

**TEKNISKA HÖGSKOLAN**

HÖGSKOLAN I JÖNKÖPING

**Virtualisering för en effektivare labora-  
tionsmiljö**

**Virtualization, for a more efficient laboratory envi-  
ronment**

Martin Andersson

Niklas Strand

**EXAMENSARBETE 2011**

Teknikens Tillämpning med inriktning mot Datanät-  
teknik

---

Postadress:

Box 1026

551 11 Jönköping

Besöksadress:

Gjuterigatan 5

Telefon:

036-10 10 00 (vx)

Detta examensarbete är utfört vid Tekniska Högskolan i Jönköping inom huvudområdet Teknikens Tillämpning med inriktning mot Datanätteknik. Arbetet är ett led i det 1-åriga kandidatpåbyggnadsprogrammet. Författarna svarar själva för framförda åsikter, slutsatser och resultat.

Examinator: Kristina Säfsten

Handledare: Erik Gunnarsson

Omfattning: 15 hp (grundnivå)

Datum: 2012-03-26

---

Postadress:

Besöksadress:

Telefon:

Box 1026

Gjuterigatan 5

036-10 10 00 (vx)

551 11 Jönköping

## **Abstract**

IT-companies today need to experiment with the technology they deliver to their customers. A laboratory environment with little or no planning behind it can have a quite low efficiency. The term “efficiency” is used, in this work, as a measurement of how much time it takes to complete a certain work-task. However, efficiency also means reducing the usage of resources. Increased efficiency means that less time is used to complete one or more tasks. Today, IT-companies normally utilize a fraction of their total server capacity. To utilize the servers more efficiently, and be more economic, virtualization is a good solution. TDC, who delivers communication solutions, has a laboratory environment at their office in Växjö. The work includes, among other things, time-consuming reinstallation of servers. Another problem is that people currently often need to be in the environment, physically. This limits the use of the environment because people in other offices have no chance of using it. The purpose of this work is to develop guidelines as to build efficient laboratory environments for servers and network-units, and to reduce the need of physical access to the laboratory equipment.

During the work a workshop was conducted, regular telephone reconciliations were held, experiments were done and a literature research was made. The experiments dealt with virtualization and external storage. The goal has been to develop guidelines on how to build efficient laboratory environments. Through the literature research, in multiple databases, sources were found that support the theory and the results.

The guidelines developed show how to reduce the need of physical presence by allowing remote access and having a well-planned network structure. By configuring the remote access properly the equipment can be accessed even if some parts of the network structure are not fully functional. Servers can be accessed via a special expansion card as long as they have power and network access. This way, you can start and stop the server and even reinstall its operatingsystem, all of it remotely. It is essential in a laboratory environment to be able to restore the configuration of a device to a initial state, or known working state. For networking units this is accomplished by storing (and getting) configurations from a central server. Virtualization is a good option when it comes to restoring servers, because it is easy to save snapshots of virtual machines. The whole environment and its routines should be well documented.

### **Keywords**

Efficiency

Laboratory environment

Virtualization

Guidelines

Remote Access

## Sammanfattning

Som IT-företag har man ett behov av att laborera med den teknik man levererar till sina kunder. För att öka effektiviteten i dessa laborationsmiljöer bör man sträva efter att minska genomförandetiden för de olika arbetsmoment som ingår i laborationerna. TDC, som levererar kommunikationslösningar, har en laborationsmiljö på sitt kontor i Växjö. Arbetet innefattar tidsödande ominstallationer av servrar. Ett annat problem är att man i dagsläget ofta fysiskt måste befinna sig i laborationsmiljön vilket begränsar arbetet då personer som inte befinner sig på samma kontor inte kan använda utrustningen. Under laborationer finns ofta ett behov av att rensa servrar och nätverksenheter för att sedan utföra ominstallation eller omkonfiguration, arbetsmoment som kan ha hög tidsåtgång. Syftet med arbetet är att ta fram riktlinjer för hur man bygger upp effektiva laborationsmiljöer för nätverksenheter och servrar med hjälp av bland annat virtualisering, samt att reducera behovet av fysisk tillgång till laborationsutrustning.

Under arbetets gång utfördes en workshop, regelbundna telefonavstämningar, laborationer och en litteratursökning. Målet har varit att ta fram riktlinjer för hur man bygger effektiva laborationsmiljöer. Laborationerna innefattade virtualisering och extern lagring. Genom att utföra litteratursökningen i ett flertal databaser har man hittat källor för att stödja teori och resultat.

Riktlinjerna som tagits fram visar hur man minskar behovet av fysisk närvaro genom att möjliggöra fjärråtkomst samt genom att ha en väl planerad nätverksstruktur. Genom att använda fjärranslutningar på rätt sätt kommer man åt utrustningen även om nätverksutrustningen i laborationsmiljön inte är fullt funktionell. Om man har ett instickskort för remote management i servern går det ofta att komma åt servrar via ett gränssnitt så länge den har ström och nätverksåtkomst. På det sättet kan man via fjärranslutning stänga av och starta servern samt installera operativsystem på den. För att minska tidsåtgången vid rensning och ominstallation/omkonfiguration av servrar och nätverksenheter kan man använda sig av virtualisering med ögonblicksbilder och nätverksenheters konfiguration kan sparas på servrar. Företag utnyttjar generellt sett bara en liten del av sin serverkapacitet. För att använda kapaciteten effektivare och på så sätt vara mer ekonomisk är virtualisering en bra lösning [1]. Rent administrativt bör laborationsmiljön och dess rutiner vara mycket väl dokumenterade.

### Nyckelord

Effektivisering

Laborationsmiljö

Virtualisering

Riktlinjer

Fjärråtkomst

# Innehållsförteckning

<b>1</b>	<b>Inledning.....</b>	<b>4</b>
1.1	BAKGRUND OCH PROBLEMBESKRIVNING .....	4
1.2	SYFTE OCH FRÅGESTÄLLNINGAR .....	4
1.3	AVGRÄNSNINGAR .....	5
1.4	DISPOSITION .....	5
<b>2</b>	<b>Teoretisk bakgrund .....</b>	<b>6</b>
2.1	ITIL .....	6
2.2	VIRTUALISERING.....	6
2.2.1	Lagring i ESXi .....	8
2.3	VIRTUALIZATION TECHNOLOGY (INTEL VT) .....	8
2.4	NAS/SAN.....	9
2.5	NFS .....	10
2.6	iSCSI .....	10
2.7	HÅRDISKKLONING .....	11
2.8	MESH-TOPOLOGI.....	11
2.9	TFTP .....	12
2.10	CONSOLE-SERVER.....	12
2.11	RAC .....	12
<b>3</b>	<b>Metod och genomförande.....</b>	<b>13</b>
3.1	TDC.....	13
3.2	WORKSHOP .....	13
3.3	LITTERATURSÖKNING .....	14
3.4	LABORATION .....	14
3.4.1	ESXi.....	15
3.4.2	Extern lagring.....	15
<b>4</b>	<b>Resultat och analys.....</b>	<b>17</b>
4.1	PRINCIPER FÖR UPPBYGGNAD AV EN EFFEKTIV LABORATIONSMILJÖ.....	17
4.1.1	Fjärråtkomst av nätverksutrustning och servrar .....	17
4.1.2	Återställning till ursprungsläge .....	17
4.1.3	Rutiner .....	18
4.1.4	Säkerhet .....	18
4.1.5	Framtidssäkerhet .....	19
4.1.6	Dokumentation.....	19
4.2	EXTERN LAGRING .....	19
4.3	FÖRSLAG FÖR TDCS LABORATIONSMILJÖ.....	21
4.3.1	Nätverksstruktur .....	21
4.3.2	Servermiljö.....	22
<b>5</b>	<b>Diskussion och slutsatser .....</b>	<b>23</b>
5.1	RESULTATDISKUSSION .....	23
5.2	METODDISKUSSION .....	24
5.3	SLUTSATSER OCH REKOMMENDATIONER .....	24
<b>6</b>	<b>Referenser .....</b>	<b>26</b>
<b>7</b>	<b>Sökord.....</b>	<b>28</b>

# 1 Inledning

De IT-företag vi har varit i kontakt med genom utbildning, arbete etc. har alla någon form av laborationsmiljö för att testa mjukvaror och konfigurationer innan dessa sätts i skarp drift. En liten ändring i ett system kan få enorma konsekvenser. Det är dock inte säkert att en laborationsmiljö planeras och byggs från grunden. Möjligen har någon lånat hårdvara som har samma egenskaper som den skarpa miljön för att laborera med, och sedan byggs det på. En miljö som uppkommit på det sättet och alltså inte är planerad och genomtänkt kan ha en ganska låg effektivitet.

ITILv3 (en standard som beskriver tjänstehantering på IT-företag) definierar effektivitet som: ”Ett mått på om rätt mängd resurser har använts för att leverera en process, tjänst eller verksamhet. En effektiv process uppnår sina mål med den minsta mängd tid, pengar, människor eller andra resurser.” [2]

## 1.1 Bakgrund och problembeskrivning

Ett genomsnittligt företag utnyttjar 5 - 25% av sin serverkapacitet och det finns således en stor mängd outnyttjad serverkapacitet. [1] Det är oekonomiskt att köpa in flertalet servrar och sedan bara använda en liten del av kapaciteten. Med hjälp av virtualisering (se kapitel 2.2) kan man använda en mycket större del av varje servers kapacitet. Att använda virtualisering i produktionsmiljön börjar därför bli ganska vanligt förekommande. [3] I en laborationsmiljö prioriterar man andra saker men även här kan virtualisering ge många fördelar, det går till exempel mycket snabbare att skapa nya maskiner. [4]

TDC i Växjö har idag en laborationsmiljö där de löpande behöver ta fram nya installationer av operativsystem. I dagsläget installeras och konfigureras operativsystem på fysiska servrar. Laborationsmiljön är enligt TDC själva ineffektiv då det tar lång tid när man gång på gång måste installera om servrar för att få rätt version på mjukvaran. Tiden det tar att installera om servrar är tid som inte kan debiteras till kund, vilket gör det till en förlust för företaget. Man vill effektivisera detta så att det snabbt går att ta fram installationer med rätt mjukvara i, och på så sätt minska sin icke-debiterbara tid. När laborationerna innefattar nätverksutrustning behöver man ofta koppla om kablar mellan switchar, routrar och annan nätverksutrustning vilket innebär att man fysiskt måste befinna sig i laborationsmiljön. En mer optimal lösning hade varit om man kunde sköta allt från sin dator på kontoret, speciellt för de personer som arbetar på andra kontor i landet som då får möjlighet att använda laborationsmiljön.

## 1.2 Syfte och frågeställningar

Syftet med arbetet är att ta fram riktlinjer för hur man bygger upp effektiva laborationsmiljöer för nätverksenheter och servrar med hjälp av bland annat virtualisering, samt att reducera behovet av fysisk tillgång till laborationsutrustning.

Syftet kan brytas ner till följande frågeställningar:

- Hur kan virtualisering effektivisera en laborationsmiljö för servrar och nätverksutrustning?

- Hur implementerar man detta praktiskt i en laborationsmiljö?
- Hur kan behovet av fysisk tillgång till laborationsmiljön reduceras?

### **1.3 Avgränsningar**

TDC använder sig idag av virtualiseringsmjukvaror från VMware. Eftersom de vill ha en homogen miljö vill de att VMwares mjukvaror ska användas i lösningen, därför kommer inga andra mjukvaror för virtualisering att utredas.

### **1.4 Disposition**

Inledningen beskriver vilka TDC är. Den beskriver även den problematik som de har haft och som ligger till grund för examensarbetet.

I teoridelen förklaras de tekniska begrepp som läsaren behöver kunskap om för att förstå rapporten.

I metod och genomförande beskrivs sedan arbetsprocessen uppdelat i tre delar. Stycket ”TDC” beskriver hur kontakten med TDC har fungerat och vad deras roll i arbetet har varit. Stycket ”Framtagande av riktlinjer” förklarar hur riktlinjerna togs fram och under stycket ”Laboration” beskrivs de praktiska laborationer som genomförts.

Resultatdelen är uppdelad i två delar. Den första delen innehåller riktlinjerna för hur man bygger effektiva laborationsmiljöer vilket inkluderar virtualisering och att reducera behovet av fysisk tillgång. Den andra delen beskriver hur dessa skulle kunna appliceras gentemot TDC.

I diskussion och slutsats sammanfattas arbetet. Reflektioner om frågeställningarnas besvarande samt andra avslutande punkter diskuteras.

## 2 Teoretisk bakgrund

Kapitlet behandlar den teori som arbetet stödjer sig i.

### 2.1 ITIL

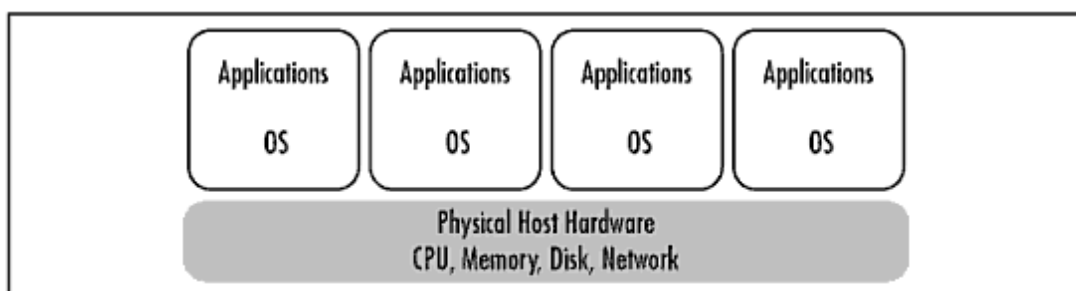
Information Technology Infrastructure Library (ITIL) är ett ramverk för IT Service Management (ITSM). ITSM är studiet av hur man skapar, administrerar, vidareutvecklar och ger support på IT-tjänster. På 80-talet växte användandet av ITSM och den brittiska regeringen bestämde sig för att ta reda på hur de mest framgångsrika företagen använde sig utav ITSM. Det sammanfattade resultatet blev ITIL som innehöll principer och processer för att arbeta med IT. ITIL är inte anpassat till någon specifik typ av företag eller någon speciell teknisk plattform. Det kan alltså användas av i princip alla företag i IT-branchen. [5]

### 2.2 Virtualisering

En kort definition av virtualisering kan lyda:

*A framework or methodology of dividing the resources of a computer hardware into multiple execution environments, by applying one or more concepts or technologies such as hardware and software partitioning, time-sharing, partial or complete machine simulation, emulation, quality or service, and many others. [6]*

En virtuell maskin består i praktiken av ett eller flera operativsystem (gäst) som körs helt isolerat i ett annat operativsystem (värd). På en server körs en hypervisor som hanterar de virtuella maskinerna. Hypervisorn fungerar som ett virtualiseringslager mellan hårdvara och virtuella maskiner. Eftersom de virtuella maskinerna jobbar mot samma hårdvara måste hypervisorn kontrollera hur de får åtkomst till hårdvarurresurser. [7] Varje virtuell maskin kör sitt eget operativsystem och sina egna applikationer.



Figur 1: Virtuella maskiner delar på hårdvarurresurser [6]



VMware är en stor aktör på marknaden för virtualiseringsplattformar. Deras plattform heter vSphere och använder sig av hypervisorn ESXi. Det går att få en gratislicens till ESXi men då ingår inte mer grundläggande funktionalitet för att virtualisera. För att kunna använda flera servrar i ett kluster och för att få de mer avancerade funktionerna vSphere har att erbjuda behöver man en betallicens. Mer avancerad funktionalitet innefattar tekniker för hög tillgänglighet, säkerhet, skalbarhet, lagring och nätverksstruktur. [8] [9] VMware har också en produkt som heter VMware Workstation, denna är tänkt att användas för att virtualisera på en vanlig arbetsdator. Det finns inte samma stöd för kloning, redundans och lastbalansering som i vSphere.

	Essentials Kit	Essentials Plus Kit	Standard Acceleration Kit	Advanced Acceleration Kit	Midsize Acceleration Kit (w/Enterprise)	Enterprise Plus Acceleration Kit
Centralized Management	vCenter for Essentials	vCenter for Essentials Plus	vCenter Standard	vCenter Standard	vCenter Standard	vCenter Standard
Included Entitlement	3 servers with up to 2 processors each	3 servers with up to 2 processors each	8 Processors - scalable with additional licenses	6 Processors - scalable with additional licenses	6 Processors - scalable with additional licenses	8 Processors - scalable with additional licenses
Memory/Physical Server	256GB	256GB	256GB	256GB	256GB	No Memory Limit
Cores per Processor	6	6	6	12	6	12
vCPU Entitlement	4-way	4-way	4-way	4-way	4-way	8-way
Thin Provisioning	✓	✓	✓	✓	✓	✓
Update Manager	✓	✓	✓	✓	✓	✓
vStorage APIs for Data Protection	✓	✓	✓	✓	✓	✓
Data Recovery		✓	✓	✓	✓	✓
High Availability		✓	✓	✓	✓	✓
vMotion		✓	✓	✓	✓	✓
Virtual Serial Port Concentrator				✓	✓	✓
Hot Add				✓	✓	✓
vShield Zones				✓	✓	✓
Fault Tolerance				✓	✓	✓
vStorage APIs for Array Integration					✓	✓
vStorage APIs for Multipathing					✓	✓
Storage vMotion					✓	✓
Distributed Resources Scheduler (DRS), Distributed Power Management (DPM)					✓	✓
Storage I/O Control						✓
Network I/O Control						✓
Distributed Switch						✓
Host Profiles						✓

Tabell 1: VMware licensmatris [10]

### 2.2.1 Lagring i ESXi

En ESXi server hanterar olika så kallade *Datastores*, vilket är olika lagringsenheter. Som standard har en nyinstallerad server endast en datastore, vilket är den interna hårddisken (eller RAID-enheten) i servern (hädanefter kallad *intern datastore*). En intern datastore kan bara den komma åt av den specifika servern. Har man bara en server så fungerar detta utmärkt, det är till och med att föredra i en laborationsmiljö eftersom servern då inte blir beroende av någon extern enhet. Har man flera ESXi-serverar i ett *kluster* (som administreras från en vCenter server) så blir situationen annorlunda. I vCenter kan man flytta virtuella maskiner mellan olika ESXi-serverar, antingen manuellt efter eget tycke eller via DRS (DRS står för Distributed Resource Scheduler och är endast tillgängligt i vissa licensformer) om man har ett kluster. Om serverarna har interna datastores och därmed inte kommer åt varandras datastores så kommer de virtuella maskinerna att behöva flyttas över nätverket. Detta fungerar förvisso, men det tar ganska lång tid. Hur lång tid beror på hur snabb nätverksanslutning man har samt hur stora de virtuella maskinerna är.

Har man flera serverar är det istället rekommenderat att använda någon form av extern lagring (hädanefter kallat *extern datastore*). Skillnaden är att man då har en datastore på en extern enhet (exempelvis på någon form av NAS, Network Attached Storage, eller SAN, Storage Area Network), samt att flera maskiner kan ha tillgång till samma datastore. Om maskiner ligger på en extern datastore som flera serverar har tillgång till så går det mycket fortare att flytta maskiner mellan serverarna. Eftersom serverarna har tillgång till samma datastore så behöver inga filer flyttas, den virtuella maskinen avregistreras helt enkelt från den ena servern och registreras sedan på den andra servern. Detta går till och med att göra när den virtuella maskinen är igång om man har vMotion (vMotion är endast tillgängligt i vissa licensformer).

## 2.3 Virtualization Technology (Intel VT)

Processortillverkaren Intel har en teknik för hårdvaruassisterad virtualisering som kallas Intel VT. Tekniken innebär att processor och chipset i en server eller dator har inbyggda funktioner för virtualisering. Virtualiseringsmjukvaran måste emulera en hårdvarumiljö för virtuella maskiner, vilket är prestandakrävande. Intel VT avlastar systemet genom att i högre utsträckning sköta låta hårdvaran sköta resurstilldelningen. [11]

Intel VT innefattar teknikerna VT-x, VT-d och VT-c. [12]

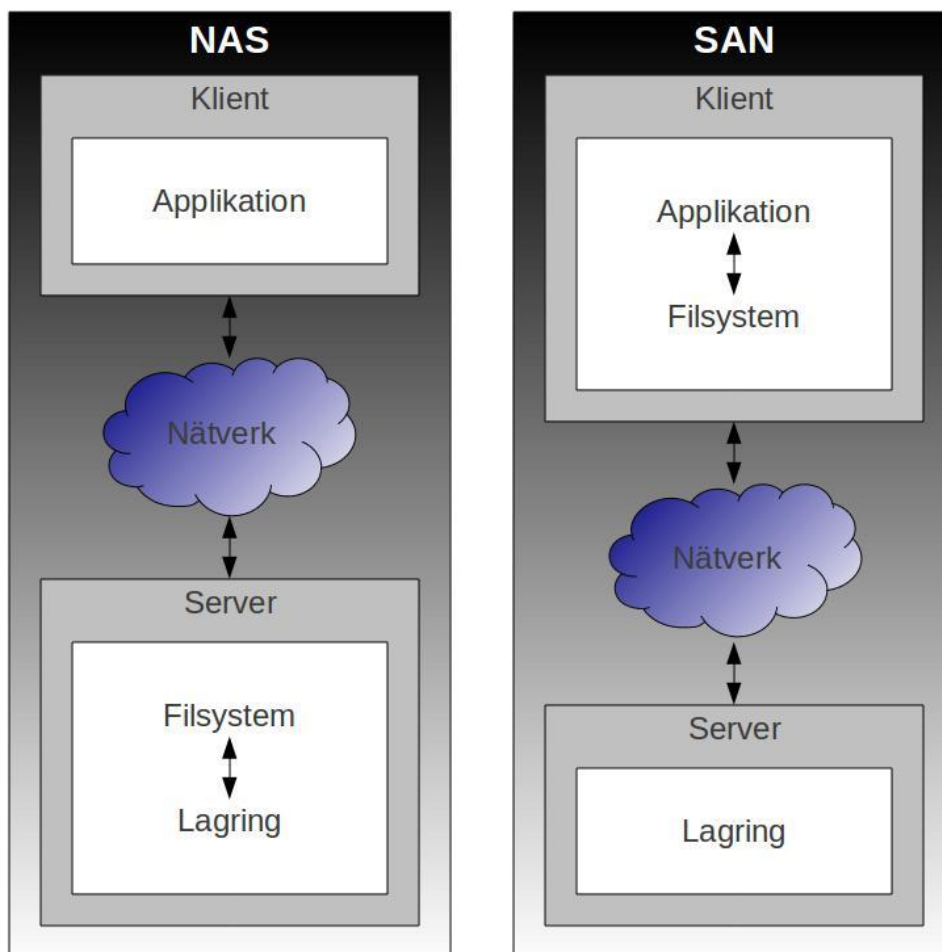
- VT-x är en funktion i processorn, utan VT-x måste hypervisorn hantera instruktioner mellan virtuella maskiner och processorn. VT-x hanterar en del av de instruktionerna så att de virtuella maskinerna kan fungera mer effektivt. VT-x är nödvändigt för att virtualisera 64-bitars maskiner.
- VT-d är en funktion i processorn/chipset, och låter virtuella maskiner få direkt tillgång till hårdvarurersurser. Lasten på processorn minskar när hypervisorn inte behöver hantera I/O flöden mot hårdvaran.

- VT-c är en funktion i nätverkskort och sorterar inkommande paket på nätverkskortet vilket gör att det blir en mindre arbetsbörda för hypervisorn att se till så att rätt paket kommer till rätt virtuell maskin.

Processortillverkaren AMD har en liknande teknik som heter AMD-V, som bland annat möjliggör att köra 64-bitars maskiner.

## 2.4 NAS/SAN

Network Attached Storage (NAS) och Storage Area Network (SAN) är två olika koncept för att hantera extern lagring [13]. De har samma syfte, att lagra saker över nätverk, men de fungerar olika. Se Figur 2. En NAS har ett lagringsutrymme formaterat med valfritt filsystem, som sedan delas ut med ett protokoll, till exempel NFS. När en klient skall lagra en fil så skickas filen via NFS till servern som sedan ansvarar för att skriva block-data till lagringen. Ett SAN har ett lagringsutrymme som inte är formaterat, som sedan delas ut via till exempel iSCSI. Klienten formaterar sedan utdelningen med valfritt filsystem. När klienten vill lagra en fil så omvandlas den till block-data för filsystemet. Det är sedan block-data som skickas över nätverket. Även om servern fysiskt skriver blocken till disk så har den ingen uppfattning om filsystemet.



Figur 2: Skillnaden mellan NAS och SAN

SAN/NAS köps ofta som en färdig produkt, med både hårdvara och mjukvara. Det går även att använda egen hårdvara tillsammans med ett operativsystem med stöd för tekniken man vill använda. Både FreeNAS och Openfiler är så kallade ”Appliances”, programvaror som är specialiserade på en viss uppgift som i det här fallet är att agera NAS/SAN. (FreeNAS bygger på FreeBSD och Openfiler bygger på rPath Linux). Båda två innehåller funktioner för att dela ut lagringsutrymme via Network File System (NFS), och Internet Small Computer System Interface (iSCSI), vilket kan användas av ESXi.

## 2.5 NFS

NFS är ett protokoll för att tillåta delad åtkomst till en resurs över nätverk. Det är ett förhållandevis gammalt protokoll, som utvecklades av Sun Microsystems runt 1984. Protokollet är vanligt på Unix och Unix-dialekter (så som Linux och BSD). Senaste versionen är 4.1 som specificerades i januari 2010. [14] På klienten monterar man utdelningen i en tom katalog, innehållet finns sedan i katalogen som om det vore lokalt.

Ett vanligt användningsområde för NFS är att montera home-katalogen i POSIX-system över nätverket (POSIX är en standard, som beskriver hur operativsystem ska fungera som de flesta Linux och UNIX-liknande system följer). Klienter skickar bara NFS-kommandon, och bryr sig inte om vad det är för filsystem på servern. Det är upp till servern att hantera och lagra filerna på disken.

## 2.6 iSCSI

iSCSI [15] är ett protokoll för att hantera lagring över nätverk. SCSI är ett protokoll för lokala diskar, iSCSI skickar samma kommandon men över IP-protokollet. Detta gör att kommandon kan skickas på stora avstånd i befintligt IP-kapabelt nätverk. Protokollet tillåter klienter (som kallas *Initiators*) att ansluta till SCSI-lagringsenheter på servrar (så kallade *Targets*). Det finns olika typer av initiators, största skillnaden är om den är helt mjukvarubaserad (Software initiator), eller om den har någon form av dedikerad hårdvara för att avlasta resten av systemet (Hardware initiator). Oavsett om man har en Soft- eller Hardware initiator så kommer den att registrera sig som en SCSI-buss på klienten, när man sedan ansluter till server så kommer utdelningen att se ut som en lokal disk kopplad till den tidigare nämnda SCSI-bussen.

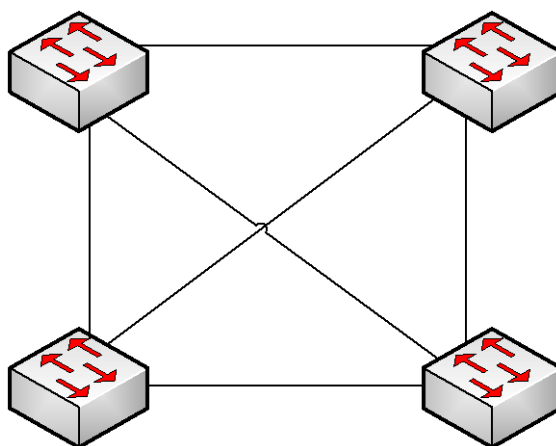
Det är klienten som hanterar filsystem och liknande, iSCSI hanterar bara block-data, som en lokal SCSI-buss. En stor skillnad mot en lokal disk är att flera klienter kan vara anslutna samtidigt till samma utdelning. Detta ställer vissa krav på filsystemet, till exempel att filer som används kan låsas så att inte flera klienter försöker ändra en fil samtidigt.

## 2.7 Hårdiskkloning

Man kan spara en färdig installation på en server genom att göra en klon på hårddisken. Det innebär att en avbild av hårddisken kopieras till en fil som sedan kan kopieras tillbaka genom att skriva tillbaka avbilden. Detta kan göras lokalt på servern eller över nätverket. [16] Återställningen görs av ett program som kan skriva tillbaka avbilden till hårddisken.

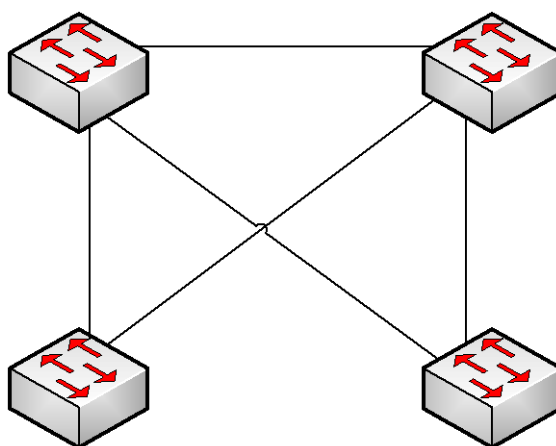
## 2.8 Mesh-topologi

Nätverkstopologier beskriver vilken logik den fysiska sammankopplingen av enheter följer. En mesh-topologi [17] bygger på nätverksenheter i så stor grad som möjligt är direktkopplade till varandra. I en full mesh-topologi är alla enheter sammankopplade direkt till varandra. Se Figur 3.



Figur 3: Full mesh-topologi.

När man har många enheter blir antalet kopplingar många och det kan då vara lämpligt att istället använda sig av en så kallad partiell mesh-topologi. Detta innebär att bara vissa enheter är sammankopplade. Se Figur 4.



Figur 4: Partiell mesh-topologi.

## 2.9 TFTP

Trivial File Transfer Protocol (TFTP) är ett enkelt filöverföringsprotokoll. Det har mindre funktioner än FTP och kräver mindre av hårdvaran, speciellt mindre RAM. Det används därför ofta på enheter med begränsad kapacitet, som till exempel switchar och routrar. TFTP har ingen felkontroll och ingen autentisering och används därför nästan uteslutande i interna nätverk. TFTP används vanligen för att överföra uppstartsfiler (bootimages) samt konfigurationsfiler. De flesta mer avancerade switchar och routrar kan hämta både bootimage och konfiguration från en TFTP-server vid uppstart, samt spara och hämta konfigurationsfiler under drift.

## 2.10 Console-server

D. Sloan föreslår i hans arbete om fjärråtkomst till en laborationsmiljö att servrar ska kopplas samman via serielänkar. [18] Om det då inte går att nätverksansluta direkt till en server så går det att komma åt den via en annan server över serielanslutningen. En liknande lösning går att använda för att försäkra sig om att det går att komma åt nätverksenheter. Det finns speciella enheter som är gjorda för att användas som console-servrar. Dessa har seriella interface som kopplas till console-porten på nätverksenheter. Console-portar på dessa enheter används för att ge grundläggande konfiguration när enheten inte går att nå på annat sätt. Console-servern används för att kunna komma åt flera enheter via nätverket (när de i annat fall enbart går att nå via console). Fördelen med att använda en console-server är att om den är kopplad till någon fungerande del av nätverket kan man alltid komma åt de enheter den har seriella anslutningar till.

## 2.11 RAC

Ett så kallat Remote Access Card (RAC) [19] kan användas för att administrera servern över nätverket. DRAC (för DELL servrar) och ILO (för HP servrar) är två vanliga typer av RAC. Funktioner som oftast finns i ett RAC är:

- Fjärrstyrning – Oberoende av installerat operativsystem, fungerar även om det inte finns något operativsystem.
- Strömhantering – Det går att stänga av, sätta på och starta om servern.
- Montering av imagefiler – En imagefil för en CD eller DVD går att montera så att man kan installera operativsystem på servern.

### 3 Metod och genomförande

Avsnittet beskriver hur arbetet har genomförts och vilka delmoment som har ingått, hur riktlinjerna har arbetats fram, vad TDCs roll i arbetet har varit och vilka laborationer som genomförts.

#### 3.1 TDC

På TDCs initiativ bestämdes det att en workshop skulle hållas på kontoret i Växjö. Syftet med workshopen var att få fram vilka moment som ansågs mest vara mest tidsödande samt vilka idéer teknikerna hade för att förbättra laborationsmiljön.

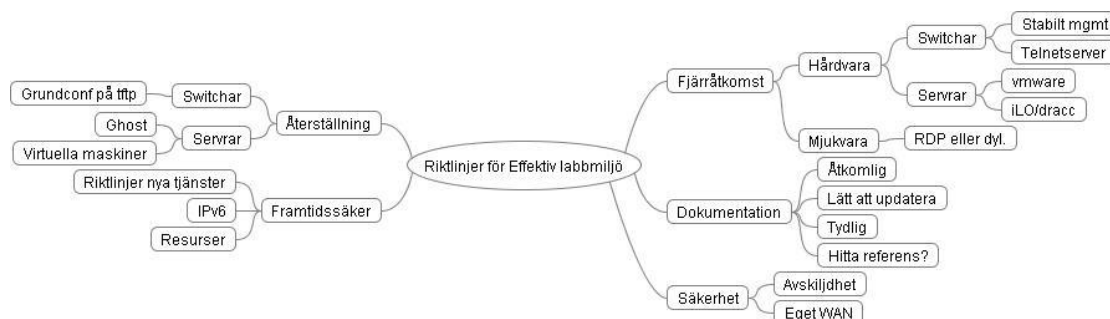
För att ge proof of concept på riktlinjerna exemplifierades de gentemot TDC. Utifrån de riktlinjer som tagits fram, arbetades det fram lösningsförslag för TDCs laborationsmiljö. Till viss del var förslaget teoretiskt men det genomfördes även laborationer. Tanken var inte att ta fram en heltäckande lösning utan snarare att den skulle arbetas fram i mån av tid.

#### 3.2 Workshop

Workshopen hölls med merparten av de tekniker som använder laborationsmiljön närvarande. Teknikerna fick sedan i enrum förklara vad just de använde laborationsmiljön till, vad som krävdes för att de skulle kunna använda den, och vilka förbättringar som skulle gagna dem. Avstämningar har gjorts löpande via telefon. Avstämningarna har skett veckovis med vissa undantag. Under mötena har diskussioner förts om vad som gjorts hittills och vilka önskemål samt prioriteringar som funnits från TDCs håll.

Efter workshopen bestämdes det att man skulle svara på frågeställningarna "Hur kan virtualisering effektivisera en laborationsmiljö för servrar och switchar?" och "Hur kan behovet av fysisk tillgång till laborationsmiljön reduceras?" genom att formulera riktlinjer för hur man bygger upp effektiva laborationsmiljöer.

Inledningsvis ritades en mindmap för att få en struktur på de idéer som kom fram efter workshopen, se Figur 5. Utifrån denna mindmap började skrivarbetet. Efterhand tillkom riktlinjer, baserat på egna idéer, vad som kom fram under avstämningarna med TDC samt genom litteratursökning.



Figur 5: Mindmap för riktlinjer

En specifik ide som uppkom under workshopen var att koppla nätverksutrustning i ett fullt mesh-nät. Valfri topologi byggs sedan upp genom att aktivera vissa länkar. Fördelen med att använda ett fullt mesh-nät är att ingen omkoppling av kablage behövs. Allt sköts i konfigurationen av hårdvaran.

### **3.3 Litteratursökning**

För att täcka ett stort utbud litteratur, forskning och tidskrifter användes flera olika databaser i en litteratursökning som utfördes löpande under arbetets gång. De böcker, artiklar och rapporter som var intressanta för arbetet sparades digitalt i en delad katalog för att de skulle vara lättillgängliga. Fulltext-databaserna ACM Digital Library, ScienceDirect, SpringerLink, Google Books och Google Scholar nyttjades. Vi använde sig även av ett tidigare examensarbete inom samma ämnesinriktning [20].

Generella sökord så som virtualization, efficiency, remote-access och laboratory-environment användes i de olika databaserna för att hitta källor till skrivandet av riktlinjerna. Under framtagandet av riktlinjer sågs en brist i att det som skrevs hade en svag teoretisk förankring. Då fick man gå tillbaka och komplettera med ytterligare källor. Genom att bygga slutsatser på teori som i sin tur är förankrad i källor gavs bäring åt resultatet. Lämpliga källor letades fram för att kunna skriva teoridelen. De slutsatser som dras senare i rapporten grundar sig i teorin. Problembeskrivningen gick att stödja med ett flertal källor som framkom under litteratursökningen. Litteraturen hjälpte även till att styrka riktlinjer då det gick att hitta exempel på hur problem hade lösts i liknande situationer. I vissa källor kunde man se likheter mellan problem TDC hade och problem som källorna tog upp.

### **3.4 Laboration**

Laborationerna ägde rum på JTH med skolans utrustning och en del utrustning som lånades från TDC. De genomfördes för att exemplifiera riktlinjerna och med hänsyn till TDCs önskemål.



### 3.4.1 ESXi

Den senaste versionen av ESXi i skrivande stund är 4.1 och går enbart att köra på 64-bitarsplattformar. Installationen är simpel och efter att ha gjort nätverksinställningar fjärradministreras servern med vSphere client. ESXi kan alltid köra virtuella maskiner med 32-bitars operativsystem, men för att kunna köra 64-bitars-system måste processorn ha stöd för VT-x. En laboration utfördes där ESXi installerades på två identiska servrar. Oftast vill man inte köra två separata instanser utan man vill kunna administrera flera virtualiseringsservrar centralt. På en till server installerades Windows Server 2008 R2 64-bit och vCenter 4.1. I vCenter lades båda ESXi servrarna till och efter det kunde man kontrollera virtualiseringsmiljön genom att ansluta med vSphere client till vCenter-servern. Även om de servrar som användes hade en 64-bitars processorer och körde ESXi 4.1 (vilket enbart finns i 64-bitars version) så gick det inte att köra virtuella maskiner med 64-bitars operativsystem. Det problemet uppstod för att 64-bitars gäster i ESXi kräver att processorn har stöd för Intel VT-x.

TDC hade en server där de testat virtualisering. På den kördes Windows Server 2008 och VMware Workstation. Man ville slippa köra Windows Server i botten och istället testa ESXi. De virtuella maskinerna säkerhetskopierades och sedan installerades ESXi på servern. De maskiner som tidigare körts i VMware Workstation behövde konverteras och kopieras över till de interna datastores som fanns på servern. Detta gick att göra med programmet VMware vCenter Converter. Konverteringen och överföringen tog lång tid då de virtuella maskinerna var mycket stora. Servern hade stöd för Intel VT-x och klarade av att köra 64-bitars operativsystem på de virtuella maskinerna.

### 3.4.2 Extern lagring

För att testa skillnaden mellan iSCSI och NFS för extern lagring så installerades NAS/SAN-programvaran FreeNAS på två servrar. Programvaran var inte tillfredsställande och byttes senare ut till Openfiler innan testerna gjordes. För jämförelsen mellan NFS och iSCSI installerades Openfiler på två identiska maskiner med följande specifikationer: En Intel Xeon E5405 2GHz Quad-core CPU, 12 GB RAM, En 146GB 10k SCSI-disk som systemdisk samt tre likadana diskar i RAID-0 som lagringsutrymme.

Servrarna anslöts till samma labbnät som resten av utrustningen. Den ena lagrings-servern formaterades med filsystemet eXtended File System (XFS). Servern konfigurerades sedan att dela ut sitt lagringsutrymme via NFS. På den andra servern skapades en iSCSI utdelning. Enheten som ansluter, i det här fallet ESXi-servern, ser anslutningen logiskt som en lokal disk och formaterar den själv. ESXi har bara ett alternativ för att formatera diskar, nämligen VMwares egna filsystem VMFS, som är specifikt utvecklat för att hantera virtuella maskiner.

För att testa prestandan med så lika förutsättningar som möjligt så skapades en virtuell maskin, och på denna installerades Windows Server 2003 och testprogrammet HD Tune Pro 4.60 [21]. Den virtuella maskinen klonades sedan och de två maskinerna placerades i varsin extern datastore, anslutna via NFS respektive iSCSI. Sedan kördes olika tester, bland annat läs- och skrivtester med varierande överföringsstorlekar, maximal IOPS (Input/Output Operations Per Second) samt genomsnittlig accesstid vid olika filstorlekar. För att förhindra att resultatet blev skevt på grund av tillfälliga dippar i prestanda eller andra okända faktorer så kördes alla tester tre gånger. Alla värden som används i resultatet är medelvärden från dessa.

## 4 Resultat och analys

Resultatet innefattar riktlinjer och en fokusering på TDC. Riktlinjerna förklarar ett generellt upplägg för hur en laborationsmiljö kan byggas upp. Fokuseringen visar hur dessa skulle kunna appliceras på TDC.

### 4.1 Principer för uppbyggnad av en effektiv laborationsmiljö

#### 4.1.1 Fjärråtkomst av nätverksutrustning och servrar

Behovet av att behöva vara fysiskt närvarande i laborationsmiljön för att kunna utföra sitt arbete är inte helt optimalt. Det hindrar personal på annan ort att använda laborationsmiljön. En effektiv lösning är då att ordna fjärråtkomst till laborationsmiljön.

För att helt eliminera behovet av fysisk närvaro måste man se till att kablage inte behöver kopplas om då detta kräver fysisk närvaro. Istället för att dra ut en kabel kan man via inställningar i switchen stänga av de portar kabeln är kopplad till. Det kommer då i praktiken fungera som att kabeln inte finns. Genom att koppla nätverksenheter i ett fullt mesh-nät kan man bygga upp den struktur man vill ha genom att inaktivera de länkar som inte ska användas. Sedan kan man fjärransluta och stänga av eller aktivera länkar för att bygga upp den struktur man behöver. För att kunna fjärrkonfigurera nätverksenheter bör man inte förlita sig på nätverksanslutningar direkt till dem då det kanske inte alltid är säkert att de är nåbara. Man bör därför använda en console-server eller liknande för att konfigurera switcharna och denna bör vara kopplad till en stabil del av nätverkstrukturen. [18]

Ett problem med att installera operativsystem på servrar är att det kräver fysisk närvaro tills dess att man har konfigurerat servern för att ta emot fjärranslutningar. För att ta bort behovet av fysisk närvaro behöver man möjliggöra fjärråtkomst till servern innan dess att ett operativsystem är installerat. En bra lösning är att använda ett RAC. Virtualisering löser också problemet och ger andra fördelar på köpet. Om man inte har ett RAC behöver man bara vara fysiskt närvarande när man installerar hypervisorn. Sedan kan man lätt skapa virtuella maskiner och installera operativsystem på dem när man ansluter till hypervisorn. Om man virtualiserar på servern bör den administrativa nätverksporten vara kopplad till en stabil del av nätverket.

#### 4.1.2 Återställning till ursprungsläge

Det arbete som går åt till att installera om mjukvara på servrar och att konfigurera nätverksutrustning kan ta onödig tid. Det kan i vissa fall vara effektivt att ha ett ursprungsläge att återgå till. Med ”återställning till ursprungsläge” menas att operativsystemet och alla dess filer, eller konfiguration i nätverksutrustning, tas tillbaka till ett sparad läge.

I nätverksenheter vill man spara konfigurationen. Man bör därför ha rutiner/metoder för att spara och återställa konfigurationen. De flesta nätverksenheter för professionellt bruk kan ladda upp och ladda ner sin konfiguration till/från en TFTP-server. [22] Det går även att ansluta till nätverksenheten på valfritt sett för att säkerhetskopiera konfigurationen.

Hårddiskkloning är ett sätt att återställa en server till ett visst läge men är troligen inte en optimal teknik i en servermiljö annat än i säkerhetskopieringssyfte. Det är en långsam process som antingen kräver fysisk närvaro vid servern eller utnyttjar stora nätverksresurser då avbilden återställs. Det behöver även finnas en fysisk server för varje operativsystem som körs. Det är det enklare, smidigare och snabbare att ta backup på en virtuell hårddisk i en virtuell miljö. En virtuell maskin består av ett antal filer och man kan skapa en identisk maskin genom att kopiera dessa. För att återställa en server till ursprungsläge kan man ta backup på dessa filer. Beroende på vilken virtualiseringsplattform som används så kan det finnas funktioner för att skapa snapshots eller templates. [23]

### 4.1.3 Rutiner

Ofta tar arbetssysslör mindre tid om man har en klart beskriven rutin för hur de ska genomföras. Därför bör man försöka dokumentera de arbetsrutiner som kan snabba upp arbetet. Förslagsvis används ett dokument för varje system/tjänst som används. [20] Dokumentation av rutinerna bör förutom innehållsförteckning innehålla följande för varje rutin:

- **Rubrik** – Beskrivande rubrik för rutinen.
- **Mål** – Kort beskrivning av vad rutinen ämnar åstadkomma.
- **Anvisningar** – Beskrivning av hur rutinen genomförs.

Efterhand är tanken att de som använder laborationsmiljön ska kunna lägga till nya rutiner allt eftersom.

### 4.1.4 Säkerhet

I en laborationsmiljö prioriteras ofta enkelhet och funktion före säkerhet. Det blir så då det oftast inte är en driftkritisk miljö man laborerar med. För att kunna få in ett viss säkerhetstänk bör man försöka skapa så mycket avskildhet från övrig serverdrift och nätverksstruktur som möjligt. [20] Mängden säkerhet som krävs skiljer sig ifrån fall till fall men punkter att ta hänsyn till är:

- **Fysisk avskildhet** – Avser placering av utrustning. Laborationsutrustningen bör kanske finnas i ett annat rum än den utrustning som används för företagets produktion. Genom att ha produktionsutrustning och laborationsutrustning i olika rum kan man ha olika rutiner för åtkomst av hårdvara. Man vill t.ex. kanske att många personer ska kunna nå laborationsutrustningen på ett smidigt sätt utan att behöva lätta på säkerheten för produktionsutrustningen.

- Logisk avskildhet – Innefattar IP-adressering och routing. Det är lämpligt att laborationsmiljön har egna IP-nät och åtkomst från dessa till produktionen begränsas. [20]

#### 4.1.5 Framtidssäkerhet

Med framtidssäkerhet menas att man tar hänsyn till hur laborationsmiljön kommer att användas längre fram i tiden. Om man inte har planerat ordentligt från början kan onödigt mycket tid gå åt till problemlösning längre fram. De servrar som används bör ha prestanda för att klara en framtida tillväxt. Man bör då utgå ifrån hur många virtuella maskiner man kommer att ha igång samtidigt och vilken typ av belastning de kommer att ha.

Hårddiskutrymme i virtuella miljöer kan gå åt snabbt när man skapar nya virtuella maskiner. Därför är det viktigt att planera för framtida lagring. Valet man står i är huruvida man ska använda intern eller extern lagring. Intern lagring innebär att de hårddiskar som sitter i servern används och extern lagring att man lagrar data på en filserver. Att använda extern lagring är dyrare och mer komplext men erbjuder en del fördelar. Bland annat att man kan satsa på att köpa mer lagringsutrymme som kan användas av flera servrar. Om man vill ha ett virtualiseringskluster kommer man dessutom behöva en filserver för att flera värdmaskiner ska kunna läsa virtuella maskiner från samma lagring.

Infrastrukturen, alltså nätverket, måste vara tillräcklig när det gäller antal enheter samt att den skall kunna leverera den funktionalitet och kapacitet som behövs. Man bör tänka på att anpassa laborationsmiljön efter kommande tekniker som man eventuellt kommer behöva stöd för i framtiden. Ett aktuellt exempel är IPv6. [24]

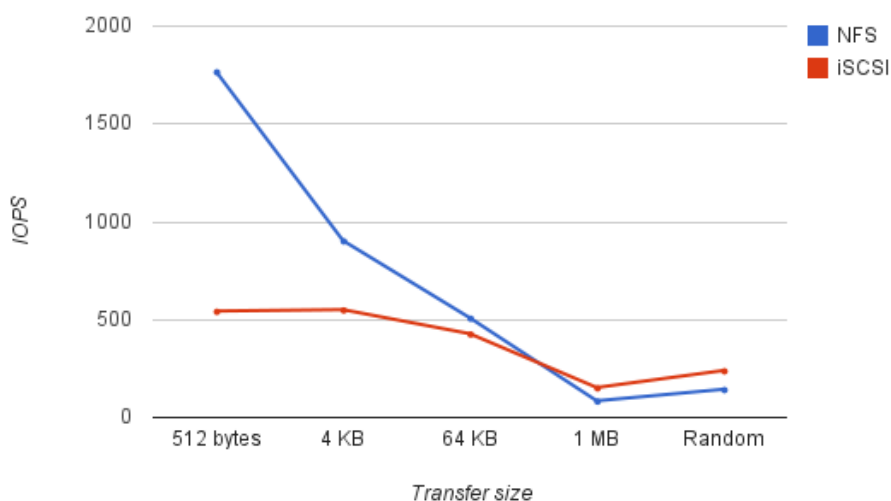
#### 4.1.6 Dokumentation

Laborationsmiljön bör vara dokumenterad. Dokumentationen bör vara enkel att uppdatera då miljön löpande förändras. Det innebär att den måste vara sparad i ett filformat som alla medarbetare kan läsa/redigera på ett lagringsmedium som går att komma åt över nätverket. Dokumentationen bör utgå från en inledning som förklarar hur det är tänkt att dokumentationen ska användas och har en innehållsförteckning. [25]

### 4.2 Extern lagring

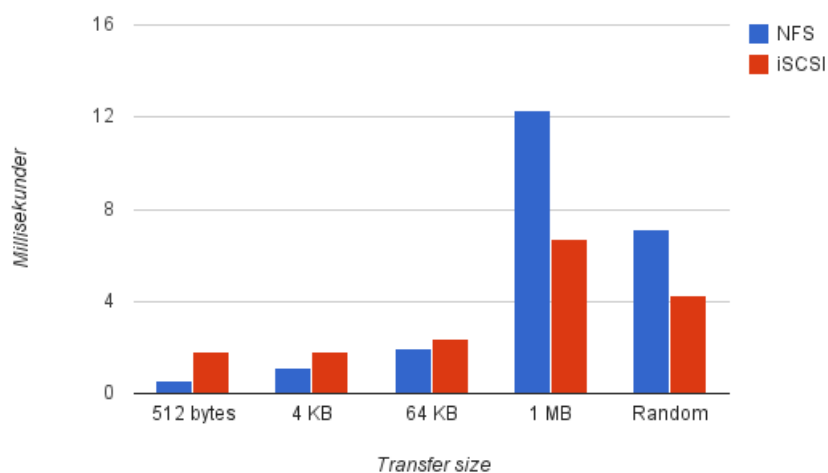
FreeNAS ansågs ha ett svårhanterat och ologiskt gränssnitt och saknade vissa konfigurationsmöjligheter. Det gick till exempel inte att ställa in mer än grundläggande inställningar för NFS, inga möjligheter till finjusteringar av prestandan. Under iSCSI fanns det istället så mycket inställningar att det hade tagit för lång tid att sätta sig in i alla. Istället testades då Openfiler, vilket hade ett betydligt mer organiserat och logiskt gränssnitt. Det fanns även information om vad varje variabel hade för funktion.

Antalet Input/Output Operations per Second (IOPS) är mycket större för NFS vid små överföringsstorlekar, vid 512 bytes klarar NFS mer än tre gånger så många IOPS. Vid större överföringsstorlekar blir skillnaden desto mindre tills iSCSI går om vid 1 MB överföringsstorlek. iSCSI har en rätare kurva jämfört med NFS som har en ganska brant nedåtgående kurva, se Figur 6.



Figur 6: IOPS jämförelse (Högre är bättre)

Resultaten får liknande karaktär när man mäter genomsnittlig accesstid, mätningarna kan ses i Figur 7. Här har dock inte NFS lika stort försprång vid små överföringsstorlekar, och halkar efter desto mer vid större storlekar. Vid 1 MB överföringsstorlek har iSCSI nästan dubbelt så kort accesstid jämfört med NFS. Även i den här mätningen visar iSCSI ett jämnare resultat än NFS.



Figur 7: Genomsnittlig accesstid (Lägre är bättre)

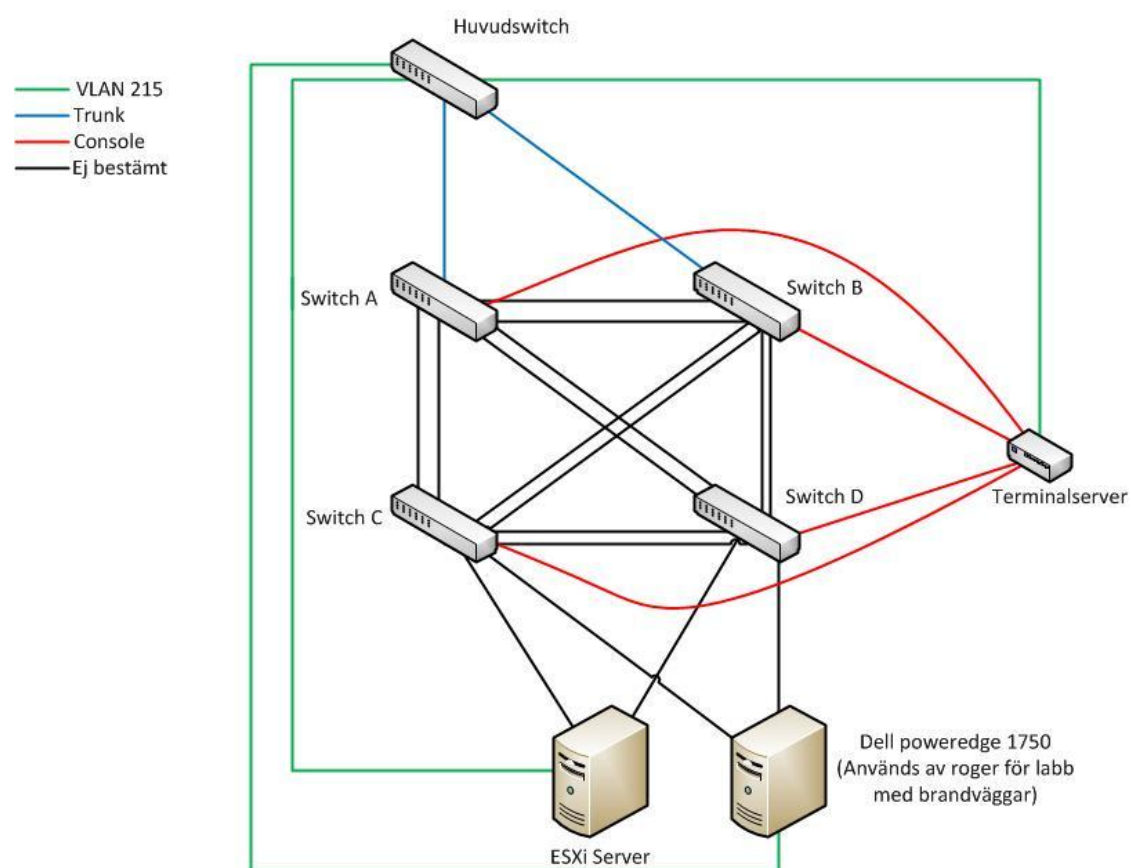
## 4.3 Förslag för TDCs laborationsmiljö

Resultatet för TDC innefattar ett förslag på nätverksstruktur och information om den server som står för virtualiseringen.

### 4.3.1 Nätverksstruktur

De fyra switcharna A, B, C och D (se Figur 8) är tänkta att användas för laboration. Då man kan använda console-servern för att konfigurera switcharna behöver man inte förlita sig på nätverksanslutningar till dem. Därmed kommer man alltid åt switcharna även om de nollställs. Switcharna är kopplade i en full mesh-topologi för att man ska kunna stänga av portar och på så sätt bygga upp den nätverksstruktur man vill ha under laboration.

ESXi servern bör ha minst ett interface kopplat till huvudswitchen, detta för att man alltid ska kunna ansluta till den med vSphere client. Servern har för närvarande bara två nätverksinterface, den kommer enligt Figur 8 att behöva minst 3 stycken (möjligen ett till för extern lagring). Alltså måste ett extra nätverkskort införskaffas för att tillgodose detta.



Figur 8: Förslag på nätverksstruktur

### **4.3.2 Servermiljö**

På servern körs i dagsläget ESXi. De virtuella maskinerna kördes tidigare i VMware Workstation. De är nu konverterade och körs i ESXi. Den virtuella maskinen som heter "vCenter" kör Windows Server 2008 R2. vCenter är installerat med en gratis 60 dagars utvärderings-licens. Det är upp till TDC själva att avgöra huruvida det är värt att investera i en licens. Skillnader mellan licenser finns i Tabell 1.

Nätverksinställningar är svårt att göra på servern eftersom det beror på hur TDC väljer att sätta upp sin miljö samt att fler nätverksinterface bör läggas till. Ett interface och tillhörande virtuell switch bör kopplas till TDCs stabila nätverk. Skall extern lagring användas bör ett interface användas endast för detta. Resterande interface bör kopplas till laborationsnätet.



## 5 Diskussion och slutsatser

Det har varit ett lärorikt examensarbete, speciellt inom virtualiseringsområdet. Förhoppningsvis har vi hjälpt TDC en bit på vägen till en effektivare labbmiljö.

### 5.1 Resultatdiskussion

Syftet har varit att ta fram riktlinjer för hur man bygger effektiva laborationsmiljöer. En del av syftet har också varit att till viss del exemplifiera riktlinjerna gentemot uppdragsgivaren TDC.

#### **Hur kan virtualisering effektivisera en laborationsmiljö för servrar och switchar?**

Under arbetet ringades ett antal områden in som kunde effektiviseras med hjälp av virtualisering. Den mest slående effektiviseringen är lättheten att snabbt skapa nya maskiner, speciellt genom att kлона befintliga maskiner. [23] En utökning på detta är så kallade snapshots, om man gör ett snapshot på en maskin innan man börjar laborera så kan man snabbt gå tillbaka till snapshoten om laborationen går dåligt. Virtuella maskiner innebär även att hårdvaran är flexibel, prestandan kan ökas vid behov och ny hårdvara kan läggas till på ett enkelt sätt. Detta kan även leda till effektivisering eftersom man kanske annars måste installera om systemet man testat på en annan server om prestandan inte räcker till eller om viss hårdvara som behövs saknas.

Den här frågeställningen är ganska öppen och omöjlig att svara helt på. Arbetet har dock visat att virtualisering markant minskar behovet av installation/ominstallation av maskiner, som har varit den största källan till onödig tid. Detta gör att frågeställningen får anses vara acceptabelt besvarad.

#### **Hur implementerar man detta praktiskt i en laborationsmiljö?**

En första steg till en virtuell miljö har tagits genom att ESXi har installerats på en server. Det medföljer även en virtuell maskin med vCenter som har en 60-dagars utvärderingslicens (det har även framförts ett förslag på en lämplig betallicens att använda). Med hjälp av denna server ska TDC kunna börja använda och testa virtualisering i VMwares plattform vSphere. Tidigare har man använt sig av VMware Workstation så för att kunna göra den övergången har maskinerna konverterats till ett format ESXi kan hantera.

Ett teoretiskt förslag på hur en nätverksstruktur för laborationsmiljön har tagits fram. Detta med hänsyn till de riktlinjer som har tagits fram. Genom att bygga upp sitt nätverk enligt förslaget samt koppla in den server som har preparerats anser vi att man har byggt en början till en effektivare laborationsmiljö.

#### **Hur kan behovet av fysisk tillgång till laborationsmiljön reduceras?**

Behovet av fysisk tillgång till servrar för konfiguration och installation reduceras kraftigt med virtualisering. Det är dock viktigt att virtualiseringsservern/servrarna har minst en anslutning till en stabil del av nätverket, alltså inte den ostabila laborationsmiljön. Servrar som inte är virtuella går också att komma åt om de har ett RAC, vilket de flesta nyare servrar har.

En console-server gör att nätverkshårdvara kan fjärrkonfigureras, även denna behöver vara kopplad till en stabil del av nätverket. Att koppla ihop switcharna i ett mesh-nät, helst med så många kablar som möjligt, gör att behovet att koppla om kablar minskar. Detta behov är dock extremt svårt att eliminera helt, eftersom en del laborationer kan kräva obskyra kabeldragningar. Rapporten belyser viktiga punkter för att reducera behovet av fysisk tillgång till ett absolut minimum för servrar och switchar, och frågeställningen känns därför besvarad.

## **5.2 Metoddiskussion**

Arbetet inleddes med en workshop på TDCs kontor i Växjö. Under denna framkom en del information om vad som upplevdes vara tidsödande och ineffektivt med den nuvarande situationen. Liknande information har kommit fram eftersom genom de telefonavstämningar som har ägt rum.

Det har varit svårt att hitta en teoretisk grund som inriktar sig på just laborationsmiljöer för nätverk. Istället har man fokuserat på att utgå ifrån vad TDC har upplevt som ineffektivt i sitt arbete. Resultatet är således grundat främst på observationer som kommit fram under arbetsprocessen och har därför en liten teoretisk grund att stå på.

Tanken från början var att man skulle bygga om TDCs laborationsmiljö utifrån de riktlinjer som togs fram. Genom att exemplifiera riktlinjerna mot TDC skulle de styrkas något. På grund av tidsbrist har ingen sådan ombyggnad genomförts. Däremot har lite praktiskt förändringar genomförts och förslag på nätverksstruktur tagits fram.

## **5.3 Slutsatser och rekommendationer**

Sammanfattningsvis så har det konstaterats att virtualisering kan vara effektiviserande i en laborationsmiljö och ett antal riktlinjer för att skapa en effektiv, flexibel och framtidssäker laborationsmiljö har tagits fram. Om företag (exempelvis TDC) följer dessa riktlinjer när de designar en laborationsmiljö så kommer denna miljö att spara tid åt företaget i fråga. Det är väldigt svårt att avgöra hur mycket tid man sparar genom dessa riktlinjer, eftersom detta beror på en mängd faktorer. Några exempel på faktorer som påverkar tidsvinsten är hur effektiv laborationsmiljön var innan och vad man laborerar med, olika typer av laborationer ger olika tidsvinster. Just i TDCs fall så var den största tidsförlusten ominstallation av servrar, eftersom just detta moment går markant fortare med virtualisering så ger detta en stor tidsvinst för TDC. Möjligheten att skapa nya maskiner från mallar och/eller befintliga maskiner samt att kunna spara gamla maskiner ger även det stora tidsvinster gentemot att alltid installera nytt, och därmed behöva ta bort det man höll på med innan.

Frågan om reducerad tillgång till laborationsmiljön fanns inte med från början, utan framkom under arbetets gång. Riktlinjerna har färgats av denna frågeställning, men frågan går absolut att fortsätta forska i. Ett examensarbete som utgår från denna frågeställning, eller om inte annat har med den från början skulle troligen kunna få ett mycket mer utförligt och nyanserat resultat. I exemplet med TDC så ger fjärråtkomst en stor fördel, nämligen att andra lokalkontor som inte har en egen laborationsmiljö (till exempel kontoret i Jönköping) nu kan få tillgång till en sådan miljö.

Om man inte planerar laborationsmiljön med hänsyn till framtiden kan det leda till problem såsom att man saknar funktionalitet eller kapacitet. Att då behöva inventera i ny dyr utrustning när man från början hade kunnat undvika problemen genom att köpa något dyrare utrustning är resursslöseri. De servrar och nätverksenheter som används bör ha den funktionalitet och kapacitet som förväntas behöva längre fram i tiden. När det gäller virtualiseringen är det viktigt att avgöra vilken typ av lagring som ska användas. De tre typer som finns att välja mellan är Intern lagring, NAS och SAN. Om man virtualiserar i kluster och vill använda avancerade funktioner för att öka tillgänglighet behövs någon form av nätverkslagring, alltså NAS eller SAN. I virtualiseringkluster kräver vissa funktioner att flera fysiska servrar kan komma åt samma virtuella maskin. Det blir då problematiskt att ha maskinerna sparade på intern lagring. Det kommer leda till att mycket stora filer måste flyttas mellan servrar, vilket belastar nätverket hårt.

## 6 Referenser

- [1] U. Pawar och M. Bhelotkar, "Virtualization: A Way Towards Dynamic IT," i *International Conference and Workshop on Emerging Trends in Technology*, Mubmai, 2011.
- [2] Office of Government Commerce, "Glossary," i *The official introduction to the ITIL service lifecycle*, TSO, 2007, p. 192.
- [3] J. Åsblom, "Norden bäst på virtualisering : 43 procent av företagen köper virtualiserade servrar," Juni 2011. [Online]. Available: <http://www.idg.se/2.1085/1.176650>.
- [4] "Virtual Machine Management," i *Mastering Virtual Machine Manager 2008 R2*, John Wiley and Sons, 2011.
- [5] Office of Government Commerce, "Introduction," i *The official introduction to the ITIL service lifecycle*, TSO, 2007, pp. 3-4.
- [6] A. Pedersen, Red., "The Answer: Virtualization Is...," i *Best Damn Server Virtualization Book Period*, Burlington, Syngress Publishing, INC, 2007, pp. 6-7.
- [7] S. Winter, "Concept for system virtualization in the field of high availability computing," Norderstedt, 2007.
- [8] VMware, Maj 2011. [Online]. Available: <http://www.vmware.com/products/vsphere-hypervisor/overview.html>.
- [9] VMware, 10 Maj 2011. [Online]. Available: <http://www.vmware.com/products/vsphere/overview.html>.
- [10] VMware, Juni 2011. [Online]. Available: [http://www.vmware.com/products/vsphere/buy/small\\_business\\_editions\\_comparison.html](http://www.vmware.com/products/vsphere/buy/small_business_editions_comparison.html).
- [11] "Intel Virtualization," i *Concepts, Terms, and Techniques for successfully Planning, Implementing and Managing Enterprise IT Virtualization Technology*, 2008, pp. 49-50.
- [12] Intel, 10 Maj 2011. [Online]. Available: [http://download.intel.com/technology/virtualization/320426.pdf?iid=tech\\_vt\\_server\\_hw+rhc\\_brochure](http://download.intel.com/technology/virtualization/320426.pdf?iid=tech_vt_server_hw+rhc_brochure).
- [13] D. Sacks, "Demystifying Storage Networking," 2001.
- [14] Internet Engineering Task Force, "RFC 5661 - Network File System (NFS) Version 4 Minor Version 1 Protocol," 2010.
- [15] J. Satran, K. Meth, C. Sapuntzakis, M. Chadalapaka och E. Zeidner, "RFC 3720 - Internet Small Computer Systems Interface (iSCSI)," 2004.
- [16] "Disc Cloning," i *Mike Meyers' CompTIA A+ Guide to Managing and Troubleshooting PCs*, The McGraw-Hill Companies, 2007, p. 447.
- [17] "Network Topologies," i *CCNA INTRO Exam Certification Guide*, Cisco Press, 2003, pp. 283-287.
- [18] J. D. Sloan, "A remotely accessible networking laboratory," *Journal of Computing Sciences in Colleges*, vol. 18, nr 2, p. 220, December 2002.
- [19] A. Muller och S. Wilson, "Remote Access Card," i *Virtualization with VMware ESX Server*, Syngress Publishing, 2005, p. 399.

- [20] A. Nauman och S. Söndergaard, ”Riktlinjer för implementering av ISO 17799,” Jönköping, 2010.
- [21] EFD Software, Juli 2011. [Online]. Available: <http://www.hdtune.com/>.
- [22] D. Jones, *The Shortcut Guide to Automating Network Management and Compliance*, Realtimepublishers.com, 2006, pp. 54-55.
- [23] M. Poniatowski, *Foundations of Green IT: Consolidation, Virtualization, Efficiency, and ROI in the Data Center*, Pearson Education, 2009, p. 4.
- [24] D. Dongo, ”Övergången från IPv4 till IPv6: varför dröjer den?,” Skövde, 2005.
- [25] M. C. Langston, ”Introduction,” i *Documentation Writing for System Administrators*, USENIX Association, 2003, pp. 1-4.

## 7 Sökord

återställning .....	17	mesh .....	11, 21, 24
console-server .....	17, 24	NAS .....	8, 9
datastore .....	8	NFS .....	9, 10, 15, 20
DRS .....	8	Openfiler .....	10, 15, 19
effektivitet .....	4, 17, 23	riktlinjer .....	4, 5, 13, 14, 23, 24
ESXi .....	7, 8, 10, 15, 21, 22, 23	rutiner .....	18
Fjärråtkomst .....	2, 17	SAN .....	8, 9
framtidssäkerhet .....	19	TDC .....	4, 5, 13, 15, 21, 22, 23, 24
FreeNAS .....	10	TFTP .....	12, 18
Intel VT .....	8, 15	vCenter .....	8, 15, 22, 23
iSCSI .....	9, 10, 15	virtualisering .....	4, 5, 6, 8, 15, 23
ITIL .....	4, 6	VMFS .....	15
kloning .....	7	VMware .....	5, 7, 15, 22, 23
lösningsförslag .....	13	vSphere .....	7, 15, 21, 23