tilfredstille mange av kravene.
- I løpet av februar, så burde vi være ferdig med hvordan alt skal være, og klare til å starte med å jobbe fysisk.
- Thomas sier at vi minimum skla ha 2 NIC, men hvis flere trengs, så er ikke det et problem.

- 802.1q (Cisco standard ta no slag) se hva det innebærer, for han ønsker at NIC kommer til å ha støtte for det.

28.01.15

Referat fra møte:

- Angående 1.del innleveringen vår:
  - Målet med denne oppgaven er ikke å bygge store nettverk, men til en senere anledning skal det være mulig (Second step). Hoved poenget i dette stedet er så ha en hypervisor som er avgrenset med grupperettigheter.
  - taskdescription as bulletpoints. Frode skulle vist ha noen lynkurs i hvordan man laget kravspec.

- Det må være mulig å copy paste mellom de forskjellige VMene.
- 10.februar skal vi være ferdig med kravspec og hvordan det skal være.
- det enkleste er hvis vi bruker linux og KVM, local group policy, så kommer det policy rules bli kopiert videre til alle andre hostene.
- Erik foreslår at vi heller har egen policy grupper som blir gitt ut til elevene, passord og brukernavn.
- Erik foreslår at jo fortere jo bedre at vi prøver å sette opp det fysiske og ser hva som er mulig og ikke, og hvordan teknologien fungerer, eller hvor grensene går.
- Bridge mode mellom VMene.

04.02.14

- **Møte**
  - Fortalt litt om kvm med libvirt til begge.
  - Libvirt know alot of about how this work in general.
  - libvirt.org/apps
  - There should be a mechanisem in libvirt to differ Admin and Users.
  - Snakke med Jon. ASAP the brenner på dass.
  - Erik recommends Kvm
    - libvirt deamon as a service on the host
    - user loggs in and gets a interface from libvirt
      - virtmangeager talks to the libvirt deamon
      - user has to be added to the virt manager on.
      - Virsh CCi

The quem-kvm package is more like a binary translator, which will not let KVM to talk directly with the HW, and this will slow down the VMs. Not cool.

http://wiki.qemu.org/Index.html

Kiosk mode er hva vi vil ha opp.

- hvordan skal man stoppe fra en bruker fra å krysse ut bilde når kiosk mode er oppe.

Describe it when we test it. the test.

Dont think much about HPV men mer om hvordan løsningen med libvirt og virtmanges (GUI)

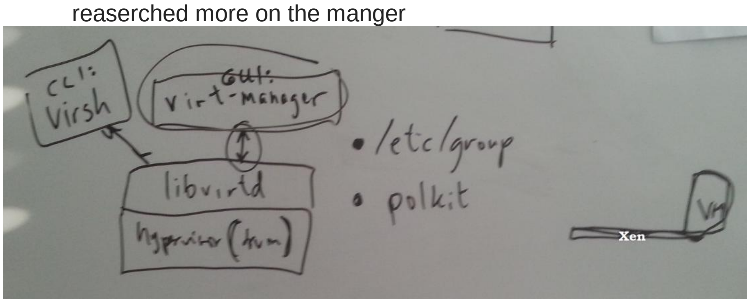
Do a sruvay on the managment managment part.

google : “local libvirt root group”

Next week:

Jon alt solutions

end of week new pc ?
reaserched more on the manger



Møte med Jon:

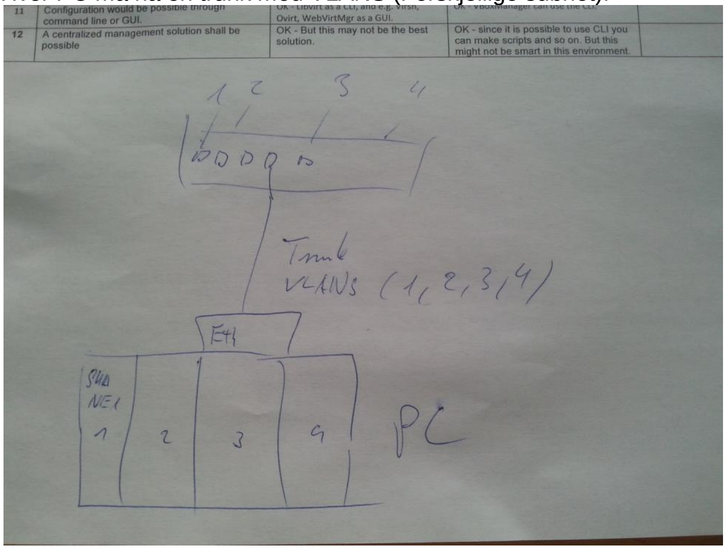
- Han mener at vi kan engelt bruke virtual box.
    - I virtual box kan man lage en virtual maskin som er en clone av en virtual maskin.
    - Da kan vi lage en root VM som da kan bli clonet som kan brukes.
    - Dropp VM ware fra tankene.
    - VM vil skape mer problemer og en del mer vedlikehold.
- Imutable disk images.
- Vi stilte dette spørsmålet om Kiosk mode med virtualbox
    - Han tror det ikke egentlig det ikke finnes.
    - Han sier at det kan være mulig å hacke det til.
- Nevnes i forhold til sikkerhet at de har tilgang til HW så sikkerhet.
- Hvorfor skal man kunne måtte starte en VM for å tilgang til putty.
- Lillemyk skal være med så må vi ha virtualisering.
- Hvis host oset er en for for linux. tre valg av teknologier. Xen og kvm og en til han ikke vil nevne.
- Utfordringen der er at de forskjellige virt mangagerne er lite standardisert de trenger som oftest mye rettiheter.
- Virtualbox er det siste valget om pakker mangerer hypwerviser og etc i en pakke.
- Xen via xem full controll gjennom ccl, libvirt med xen mister du mye funksjonalitet.
- Virtualbox du får gui og kan bruke ccl og du kan scripte.
- En vanlig måte er å lage disk imge er å bruke quem (dette er for xen og kvm)
- hvis det er sånn som at vi gjøre det som vi er nå så finnes det vm ware lab manager som har tatt 12 år og utvikle.
- Logge på med forskjellige så må vi kopiere imges til deres hjemme områder.
- Det er en viktig ting å kunne bruke hosten som en workstation.
- hvis host os er windows så funger virtual box fungerer det veldig bra.

11.02.15

Møte:

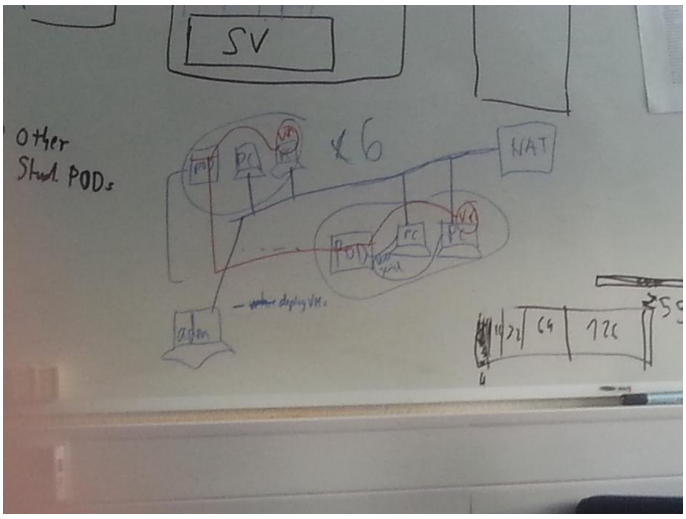
- Vi trenger ikke å tenke på at kabelen skal bli dratt ut, etc.
- Hver PC skal være tilkoblet til pod
- 12 hosts totalt sa thomas (2 per pod).
- Vi må tegne opp et nettverkskart
- Elevene skal kunne copy paste mellom VMene
- Viktig at John godkjenner nettstrukturen. Der alle PCene er koblet sammen til HIG-backbone

- Ikke at det er en del av oppgaven vår, men han ønsker at det skal være mulig til senere å configurere VMene remote.
- Hver PC må ha en trunk med VLANS (Forskjellige subnet):


11 Configuration would be possible through command line or GUI.
OK - Libvirt as a CLI, and e.g. Virsh, Ovirt, WebVirtMgr as a GUI.
OK - VboxManager can use the CLI.
12 A centralized management solution shall be possible
OK - But this may not be the best solution.
OK - since it is possible to use CLI you can make scripts and so on. But this might not be smart in this environment.
1 2 3 4
Trunk
VLANS (1,2,3,4)
Eth
Sub NEt
1 2 3 4
PC

- Viktig at vi har med imutual-images.
- Erik sier at det er viktig å gjøre hostene selv maintable, så de kommer f.eks til å slette temp. filer etter en vis tid, og updaterer seg selv, etc.
- PCene kommer til å ha 32GB ram. De har 15 000kr per PC å bruke
- Erik påstår at vi "burde" ha en SSD!
  - Erik synes det kan være også mulig å heller bestille PCer via komplett, men da har de ikke 3 års garanti avtaler med DELL.
- Det skal burde være forskjellige images fr de forskjellige service.
- Erik sier at det er bedre å bruke libvirt enn VirtualBox, fordi det kan være lettere å sette grenser og teste når du kan bruke commandline og prøve å teste via libvirt, istedenfor en hel applikasjon.
- Vi burde også fokusere på åbruke CLI når vi skal konfiguere og prøve KVM. Da lærer man mer og får en bedre oversikt over hva som er mulig og ikke.

- Til neste uke skal vi lage et nettverkskart
  - Snakke med john og få det godkjent
  - Finn ut hvordan vi via CLI kan foreta configurasjon på KVM.
  - Finn John og spør han hvorfor han foretrekker VirtualBox. For erik ser ikke hvorfor det er bedre enn å bruke libvirt og KVM.
  - Viktig at i oppgaven å sette klare linjer for hvorfor vi gjør og velger det vi gjør.
  - Vi må få installert KVM og Libvirt. Så skal en root-bruker opprette en VM. Deretter skal en user kunne logge seg på og starte opp den VMen, men alle endringene skal bli lagret i en egen fil. Så nestegang når en annen bruker logger seg inn, skal de kunne starte VMen, men da fra sin egen scratch, og endringene skal bli lagret til en egen fil. Slik vil alle brukere kunne starte og jobbe med et base-OS, med sine egene endringer på.
- Erik tegnet et oversikt hvordan nettverket kan se ut:



Fikk mail av Erik:

- Jeg har snakka med Jon, og er enig med han i at vi prøver virtualbo

18.02.15

Referat fra møtet:

- Til neste gang skal vi ferdiglage en liten presentasjon for å demonstrere hvordan immutable, osv. fungerer med VirtualBox og KVM.
- Johns årsak til å bruke VirtualBox:
  - Virtualbox client is known interface for students
  - har funksjonalitet for immutable
- Gjør en “**lsattr**” på immutable disken til virtualbox
  - **chattr +i** -> make the file imutable
  - ----i---
- Om to uker skal vi ha klar:
  - Noen eks. av hvordan nettverkskjema kan/ser ut i cisco rommet
    - Bruk fancy program, slik at vi har en original under, og vi kan tegne endringe rog forslag med farger oppå.
    - Google this for inspiration: “Network lab diagram”
  - Sett opp kapittler og lag struktur for hele bacheloroppgaven.
    - lag bulletpoints hvor skal hva, og sett inn diagrammer
- Hvis erik hadde satt opp systemet, ville han ha satt en manager med Forman
  - Vi kan ta en tur til terje og få visning om hvordan forman fungerer
- Thomas sa at hvis elevene selv setter opp grupperinger på PCene selv, så vil ikke det være verden undergang heller.

04.03.15

Referat fra møtet:

- Finn ut hvorfor vi ikke velger libvirt/kvm, men heller velger VirtualBox.

- Virtualbox har en mer “mature” brukergrensesnitt og er mer kjent for studentene. KVM er mye bedre og moden når det komme rtil hypervisor, men i vårt tilfelle, så vil virtualbox være mer egnet.
- Lag linux server, mailserver,(windows) og ha det tilgjengelig når Virtualbox starter.
- FInn ut om virtualbox tilfredsstiller alle kravene våre. Hvis ikke, så evt. fortelle hvorfor ikke, og hva som kan gjøres for å få det til.
- 12.april, til da burde vi ha en viss utgave av bacheloroppgaven.
- 2-3 uker før innlevering skal vi også ha en fornyet utgave som erik skal se på.
- Skaff automatic spell check for TexWorks.

11.03.15

REFERAT:

- En innlevering til erik 7.april (rett etter påsken), 1-2 before handin (rødpen)
- lage en plan for 9 ukene framover, så vi dekker alt vi skal ha med,
- Spør erik hvorfor vi har probemer med fluxbox når vi starter det i profile.d/, vi får ikke kjørt eller eksekvert noen programmer.
- Finn ut om vi trenger home directory, eller ikke. Ta hensyn til så ikke vilene blir opprettet i homedirectory på loke sin servere, men på lokale PC
- **powerbroker** for administrasjonskontrol!!!!!
- stay away of mounting the homefolder inside the cp
- Kanskje ha mulighet så en VM / mappe kan ha flere brukere, og ikke bare en (lage grupperinger på en måte).
- Husk at hvis det vil være nødvendig for at en bruker skal kunne sette grupper selv, så vil det være godkjent
- lag et skript som vil ta inn parametere, som lager og gjør at vmene kommer til å bli delt mellom de og de brukerne. Det eneste ved siden av virtualbox vinduet som skal være tilgjengelig for brukeren.
- **umask** command: setting the default access mode. Kan være til hjelp.
- Finne en måte så en bruker kan ha mulighet til å sette grupperinger selv. For egentlig så må enn være root bruker for å gjøre det.
- Brukere kan ha tilgang og setter en fil i en shared directory, der filen inneholder brukernavn som de ønsker gruppering på. Så er det crontab som kjører og sjekker hvert 5 min og exekverer filene
- MAP, enkleste måte med LDAP er “PADL software pty ltd” (**PAM_LDAP**). Spør John og få han til å gi oss LDAP mulighet. Hvis ikke, så må vi gjøre det lokalt
    - Hver PC som starter får
- Lag en presentasjon og visning til thomas om hva som fungerer og tilfredstiller kravene våre, og hva ikke, (13. april og ut burde vi ha det lagt opp klar for visning)
- Erik 21.mars - 10.april 26.mars tilgjengelig på email)
- Uke 12 er det møte.

18.03.15

Møtereferat:

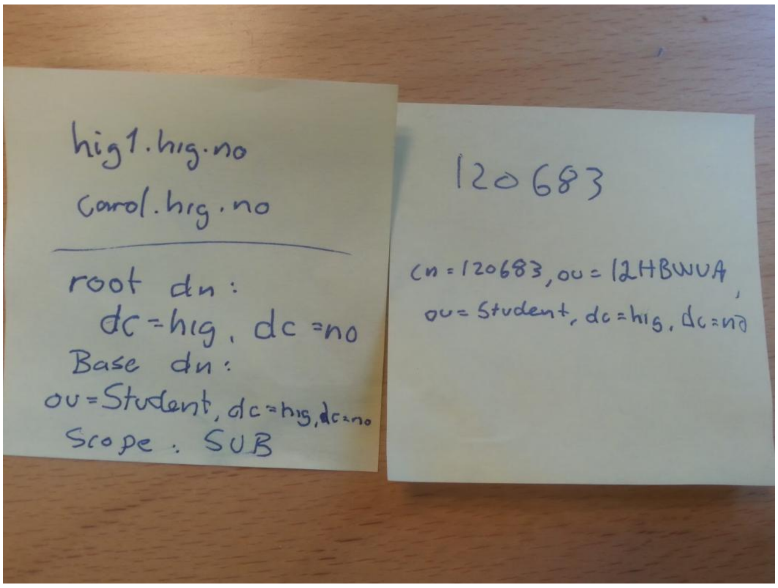
- Vi må ha testing på planen.
- Følger det gant skjemaet opp kravspecc? (Spør Erik).
- Les om user acceptance før vi gjør testing og før vi skriver om det.
    - Det burde være i henhold til kravspec.
        - Bruk Thomas, de assistentene og 2 andre random. (Maks 5)
        - 15 -20. April. burde vi gjør det.
- Vi kan ikke nå 27 mars - 10 april. Thomas.

- Møte med Erik den 10 april og møte med Thomas den 15 april. hvor vi snakker om hvordan vi skal teste.

23.03.15

REFERAT FRA MØTET MED JOHN:

- **Fant ut at ldap.hig.no er en server for enkle oppslag om brukere, som email, bilde og navn. Så vi har hittil prøvd å vertifisere passordet mot feil server de siste ukene. Årsaken var at etter et besøk hos IT-avdelingen, fikk vi beskjed om at vi bare skulle prøve oss fram. Dermed har vi klart å bruke opp 2 uker på lite fornuftig framdrift.**
- Fikk to lapper:
  - 1.note:
    Server we should connect to: **hig1.hig.no** (IP:128.39.140.7) or **carol.hig.no** (IP:128.39.140.10)
    My user account we will authenticate with: 120683
    bindDN: **cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no**
- Han sa det var to muligheter å autentisere seg, enten mot den vi fikk som gjør at vi skal bruke PAM mot Active Directory, dette betyr at vi må skreddersy PAM mye mer enn om vi da skulle ha brukt den andre servern som vi kunne koble oss i mot. Den bruker da Poisix.
- Vi kan også få mulighet til å gjøre dette med ansatte også, men da må vi filtere mer.
  - Vi fikk da gitt "treet" for studenter, så om vi skal gjøre det med ansatte også så er vi nødt til å gå tilbake til jon.



hig1.hig.no
carol.hig.no

root dn:
dc=hig, dc=no
Base dn:
ou=Student, dc=hig, dc=no
Scope : SUB

120683

cn=120683, ou=12HBWUA,
ou=Student, dc=hig, dc=no

B User Manual

Ourbox Manual

Stepan Maluchev
Gavin Thomas Garrad

May 15, 2015

Contents

1 Some instructions at first

Once you have made an .vdi image, remember to turn of the auto update, since if there is a change to the original .vdi image that resides on the desktop of the administrator and other users have work saved to the differential file. The differential file will be broken, so when you need to do an update on the original .vdi image, you have to remove all of the users differential files.

1.1 Users

All the user will have a home folder residing in the **/home/"studentnumber"**. Within their home folder there will be a Virtualbox folder which then contains all of their differential files.

1.2 Virtualbox

If the user is unlucky and deletes the VMs that are standard for the system this won't be any problem they just need to relog, but all their work will be lost.

2 Installation

1. Step Configure your .vdi images in a hypervisor preferably Virtualbox. Configure them with the necessary applications you want. Set the .vdi images in a folder named **os**

2. Step Have the folder which contains the script to run and the necessary files to go with and the **os** folder on a flashdrive.

3. Step Install ubuntu on the host machine.

4. Step Copy these two folder into the **desktop** of the administrator. On the host machine you want the system.

5. Step Open a terminal and run: **chmod +x /path/to/runMe.sh**. Where the "/path/to" is the path to the script.

6. Step Type in terminal **./path/to/runMe.sh**. Where the "/path/to" is the path to the script.

7. Step Wait until it reboots, after reboot you should see something like

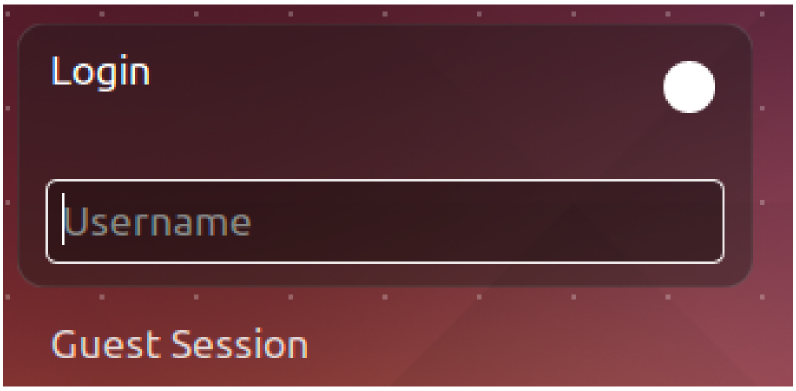


Figure 1: Unity greeter.

8. Step Before you log in, press the circle, and select "Fluxbox". Left click on Fluxbox, even if it is set as default.

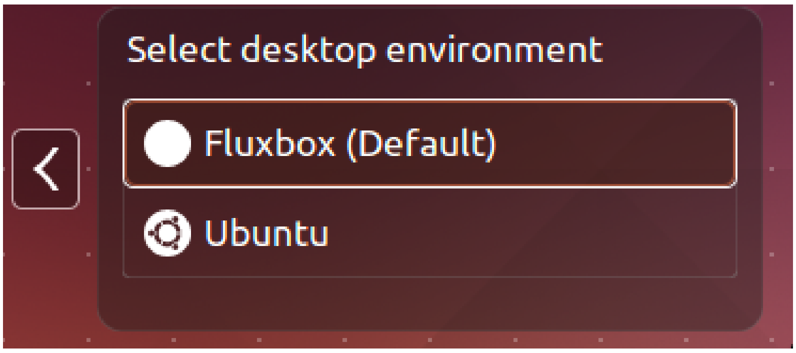


Figure 2: Valg av desktop.

9. Step Now you can log in with the administrator and you are able to have the Fluxbox desktop. Now you need to remove the choice of having the possibility to choose "Unity". Just right click the desktop, a menu should pop up and there will be a choice which says: **Switch Environment** enter that menu and choose **Fluxbox**.

Now you have fully installed the system, you won't be able to log into the "Unity" aka Ubuntu but only Fluxbox. Same with the users.

3 Switch the Desktop

If there is anything you as an administrator needs to be done, what you do to get back to the regular Ubuntu desktop.

1. Step Log on as the administrator.

2. Step Right click on the desktop

3. Step Enter the submenu **Switch Environment**

4. Step Select the **Unity**

5. Step Log out.

6. Step Left click the circle in the login prompt, it should be in the top right. See figure 1

7. Step Select Ubuntu. See figure 2

8. Step Now you you got the regular ubuntu, here you can do your configuration.

9. Step Now that you are done with what you needed to do, log out.

10. Step Do the same you did in step 6-7 except you choose **Fluxbox**.

11. Step Do the same in step 3-4 except you choose **Fluxbox**.

4 Semester maintenance

Our recommendation is to just redo section 2 installation. Since there will probably be a lot of user that has logged onto the system and then they also will have a home folder on the machine, and their user will be in the accounts on the system.

5 How to make an image

1. Step Download the desired operating system.

2. Step Open Virtualbox.

3. Step Select ”New”.

4. Step Enter desired name for the image and OS version. Click ”Next”.

5. Step Select how much memory. Preferably 2048 MB. Click ”Next”.

6. Step Select ”Create a Virtual hard drive now”. Click ”Create”.

7. Step Select ”VDI (VirtualBox Disk Image)”. Click ”Next”.

8. Step Select ”Dynamically allocated”. Click ”Next”.

9. Step Choose how much disc space the Virtual drive should have. Preferably 8 GB if it is Ubuntu or more than 30 GB if Windows.

10. Step Now you have created a .vdi image. Now you need to install the desired OS into that image. Right click VM in Virtualbox.

11. Step Choose ”Settings”.

12. Step Choose ”Storage”.

13. Step Select the option ”Controller: IDE”, when this option is selected you should see two icons, a CD with a plus icon and a hard drive with a plus icon. Select the CD with a plus icon.

14. Step A window should appear, Click ”Choose Disk”.

15. Step Navigate to the operating system you want to install. Should be a .iso file. Select it and click ”Open”.

16. Step Now you should see a new option under Controller: IDE. If you see that just Click ”ok” in the bottom right.

17. Step Now run your VM by click the arrow which says ”start”.

18. Step Install your OS.

19. Step Now that you have made the image it should be in **/home/”your folder”/Virtualbox VMs/”name of the image you made”/”name”.vdi** This is the image you should use in 2

C Code

C.1 afterInstallation.sh

```
#!/bin/bash
#This has to be run as a the user, not root.
echo -e "[begin] (fluxbox)
[include] (/etc/X11/fluxbox/fluxbox-menu)
[submenu] (Switch Environment)
        [exec] (Unity) {gnome-terminal -x sudo  mv /usr/share/ubuntu.desktop /usr/share/xsessions/ }
        [exec] (Fluxbox) {gnome-terminal -x sudo mv /usr/share/xsessions/ubuntu.desktop /usr/share/ }
[end]
[end]" > ~/.fluxbox/menu
```

C.2 runMe.sh

```bash
#!/bin/bash

admin=$(zenity --entry --text="What is the username of the administrator?");
path="/home/$admin/Desktop/firstRun";

apt-get -y update;
apt-get -y upgrade;

# Put in the admin name into the "admin" variable in the file installVirtualBox.sh
# This script will install virtualbox-4.3 and the extention pack:
sed  -i "/admin=/c\admin=\"$admin\";" $path/installVirtualBox.sh;
# making installVirtualBox.sh executable
chmod +x $path/installVirtualBox.sh;
$path/installVirtualBox.sh;

apt-get install -y fluxbox;

# Configure fluxbox:
cat $path/fluxbox-menu > /etc/X11/fluxbox/fluxbox-menu;
cat $path/keys > /etc/X11/fluxbox/keys;
sed  -i ’s/exec fluxbox/ exec virtualbox\&\n exec fluxbox/g’ /usr/bin/startfluxbox;
echo -e "[SeatDefaults]\nuser-session=fluxbox" > /usr/share/lightdm/lightdm.conf.d/50-ubuntu.conf;
# Script adds the ability to switch from between unity and fluxbox environments
cp -r $path/.fluxbox /home/$admin/;

# Configure the greeter, which hides users and allows manual login:
echo -e "[SeatDefaults]\n greeter-hide-users=true\n greeter-show-manual-login=true \n allow-guest=false" > /etc/lightdm/lightdm.conf;

# Install LDAP + Conf:
DEBIAN_FRONTEND=noninteractive apt-get install -y libpam-ldapd;
sed -i ’s/compat/compat ldap/’ /etc/nsswitch.conf;
cat $path/nslcdConfigFile > /etc/nslcd.conf;
service nslcd restart;
echo "session required pam_mkhomedir.so skel=/etc/skel/ umask=0077" >> /etc/pam.d/common-session;

#Move tty (virtual consoles)
mv /etc/init/tty1.conf /home/$admin/Desktop
mv /etc/init/tty2.conf /home/$admin/Desktop
mv /etc/init/tty3.conf /home/$admin/Desktop
mv /etc/init/tty4.conf /home/$admin/Desktop
mv /etc/init/tty5.conf /home/$admin/Desktop
mv /etc/init/tty6.conf /home/$admin/Desktop
```

```

# Put in the admin name into the "admin" variable in the file verifyNewOSes.sh
# verifyNewOSes.sh will give all .vdi files 755 permisions.
sed  -i "/admin=/c\admin=\"$admin\";" $path/verifyNewOSes.sh;
# making verifyNewOSes.sh executable
chmod +x $path/verifyNewOSes.sh;
$path/verifyNewOSes.sh;

# Copying the ...
sed  -i "/admin=/c\admin=\"$admin\";" $path/defaultVMScript.sh;
#cp $path/defaultVMScript.sh /etc/profile.d/;
cat $path/defaultVMScript.sh > /etc/profile.d/defaultVMScript.sh;

reboot;
```

C.3 verifyNewOSes.sh

```
#!/bin/bash

admin="";

chmod 755 /home/$admin/Desktop/os;

for file in $(ls -l /home/$admin/Desktop/os/*.vdi | awk ’{ print $9}’)
do
        chmod 755 $file;
done
```

C.4 installVirtualBox.sh

```bash
#!/bin/bash

admin="";

# Get access to the repository where Virtualbox-4.3 can be downloaded from:
sudo echo "deb http://download.virtualbox.org/virtualbox/debian trusty contrib" >> /etc/apt/sources.list;

# Download the Oracle public key for apt-secure and automatically add the key:
sudo wget -q https://www.virtualbox.org/download/oracle_vbox.asc -O- | sudo apt-key add -;

# Update that host with the new key:
sudo apt-get update;

# Install virtualbox 4.3:
sudo apt-get install virtualbox-4.3;

# Install the extension pack for VirtualBox 4.3.26. This is needed for get access to the USB ports
# on the guest hosts, etc. (This is not the same as guest edition). First we redirect to /tmp/
# and download the extention pack (after a reboot, the downloaded extention file will be removed
# automatically):
wget -O  extention.vbox-extpack  http://dlc-cdn.sun.com/virtualbox/4.3.26/Oracle_VM_VirtualBox_Extension_Pack-4.3.26-98988.vbox-extpack;

# Make VirtualBox install the extension pack:
sudo VBoxManage extpack install extention.vbox-extpack;

# The user must be added to the "vboxusers" group to get full use of the pack:
sudo adduser $admin vboxusers;
```

C.5 defaultVMScript.sh

```
#!/bin/bash

admin="";
me=$(whoami);
os="";

for oses in $(ls -l /home/$admin/Desktop/os/ | awk '{print $9
     }' | grep ".vdi" | cut -d "." -f 1)
do
        os="$os $oses";
done

if [[ $me != $admin ]];then

        for eachOS in $os;do
                osRegEx="^\"${eachOS}\"";
                knownVMs=$(VBoxManage list vms | grep -e "
                     $osRegEx" | awk '{print $1}' | cut -d "\""
                      -f 2);
                DIRECTORY=~/VirtualBox\ VMs/$eachOS;

                if [[ $knownVMs != $eachOS ]];then
                        if [[ -d  $DIRECTORY ]]; then
                                rm -fr "$DIRECTORY";
                        fi
                        VBoxManage createvm --name $eachOS --
                             ostype Ubuntu_64  --register;
                        VBoxManage modifyvm $eachOS --memory
                             1024 --vram 256 --cpus 2 --usb on
                             --nic1 bridged --bridgeadapter1
                             eth1;
                        VBoxManage storagectl $eachOS --name
                             sata1 --add sata;
                        VBoxManage storageattach $eachOS --
                             storagectl sata1 --port 0 --device
                              0 --type hdd --medium /home/
                             $admin/Desktop/os/$eachOS.vdi  --
                             mtype immutable;
                fi
        done
fi
```

D nslcd.conf

```
# /etc/nslcd.conf
# nslcd configuration file. See nslcd.conf(5)
# for details.

# The user and group nslcd should run as.
uid nslcd
gid nslcd

# The location at which the LDAP server(s) should be reachable.
uri ldap://128.39.140.10/

# The search base that will be used for all queries.
base ou=student,dc=hig,dc=no
# The LDAP protocol version to use.
ldap_version 3

# The DN to bind with for normal lookups.
binddn cn=120683,ou=12HBWUA,ou=student,dc=hig,dc=no
bindpw Ourbox92

# The DN used for password modifications by root.
#rootpwmoddn cn=admin,dc=example,dc=com

# SSL options
ssl off
#tls_reqcert never

# The search scope.
scope sub

# Mappings for Active Directory
pagesize 1000


filter passwd (&(objectClass=user))
map    passwd uid              sAMAccountName
map    passwd gidNumber         "125"
map    passwd homeDirectory    "/home/$sAMAccountName"
map    passwd gecos            description
map    passwd loginShell       "/bin/bash"
map    passwd uidNumber        msSFU30UidNumber
filter shadow (&(objectClass=user))
map    shadow uid              sAMAccountName
map    shadow shadowLastChange pwdLastSet
```

E Fluxbox config

E.1 Fluxbox menu for admin

```
[begin] (fluxbox)
[include] (/etc/X11/fluxbox/fluxbox-menu)
[submenu] (Switch Environment)
        [exec] (Unity) {gnome-terminal -x sudo  mv /usr/share/ubuntu.desktop /usr/share/xsessions/ }
        [exec] (Fluxbox) {gnome-terminal -x sudo mv /usr/share/xsessions/ubuntu.desktop /usr/share/ }
[end]
[end]
```

E.2 Fluxbox keys file

```
# click on the desktop to get menus
OnDesktop Mouse1 :HideMenus
OnDesktop Mouse2 :WorkspaceMenu
OnDesktop Mouse3 :RootMenu

# scroll on the desktop to change workspaces
OnDesktop Mouse4 :PrevWorkspace
OnDesktop Mouse5 :NextWorkspace

# scroll on the toolbar to change current window
OnToolbar Mouse4 :PrevWindow {static groups} (iconhidden=no)
OnToolbar Mouse5 :NextWindow {static groups} (iconhidden=no)

# alt + left/right click to move/resize a window
OnWindow Mod1 Mouse1 :MacroCmd {Raise} {Focus} {StartMoving}
OnWindowBorder Move1 :StartMoving

OnWindow Mod1 Mouse3 :MacroCmd {Raise} {Focus} {StartResizing NearestCorner}
OnLeftGrip Move1 :StartResizing bottomleft
OnRightGrip Move1 :StartResizing bottomright

# alt + middle click to lower the window
OnWindow Mod1 Mouse2 :Lower

# control-click a window's titlebar and drag to attach windows
OnTitlebar Control Mouse1 :StartTabbing

# double click on the titlebar to shade
OnTitlebar Double Mouse1 :Shade

# left click on the titlebar to move the window
OnTitlebar Mouse1 :MacroCmd {Raise} {Focus} {ActivateTab}
OnTitlebar Move1  :StartMoving

# middle click on the titlebar to lower
OnTitlebar Mouse2 :Lower

# right click on the titlebar for a menu of options
OnTitlebar Mouse3 :WindowMenu

# alt-tab
Mod1 Tab :NextWindow {groups} (workspace=[current])
Mod1 Shift Tab :PrevWindow {groups} (workspace=[current])

# cycle through tabs in the current window
Mod4 Tab :NextTab
Mod4 Shift Tab :PrevTab

# go to a specific tab in the current window
Mod4 1 :Tab 1
Mod4 2 :Tab 2
Mod4 3 :Tab 3
Mod4 4 :Tab 4
```

```
Mod4 5 :Tab 5
Mod4 6 :Tab 6
Mod4 7 :Tab 7
Mod4 8 :Tab 8
Mod4 9 :Tab 9

# open a terminal
#Mod1 F1 :Exec x-terminal-emulator

# open a dialog to run programs
#Mod1 F2 :Exec fbrun

# volume settings, using common keycodes
# if these don't work, use xev to find out your real keycodes
176 :Exec amixer sset Master,0 1+
174 :Exec amixer sset Master,0 1-
160 :Exec amixer sset Master,0 toggle

# current window commands
Mod1 F4 :Close
Mod1 F5 :Kill
Mod1 F9 :Minimize
Mod1 F10 :Maximize
Mod1 F11 :Fullscreen

# open the window menu
Mod1 space :WindowMenu

# exit fluxbox
Control Mod1 Delete :Exit

# change to previous/next workspace
Control Mod1 Left :PrevWorkspace
Control Mod1 Right :NextWorkspace

# send the current window to previous/next workspace
Mod4 Left :SendToPrevWorkspace
Mod4 Right :SendToNextWorkspace

# send the current window and follow it to previous/next workspace
Control Mod4 Left :TakeToPrevWorkspace
Control Mod4 Right :TakeToNextWorkspace

# change to a specific workspace
Control F1 :Workspace 1
Control F2 :Workspace 2
Control F3 :Workspace 3
Control F4 :Workspace 4
Control F5 :Workspace 5
Control F6 :Workspace 6
Control F7 :Workspace 7
Control F8 :Workspace 8
Control F9 :Workspace 9
Control F10 :Workspace 10
Control F11 :Workspace 11
Control F12 :Workspace 12

# send the current window to a specific workspace
Mod4 F1 :SendToWorkspace 1
Mod4 F2 :SendToWorkspace 2
Mod4 F3 :SendToWorkspace 3
Mod4 F4 :SendToWorkspace 4
Mod4 F5 :SendToWorkspace 5
Mod4 F6 :SendToWorkspace 6
Mod4 F7 :SendToWorkspace 7
Mod4 F8 :SendToWorkspace 8
Mod4 F9 :SendToWorkspace 9
Mod4 F10 :SendToWorkspace 10
Mod4 F11 :SendToWorkspace 11
Mod4 F12 :SendToWorkspace 12

# send the current window and change to a specific workspace
Control Mod4 F1 :TakeToWorkspace 1
Control Mod4 F2 :TakeToWorkspace 2
Control Mod4 F3 :TakeToWorkspace 3
Control Mod4 F4 :TakeToWorkspace 4
Control Mod4 F5 :TakeToWorkspace 5
Control Mod4 F6 :TakeToWorkspace 6
Control Mod4 F7 :TakeToWorkspace 7
Control Mod4 F8 :TakeToWorkspace 8
Control Mod4 F9 :TakeToWorkspace 9
Control Mod4 F10 :TakeToWorkspace 10
Control Mod4 F11 :TakeToWorkspace 11
Control Mod4 F12 :TakeToWorkspace 12
```

E.3 Fluxbox startup file

```
#!/bin/sh
#
# fluxbox startup-script:
#
# Lines starting with a ’#’ are ignored.

# Change your keymap:
xmodmap "/home/ourbox/.Xmodmap"

# Applications you want to run with fluxbox.
# MAKE SURE THAT APPS THAT KEEP RUNNING HAVE AN ’’&’’ AT THE END.
#
# unclutter -idle 2 &
# wmnd &
# wmsmixer -w &
# idesk &
#
# Debian-local change:
#   - fbautostart has been added with a quick hack to check to see if it
#     exists. If it does, we’ll start it up by default.
which fbautostart > /dev/null
if [ $? -eq 0 ]; then
    fbautostart
fi

# And last but not least we start fluxbox.
# Because it is the last app you have to run it with ’’exec’’ before it.

 exec virtualbox&
 exec fluxbox
# or if you want to keep a log:
#  exec virtualbox&
 exec fluxbox -log "/home/ourbox/.fluxbox/log"
```

E.4 Fluxbox User Menu

```
# This is an automatically generated file.
# Please see <file:/usr/share/doc/menu/README> for information.

# to use your own menu, copy this to ~/.fluxbox/menu, then edit
# ~/.fluxbox/init and change the session.menuFile path to ~/.fluxbox/menu

[begin] (Fluxbox)

# Automatically generated file. Do not edit (see /usr/share/doc/menu/html/index.html)

   [submenu] (Applications) {}
      [submenu] (VirtualBox) {}
         [exec] (VirtualBox) {/usr/bin/virtualbox} </usr/share/pixmaps/virtualbox.xpm>
      [end]
   [end]
   [restart] (Restart)
   [exit] (Exit)

[end]
```

F Read me file

```
Now that you have installed ubuntu on the machine, now you need to
make the runMe.sh executable.
Also, remember to make a folder which is called "os" in your desktop,
this is where the .vdi images are going to be.

Open a terminal and run:
sudo chomd +x /path/to/runMe.sh

Now that you have done that, you need to execute runMe.sh,
in terminal run:
/path/to/runMe.sh

this will install, virtualbox, fluxbox and libpam-ldapd,
it will configure the default for the different services.
This may take a minute, but the machine will reboot afterwards.
When the machine boots up again you need to login as the user you made
in fluxbox.
Then you can log out and log in again into the unity, and then run
afterInstallation.sh

open terminal and run:
/path/to/afterinstallation.sh

What afterInstallation does, is configure the admin users fluxbox to have
some extra functionality. So that you can lock so that the other users can not
log in into ubuntu but only fluxbox.
```

G Project agreement

HØGSKOLEN I GJØVIK

PROJECT AGREEMENT

between Gjøvik University College (GUC) (education institution),

THOMAS KEMMERICH

____________________ (employer), and

Stepan Maluchev, GAVIN T. GARRAD

____

____________________ (student(s))

The agreement specifies obligations of the contracting parties concerning the completion of the project and the rights to use the results that the project produces:

1. The student(s) shall complete the project in the period from 07.01.05 to 15.05.15.

   The students shall in this period follow a set schedule where GUC gives academic supervision. The employer contributes with project assistance as agreed upon at set times. The employer puts knowledge and materials at disposal necessary to complete the project. It is assumed that given problems in the project are adapted to a suitable level for the students' academic knowledge. It is the employer's duty to evaluate the project for free on enquiry from GUC.

2. The costs of completion of the project are covered as follows:
   - Employer covers completion of the project such as materials, phone/fax, travelling and necessary accommodation on places far from GUC. Students cover the expenses for printing and completion of the written assignment of the project.
   - The right of ownership to potential prototypes falls to those who have paid the components and materials and so on used to make the prototype. If it is necessary with larger or specific investments to complete the project, it has to be made an own agreement between parties about potential cost allocation and right of ownership.

3. GUC is no guarantor that what employer have ordered works after intentions, nor that the project will be completed. The project must be considered as an exam related assignment that will be evaluated by lecturer/supervisor and examiner. Nevertheless it is an obligation for the performer of the project to complete it according to specifications, function level and times as agreed upon.

4. The total assignment with drawings, models and apparatus as well as program listing, source codes and so on included as a part of or as an appendix to the assignment, is handed over as a copy to GUC who free of charge can use it in lessons and in research purpose. The assignment or appendix cannot be used by GUC for other purposes, and will not be handed over to an outsider without an agreement with the rest of the parties in this agreement. This applies as well to companies where employees at GUC and/or students have interests.

   Assignments with grade C or better are registered and placed at the school's library. An electronic project assignment without attachments will be placed on the library part of the school's website. This depends on that the students sign a separate agreement where they give the library rights to make their main project available both on print and on Internet (ck. The Copyright Act). Employer and supervisor accept this kind of disclosure when they sign this project agreement, and they must possibly give a written message to students and dean if they during the project period change view on this kind of disclosure.

5. The assignment's specifications and results can be used by the employer's own work. If the student(s) in its assignment or while working with it, makes a patentable invention, relations between employer and student(s) applies as described in *Act respecting the right to employees' inventions* of 17th of April 1970, §§ 4-10.

6. Beyond the publicising mentioned in item 4, the student(s) have no right to publicise his/hers/theirs assignment, fully or partly or as a part of another work, without consensus from the employer. Equivalent consent must be made between student(s) and lecturer/supervisor regarding the material placed at disposal by the lecturer/supervisor.

7. The students shall hand in the assignment with attachments electronic (PDF) in Fronter. In addition the students shall hand in a copy to the employer.

8. This agreement is drawn up with one copy to each party. On behalf of GUC it is dean/vice dean who approves the agreement.

9. In each case it is possible to enter separate agreement between employer, student(s) and GUC who closer regulate conditions regarding issues such as ownership, further use, confidentiality, cost coverage, and economic utilisation of the results.

   If employer and student(s) wish an additional or new agreement, this will occur without GUC as a party.

10. When GUC also act as employer, GUC accede to the agreement both as education institution and as employer.

11. Possible disagreements concerning understanding of this agreement are solved by negotiations between the parties. If consensus is not achieved, the parties agree that the disagreement is solved by arbitration, according to provision in Civil Procedure Act of 13th of August 1915, no 6, chapter 32.

12. Participants by project implementation:

GUCs supervisor (name): ERIK HJELMÅS

Employers contact person (name): THOMAS KEMMERICH

Student(s) (signature): Gavin T. Garrod date 28.01.15

(Stepan Maluchev) date 28.01.15

_______________ date _______

_______________ date _______

Employer (signature): _______________ date 28.01.15

IMT Dean/Vice Dean (signature): _______________ date _______

Revised 25th of November, 2010, Hilde Bakke