FOI-R--1581--SE
Februari 2005
ISSN 1650-1942
Metodrapport

Daniel Fäldt

Avancerad systemadministration i Irix, kurssammanfattning, slutsatser och idéer

Avdelningen för stridssimulering, FLSC
172 90 Stockholm

TOTALFÖRSVARETS FORSKNINGSINSTITUT
Avdelningen för stridssimulering, FLSC
172 90 Stockholm

FOI-R--1581--SE
Februari 2005
ISSN 1650-1942
Metodrapport

Daniel Fäldt

Avancerad systemadministration i Irix, kurssammanfattning, slutsatser och idéer

Utgivare Totalförsvarets Forskningsinstitut - FOI Avdelningen för stridssimulering, FLSC 172 90 Stockholm	**Rapportnummer, ISRN** FOI-R--1581--SE	**Klassificering** Metodrapport
	Forskningsområde 2. Operationsanalys, modellering och simulering	
	Månad, år Februari 2005	**Projektnummer** E52207
	Delområde 24 Luftstridssimuleringscenter	
	Delområde 2	
Författare/redaktör Daniel Fäldt	**Projektledare** Daniel Fäldt	
	Godkänd av Anders Borgvall	
	Uppdragsgivare/kundbeteckning Försvarsmakten	
	Tekniskt och/eller vetenskapligt ansvarig Folke Stoby	

Rapportens titel

Avancerad systemadministration i Irix, kurssammanfattning, slutsatser och idéer

Sammanfattning (högst 200 ord)

Det finns många konfigurationsparametrar och ett väsentligt antal konfigurationsprogram som syftar till att höja prestanda, förstärka säkerheten, förenkla konfigurationen och dylikt när det gäller operativsystemet Irix från SGI. Några sådana intressanta verktyg kan vara LDAP, RoboInst eller tcp_wrapper. Hur skall man egentligen ställa sig till verktyg som dessa? Hur konfigurerar vi egentligen våra nätverk för att passa våra syften. I denna rapport presenteras olika parametrar och program som kan vara bra att känna till vid design och konfigurering av nätverk bestående av främst Irixbaserade datorer, men även vissa Windowsdatorer och Linuxdatorer.

Nyckelord

Irix, systemadministration, sys adm, ldap, nis, dns, dhcp, nätverk, säkerhet

Övriga bibliografiska uppgifter	**Språk** Svenska
ISSN 1650-1942	**Antal sidor:** 21 s.
Distribution enligt missiv	**Pris: Enligt prislista**

Issuing organization	Report number, ISRN	Report type
FOI – Swedish Defence Research Agency Combat Simulation, FLSC SE-172 90 Stockholm	FOI-R--1581--SE	Methodology report
	Programme Areas 2. Operational Research, Modelling and Simulation	
	Month year February 2005	**Project no.** E52207
	Subcategories 24 Air Combat Simulation Centre	
	Subcategories 2	

Author/s (editor/s)	Project manager
Daniel Fäldt	Daniel Fäldt
	Approved by Anders Borgvall
	Sponsoring agency Swedish Armed Forces
	Scientifically and technically responsible Folke Stoby

Report title (In translation)

Irix Advanced Network Administration, course summary, conclusions and ideas

Abstract (not more than 200 words)

In the OS Irix from SGI there are many different configuration parameters to tune and configuration programs to use to increase the performance, strengthen the security and simplify the future configurations. Some of them could be, for example, LDAP, RoboInst or tcp_wrapper. What about these programs? Are they really useful? In this report we will discuss different kinds of configuration tools and parameters used on Irixbased machine.

Keywords

Irix, system administration, sys adm, ldap, nis, dns, dhcp, network, security

Further bibliographic information	Language Swedish
ISSN 1650-1942	**Pages** 21 p.
	Price acc. to pricelist

Sammanfattning

Det finns många konfigurationsparametrar och ett väsentligt antal konfigurationsprogram som syftar till att höja prestanda, förstärka säkerheten, förenkla konfigurationen och dylikt när det gäller operativsystemet Irix från SGI. Några sådana intressanta verktyg kan vara LDAP, RoboInst eller tcp_wrapper. Hur skall man egentligen ställa sig till verktyg som dessa? Hur konfigurerar vi egentligen våra nätverk för att passa våra syften. I denna rapport presenteras olika parametrar och program som kan vara bra att känna till vid design och konfigurering av nätverk bestående av främst Irixbaserade datorer, men även vissa Windowsdatorer och Linuxdatorer.

Innehåll

1 Introduktion

Under december månad 2004 genomfördes en utbildning i avancerad nätverksadministration i operativsystemet Irix. Detta då Stridssimulering, FLSC inköpt en ny datorpark där stor del av dem består av just SGI-datorer med operativsystemet Irix. Målet med utbildningen var att öka kunskapen om olika nätverkslösningar för nästa generation av simulatorparken. Detta för att eventuellt kunna öka kvalitén och effektivitetet på området samt att eventuellt öka säkerheten mot intrång och dylikt i systemet.

Stridssimulering, FLSC avser att se över nätverkets konstruktion och dylikt. Denna rapport avser till att sammanfatta kursinnehållet och beskriva de, för FLSC, väsentliga komponenterna samt att ge idéer och förslag på åtgärder som skall, alternativt bör, vidtas.

Rapporten är i stora delar en sammanfattning av kursmaterialet samlat i IRIX® Advanced Network Administration – Student Workbook (AINA-3.0-I6.5-S-SD-W) samt IRIX® Advanced Network Administration – Laboratory Workbook (AINA-3.0-I6.5-S-SD-L).

Då det finns många versioner av Irix och dessa har olika versioner och utbud av medföljande programpaket så förutsätter detta dokument att Irix av version 6.5.14 används då inget annat specifikt anges.

2 Grundläggande nätverksteori

Nätverkskommunikation över Ethernet, TCP/IP samt tillhörande lager är uppdelade i en struktur enligt Tabell 2:1 nedan.

Application	Telnet	ftp	rsh	Ping	NFS
Transport	TCP			UDP	
Internet	ICMP		IP		IGMP
Network interface	Ethernet	ATM	SLIP/PPP		FDDI
Hardware	10baseT	100baseT	1000baseT	Serial	Fiber

Tabell 2:1 Överblick över nätverkslagren i TCP/IP/Ethernet

Dessa skiljer sig i vissa drag från ISO[1] och ITU-T[2]s gemensamma modell Open System Interconnection mera känd i sin förkortade form som OSI-modellen.

Varje lager ansvarar för en del av kommunikationen mellan sändande och mottagande enheter i nätverket. Varje lager har sina tillhörande ”headers” och i vissa fall även någon form av felkontroll med information för just detta lager. Till exempel så adderar Internetlagret information om sändande ip-nummer och mottagande ip-nummer om IP används.

[1] International Organization for Standardization
[2] Telecommunication Union-Telecommunications Standards Sector

2.1 IPv4 versus IPv6

Internetprotokollets version 4, även kallad IPv4 har en header på 32 bitar för varje ip-adress. Dessa kan användas på ett antal olika sätt beroende på vilken klass av nätverk som används.

Följande definitioner av klasserna finns:

Klass A IP-serie: 1.xxx.xxx.xxx – 127.xxx.xxx.xxx Antal nätverk: 128 Antal ip-nummer: 16 777 214 127.xxx.xxx.xxx är reserverad för loopback[3]
Klass B IP-serie: 128.xxx.xxx.xxx – 191.xxx.xxx.xxx Antal nätverk: 16 384 Antal ip-nummer: 65 534
Klass C IP-serie: 192.xxx.xxx.xxx – 223.xxx.xxx.xxx Antal nätverk: 2 097 152 Antal ip-nummer: 254
Klass D IP-serie: 224.xxx.xxx.xxx – 239.xxx.xxx.xxx Multicast-adresser
Klass E IP-serie: 240.xxx.xxx.xxx – 254.xxx.xxx.xxx Reserverade adresser

Tabell 2:2 Klassindelning IPv4

När val av ip-adresser görs görs dessa alltså inom valda intervall (klasser). Behövs 100 adresser väljs lämpligen klass C-adresser vilket betyder att 154 adresser kommer bli oanvända på detta nätverk. Denna hieraki gjorde att antalet ip-adresser som faktiskt användes var endast 3% av de faktiskt möjliga adresserna. (ref. 1)

Den lösning som uppkom i och med detta var att införa en så kallad nätmask, med denna kan man på ett annat sätt dela upp ip-adresserna i någon form av klasslöst gränssnitt.

2.2 Kort om nätmask

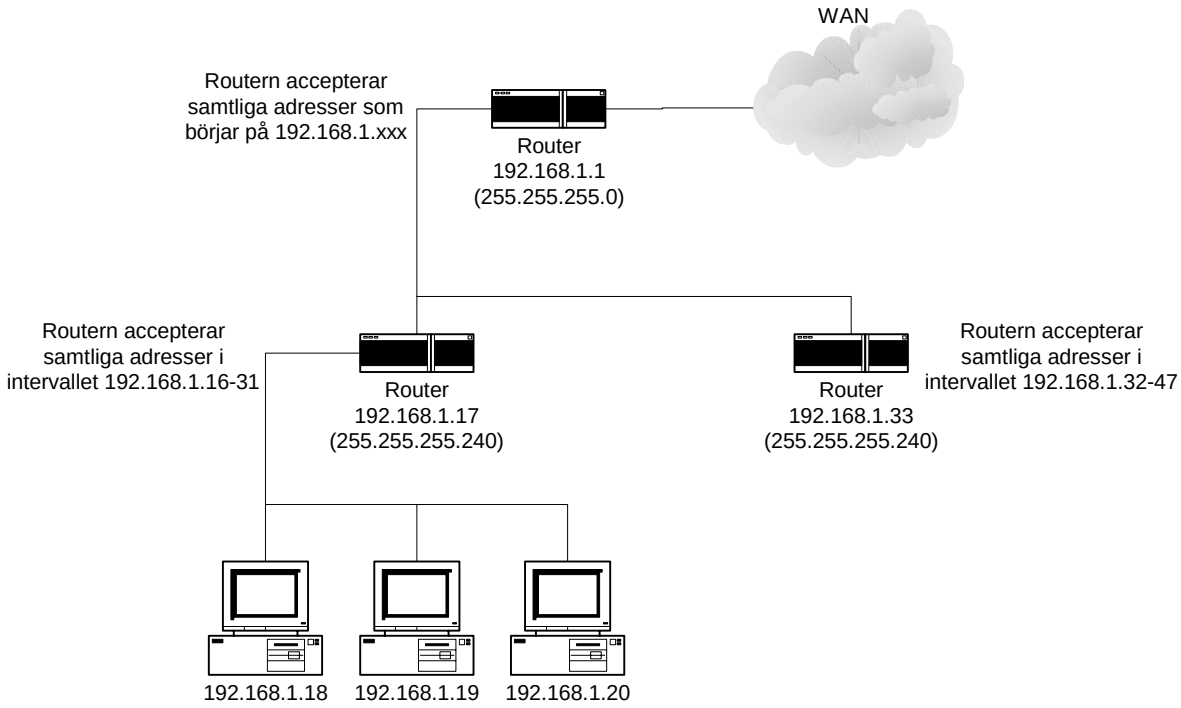
En nätmask används för att tala om för, till exempel en router, vilka ip-adresser just denna dator skall acceptera. Nätmasken adderas bitvis till routerns alternativt datorns ip-adress för att få ett antal ip-adresser.

Anta att ip-adressen 192.168.1.17 ges en router och att nätmasken 255.255.255.240 väljs till denna dator. För att se vilka ip-adresser routern accepterar konverterar vi de båda adresserna och gör en bitvis multiplikation av dem enligt följande

```
 11111111.11111111.11111111.11110000 (255.255.255.240)
 11000000.10101000.00000001.00010001 (192.168.  1. 17)
-----------------------------------
 11000000.10101000.00000001.00010000 (192.168.  1. 16)
```

[3] Loopback betyder att ingen kommunikation med nätverkskortet tas utan data loopas direkt tillbaka via högre lager (localhost).

Som vi kan se ovan är det endast de fyra sista bitarna i nätmasken som kan påverka ip-numret. Det är dessa fyra bitar som routern ignorerar i sin kontroll av ip-nummer. Alltså accepteras samtliga ip-nummer som uppfyller villkoret på övriga bitar. De möjliga ip-adresserna blir alltså 00011111-00010000 vid kontroll av den sista oktetten, det vill säga 192.168.1.16-192.168.1.31. För ett mer illustrativt exempel se Figur 1 nedan.



Figur 1 Ett exempelnätverk för att illustrera nätmaskens funktion.

Lösning med nätmasker var dock endast en temporär lösning i väntan på ett nytt protokoll. Vad man gjort med ip-adresserna i version 6 är bland annat att utöka dess storlek från 32 bitar till 128 bitar vilket naturligt ger betydligt fler adresser. Dessutom valde man att placera övriga headers i IP-protokollet som valbara istället för låsta till fasta positioner och storlekar. Detta gör att vissa paket-headers, som inte behöver alla headers, blir mindre.

2.3 Kort om IPv6-adresser

Då ip-nummerserierna är betydligt längre i version 6 än i version 4 är det högst olämpligt att representera ip-adresserna med decimala tal som brukligt i IPv4 därför representeras IPv6-numren med kommaseparerade hexadecimala tal (i grupper om fyra hexadecimala tal). Exempelvis är följande adress en giltig adress 0000:0000:0000:0000:0192:0168:0002:0005. För att förenkla adresserna något kan inledande nollor i varje grupp tas bort, vilket gör att föregående adress även kan skrivas som 0:0:0:0:192:168:2:5. Grupper av endast nollor i serie kan dessutom ersättas av ::. Detta kan naturligtvis endast ske på ett enda ställe i ett ip-nummer för entydighetens skull. Föregående ip-nummer skulle då kunna skrivas om enligt följande ::192:168:2:5. Dessutom fungerar ”.” lika bra som ”:” i serien vilket ger följande utseende ::192.168.2.5. Den vane IPv4-användaren ser då snabbt att denna adress är väldigt lik en IPv4-adress. Detta är naturligtvis ingen slump. För att översätta en IPv4-adress till IPv6 kan man alltså addera :: i början av själv adressen.

Multicast-adresser börjar i IPv6 med `FF1:`, loopback (`localhost`) är `::1`.

2.4 Irix och IPv6

För att aktivera IPv6 i Irix måste paketet `eoe.sw.ipv6` installeras. Därefter sker aktiveringen genom följande inställningar:

systune ip6_enable 1
`chkconfig ndpd on` (Konfiguration av IPv6 Neighbor Discovery daemonen)
`autoconfig -fv` (Bygga om kärnan)
`init 6` (Omstart)

Därefter bör man aktivera och starta upp routing för IPv6 också enligt följande:

`systune ip6forwarding 1` (Om systemet skall vidarebefordra ip6-paket[4])
`chkconfig route6d on` (Möjliggör routing)
`/usr/etc/route6d` (Starta routing-demonen)

2.5 Kort om Multicast

Multicast är en metod att nå flera nätverksgränssnitt[5] utan att skicka paketen separat till var och en av dem. Genom multicast skickas ett paket endast en gång. Detta paket når samtliga som prenumererar på dem. Multicast använder sig av ip-nummer i klass D enligt följande tabell.

224.0.0.1	All systems on this subnet
224.0.0.2	All routers on this subnet
224.0.0.5	OSPF routers
224.0.0.6	OSPF designated routers
224.0.0.12	DHCP server/relay agent
224.0.1.0-238.255.255.255	Globalt skyddat utrymme för till exempel NTP.

Tabell 2:3 Sammanfattning av de vanligaste multicast-ip-adresserna. (ref. 3)

För att prenumerera på data till en multicastgrupp används Internet Group Management Protocol (IGMP).

Multicast skall inte blandas ihop med broadcast vilket når samtliga nätverksnoder utan att prenumeration krävs.

3 Nätverksövervakning, justering och säkerhet

3.1 Övervakning

Det finns ett antal verktyg för Irix som lämpligen kan användas för att kontrollera nätverksinställningar samt övervaka nätverkstrafiken.

[4] Vilket betyder att systemet vidarebefordrar paket mellan nätverksinterfacen likt en router. Detta är vanligen standard i Irix och IPv6 om systemet har mer än ett nätverkskort.
[5] Med nätverksgränssnitt syftas här nätverkskort, router och liknande.

ifconfig	Identifiera nätverkstrafik, typ, hastighet, nätmask, multicast-adress mm. `ifconfig -av` Alla nätverksinterface
netstat	Identifiera nätverksstatus och statistik. Ett antal alternativ finns. För mer information se `man netstat`. `netstat -i [sec]` Kontrollera nätverkstrafik, där `[sec]` är tidsintervallet och, om uteslutet, sedan interfacet startades. `netstat -iq`
ping	Kontrollera nätverksåtkomligheten i systemet.
traceroute	Följ routingvägar mellan nätverkspunkter i nätverket.
ttcp	Mät TCP- samt UDP-genomströmningen.
nettest	Ett utförligare ttcp-test
spray	Mät genomströmningen i RPC-protokollet[6].
nfsstat	Kontrollera nfs- och rpc-interfacen mot kärnan (kernel)
gr_osview	Grafisk systemövervakning

Tabell 3:1 Verktyg för övervakning och justering av nätverk.

3.2 Justering

Ett användbart verktyg för att justera kärnans tcp/ip-parametrar är `systune`. Verktyget kontrollerar ett antal nätverksparametrar som `mtu-storlek`, `tcp_ttl`[7]. Dessa parametrar är

[6] RPC = Remote Procedure Call.
[7] TTL = Time-To-Live

dock endast ett fåtal av de parametrar som verkligen kan justeras. Samtliga parametrar kan hittas i `/var/sysgen/mtune/bsd`.

3.2.1 NFS

Något som kan påverka prestandan vad gäller nätverkskommunikationen betydligt är NFS Read och NFS Write. Dessa justeras med parametern `nsf3_default_xfer` för NFS3. Justering kan också ske vid själva monteringen med `mount -o rsize=xxx, wsize=yyy server:/dir`, där xxx och yyy bör väljas till multiplar av 512, i annat fall avrundar systemet detta själv.

Vid problem i nätverkstrafiken med många paketförluster (packet-loss) kan det vara värt att sänka värdet på läs- och skrivstorlekarna. Detta speciellt om UDP-trafik är vanligast i nätverket eftersom UDP inte håller koll på vilken ”frame” av datagrammet som försvunnit och en omsändning av hela datagrammet måste ske.

3.2.2 MTU-storlek

MTU[8]-storleken, dvs den maximala storleken ett paket får ha. I ethernet är det maximala värdet MTU kan ha 1500 bytes dock har Irix implementerat en egen jumbo-storlek vilket möjliggör MTU-storlekar på upp till 9000 bytes. Vid kommunikation med andra operativsystem och vissa routers kan detta bli ett problem. Använd därför MTU-storlekar över 1500 med försiktighet!

Ett högt värde på MTU kan dock öka prestandan i nätverket då en sändning kan slippa bli fragmenterad eftersom operativsystemets kärna inte behöver lägga ner tid på att fragmentera och sätta ihop paket. Justeringen kan göras med verktyget `systune` alternativt direkt i respektive nätverksinterface konfigurationsfil (`ifconfig-#.options`[9]). `mtu 9000` för maximal storlek.

I Irix[10] finns möjligheten att slå på en funktion för att automatiskt detektera mottagarens maximala MTU-storlek. Funktionen kallas MTU Discovery och kan, och bör, sättas genom `systune tcp_mtudisc 1`.

3.2.3 Forwarding

Om en Irixdator har mer än ett nätverksinterface installerat aktiverar Irix automatiskt ”forwarding” mellan interfacen, vilket betyder att trafik på ena interfacet vidarebefordras till det andra så som på en router.

Detta kan naturligtvis i vissa fall vara högst olämpligt och dessutom påverka nätverkens prestanda. Funktionen kan slås av med `systune ipforwarding 0`. Vanligen är det lämpligare att använda en dedikerad router för routing-funktionen istället för Irixdatorn.

3.2.4 TCP och sekvensnummer

För kommunikation på transportlagret med TCP används ett sekvensnummer för att identifiera vilket paket i ordningen just detta paket har. Detta för att paketen skall placeras i rätt ordning hos mottagaren. I Irix är det initiala sekvensnumret i en sändning slumpmässigt valt.

Om en sändnings sekvensnummer är känt kan ”hackers” göra en så kallad spoofing-attack, det vill säga skicka paket som ser ut att komma från en känd källa men med destruktiv information för mottagaren.

---

[8] MTU = Maximum Transfer Unit
[9] Där # är numret på det nätverksinterfacet med början på 1 (obs, ej 0).
[10] Irix 6.2 eller högre.

För att ytterligare öka säkerheten kan slumpmässigheten i det initiala sekvensnumret ökas genom att slå på flaggan `tcpiss_md5` med till exempel kommandot `systune tcpiss_md5 1`. Detta gör att en mer avancerad algoritm används för att skapa numret. Detta med en md5-kryptering av bland annat nanotiden och mottagar- och avsändarip-adress/port. Funktionen aktiverar dessutom slumpvis vald port för trafiken istället för en inkrementellt vald port. Detta ökar naturligtvis också säkerheten betydligt. Det rekommenderas att denna funktion aktiveras!

3.2.5 Att filtrera paket i IP

Till Irix finns ett paket för filtrering av IP-trafik[11]. Paketet har funktioner för att filtrera ip-trafiken med avseende på mottagarens respektive avsändarens ip-adress, vilket nätverksinterface paketet kom till, ip-protokollets version respektive vilken mottagar- och avsändarport.

Med parametern `ipfilterd_inactive_behavior` kontrolleras vad som skall göras med paketen om demonen `ipfilterd` inte körs, till exempel om någon dödat demonen för att genomföra ett hackförsök. Därför bör parametern vara satt till till 1, vilket betyder att all trafik då avvisas istället för motsatsen om den är satt till 0.

3.3 Säkerhet

Vad gäller säkerheten kan man dela in den i tre delar: Hemlighållande, korrekthet och tillgänglighet.[12] Samtliga tre är viktiga för att säkerheten skall anses tillförlitlig.

Data och informationen bör vara lagrad så att obehöriga ej kan få tillgång till det, obehöriga skall inte heller kunna göra informationen och datat korrupt till exempel genom att skicka falska ip-paket eller dylikt. Inte heller skall obehöriga kunna attackera systemet på ett sådant sätt att behöriga användare inte kan accessa datat.

Vanliga metoder för att attackera nätverk är *snooping*, alltså helt enkelt avlyssna datatrafiken, *subverting* vilket innebär att man till exempel utnyttjar säkerhetsluckor i nätverksprogram för att ta sig in i systemet eller att skicka felaktiga paket för att påverka en befintlig anslutning eller att helt enkelt scanna ett interface efter öppna portar och sedan attackera via dem, även kallad port scanning.

Verktyget `snoop`[13] till Irix är ett program för att just avlyssna nätverkstrafiken.

Genom att enkelt avlyssna trafiken kan man till exempel detektera eller fånga lösenord som skickas okrypterat över nätverket, till exempel en inloggning via Telnet eller pop-mail. Lämpligt är därför att, i möjligaste mån, inte använda sådana program. För fjärrinloggning rekommenderas därför ssh instället för Telnet samt imap istället för pop.

3.4 Säkra Irix

I Irix finns ett antal inställningar som bör konfigureras för att säkra systemet mot angrepp. Några av dem presenteras nedan.

I Irix och övriga Unix-världen används `/etc/passwd`-filer, som är fullt läsbara för samtliga användare, för att hantera användare och dess lösenord, hemkatalog etc. Lösenorden är

[11] eoe.sw.ipgate
[12] Fri översättning av orden Secrecy, Accuracy, Avalibility
[13] Medföljer paketet nfs.sw.nfs.

förvisso krypterade men att tillåta användare att läsa denna fil kan ändå vara ett säkerhetshål. Genom att placera lösenorden i en `/etc/shadow`-fil som endast är läsbar av `root`-användare säkrar man därmed systemet betydligt.

För att ytterligare säkra upp systemet bör vilande konton, så som `Guest`, stängas för användning samt lösenord som är svåra att gissa väljas, det vill säga en blandning av siffror , bokstäver och specialtecken som inte på något sätt bildar ett ord och med tillräckligt många tecken.

Andra åtgärder är att stänga av demoner som inte används. Ett exempel på en sådan demon är till exempel `fingerd` som sällan används, ett annat kan vara `sendmail` eller Apache.

Irix har alltid varit känd för sin knapra säkerhet. Detta har dock ändrats succesivt. Numera finns ett grafiskt gränssnitt för att hjälpa administratören att öka säkerheten. Verktyget nås via System Managern.

3.4.1 .rhost och hosts.equiv

Filerna `~/.rhosts` och `/etc/hosts.equiv` är filer i vilka inställningar kan göras för vem som skall ges tillgång till datorn via fjärråtkomst utan att behöva uppge lösenord. I filen `/etc/hosts.equiv` anger `root`-användaren globalt för datorn medans användaren själv kan ange detta för sitt konto i filen `~/.rhosts`. Dessa båda filer **får ej** vara läs- och/eller skrivbar för någon annan användare än ägaren, det vill säga `chmod 600`!

Denna funktion kan både anses vara en säkerhetshöjande åtgärd men också ett säkerhetshål. Att tillåta användare att slippa ange lösenord gör att användarens lösenord inte kommer skickas okrypterade över nätverket, vilket naturligtvis är bra. Å andra sidan kan naturligtvis bristen på lösenord vara en stor säkerhetsrisk också.

Ett tänkbart scenario skulle kunna vara följande. Användare A tillåter användare B att få tillgång till sitt konto utan lösenord. Användare B har i sin `.rhosts`-fil tillåtit användare C att få tillgång till sitt konto. Användare C loggar nu in på användare Bs konto och därefter vidare till användare A. Detta helt utan lösenord och förmodligen inte vad användare A menat eller ens insett vara möjligt.

3.4.2 tcp_wrappers

I Irix finns numera en tcp-wrapper vilken omsluter inetd-portarna och möjliggör en accesskontroll på vilka värdar (hosts), domäner, NIS-grupp som får ansluta mot vilka tjänster (services).

`tcp_wrappern` aktiveras genom att `inetd` startas med `-t`-optionen.

`-t all` - För alla services
`-t on` - För services specificerade i `/etc/inetd.conf` genom att de markerats med ett `!` före.

För att `inetd` skall starta med optionen ovan skall denna, som vanligt, anges i filen `/etc/config/inetd.options`.

Åtkomsten kontrolleras sedan enkelt via filerna `/etc/hosts.allow` och `/etc/hosts.deny` med formatet `demon_lista : klient_lista [:optioner]`. Det första matchande kriteret som wrappern hittar följs. Först läses allow-filen, därefter deny-filen, sker ingen matchning i någon av filerna **ges åtkomst**.

Genom att placera `ALL : ALL` i filen `/etc/hosts.deny` stänger man av all tillgång till alla tjänster som inte specifikt tillåtits i `/etc/hosts.allow` alltså.

4 Nätverksstyrd konfiguration

4.1 NIS

Network Information Service (NIS) tillhandahåller en central styrning av konfigurationsfiler. NIS kan hantera till exempel `/etc/hosts`-filen centralt, likaså `/etc/passwd`. NIS är tillgängligt för flera platformar, förutom Irix även Linux, Unix och Windows.

Det finns tre typer av NIS-system: Huvudserver (master), slavserver och klient.

Masterservern är den som tillhandahåller alla originalkonfigurationsfilerna (i ASCII). Servern använder dessa filer och bygger en NIS-databas (`NIS-maps`, `mdbm`).

Slavservern tillhandahåller en kopia av `mdbm`-databasen från masterservern. En slavserver kan vara bra att ha utifall masterservern går ner samt för att fördela trafiken till servern (Load balance). Det är möjligt att ha hur många slavservrar som helst. I vissa tillämpningar är det till och med lämpligt att varje klient är en slavserver.

När klienten startas gör den en sökning på nätverket via broadcast och första server som svarar blir den server klienten kommer använda. Om administratören själv vill välja server kan detta göras med `ypset server`.

För att aktivera NIS måste paketet `nsf.sw.nis` vara installerat. Därefter startas NIS enligt följande:

Masterserver:

```
# chkconfig yp on
# chkconfig ypmaster on
# chkconfig ypserv on
# ypinit -m
# nsadmin restart
```

Slavserver:

```
# chkconfig yp on
# chkconfig ypmaster off
# chkconfig ypserv on
# ypinit -s
# nsadmin restart
```

Klient:

```
# chkconfig yp on
# chkconfig ypmaster off
# chkconfig ypserv off
# ypinit -c
# nsadmin restart
```

I samtliga fall måste först NIS-domännamnet sättas samt konfiguration ske.

Om fler än ett subnätverk skall försörjas via en masterserver krävs att denna, alternativt en slavserver, placeras på en gatewaydator[14] mellan de två näten eftersom broadcast endast når det egna subnätverket.

4.2 LDAP

Light-Weight Directory Access Protocol (LDAP) är, likt NIS, ett server-klient-protokoll för att tillhandahålla en katalog med konfigurationer.

LDAP kan användas för att tillhandahålla information från flera typer av databaser. Till skillnad mot NIS så kan LDAP hantera kryptering, autentisering och åtkomstkontroll.

I Irix heter serverdemonen `slapd`. Det finns även slavservers som kan användas för att ansluta ett subnät till ett annat och genom denna ge LDAP-information till hela subnätet. Slavdemonen heter `slurpd` i Irix.

4.3 DNS

Domain Name Service (DNS) är en service som mest används för uppslag av domännamn[15] till ip-nummer och från ip-nummer till domännamn.

För uppslag av ip-adresser görs detta genom att att vända ip-adressen och addera `.in-addr.arpa`. Till exempel om ett uppslag av ip-adressen `156.27.92.35` skall ske ställs en fråga på följande namn `35.92.27.156.in-addr.arpa`. Uppslag kan i Irix, och de flesta andra Unixbaserade operativsystem, göras med `nslookup`.

I Irix finns ett antal typer av dns-servrar. Dessa är masterserver, slavserver, cachande server och forwarding server[16]. Masterservern är en så kallad authoritative server, det vill säga den server som har beslutsrätten. Eftersom masterservern är den som styr de övriga laddar denna server alltid sin zondata från filer på den lokala disken. Normalt är denne server placerad inom domänen, men detta är inget krav.

Slavservern hämtar all sin data från masterservern vid uppstart och uppdaterar sedan dessa i intervall. Slavserverns uppgifter är huvudsakligen att agera backupserver för mastern samt att hjälpa till med så kallad balance load, alltså hjälpa till med frågeställningar och på så sätt jämna ut belastningen och därgenom avlasta masterservern.

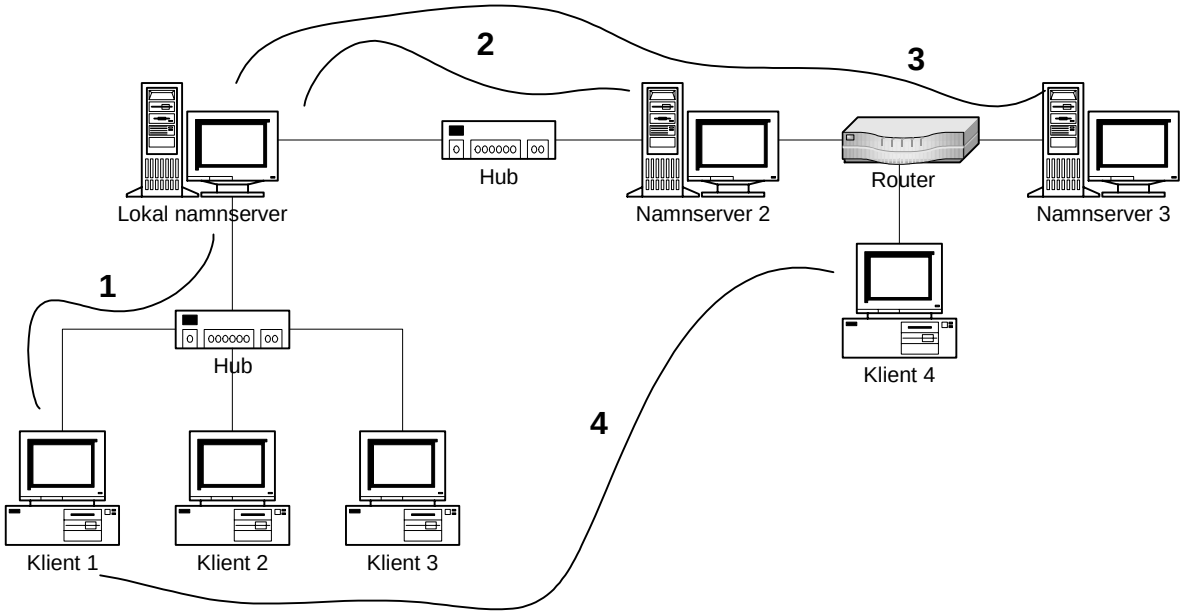
Den cachande servern har själv ingen lokal databas. Servern ställer vid förfrågan från klient frågan vidare till nästa server och vidarebefordrar sedan svaret till klienten. Servern temporärlagrar (cachar) sedan svaret för att slippa ställa frågan vidare nästa gång. Med vissa intervall töms sedan temporärlagringen. Denna typ av server används oftast som prestandaökande åtgärd.

Samtliga servrar kan agera som forwarding servrar. Det finns två typer av forwarding servrar, de som endast vidarebefordrar och de som även kan göra egna uppslag. Med forwarding server menas att alla frågor som ställs vidarebefordras till andra servrar ifrån den lista som servern har.

[14] I detta fall endast en an anslutning som ansluter två subnätverk med varandra.
[15] Till exempel www.foi.se
[16] Vidarebefordrande server

Det finns två typor av frågor som kan ställas till namnservern (dns-servern). De rekursiva frågorna och de iterativa. De rekursiva frågorna kräver svar från servern med lösningen alternativt ett felmeddelande om att svaret inte kunde hittas. De iterativa frågorna får förutom felmeddelande från servern även en ledtråd om annan namnserver som kan tänkas ha svaret. På så sätt kan svaret iterativt hittas.

Klienter ställer alltid rekursiva frågor till sin lokala namnserver. Har inte den lokala namnservern svaret skickar denne frågan vidare med en iterativ frågemetod.



Figur 2 Enkel modell över frågevägen för namnserverfrågor.

Låt säga att klient 1 i Figur 2 ovan skall ansluta till klient 4. Klient 1 ställer en domännamnsfråga för att finna klient 4. Frågan sker till den lokala namnservern (1). Låt säga att denne inte har svaret i sin lokala databas. Frågan ställs då vidare till namnserver 2 (2) och får ett negativt svar med en ledtråd om att namnserver 3 förmodligen har svaret på frågan. Den lokala namnserver gör då ytterligare en fråga, denna gång till namnserver 3, som returnerar ett positivt svar. Detta svar returneras sedan från den lokala namnservern till klient 1 som sedan kan ansluta till klient 4.

I stora nätverk finns ofta namnservrar som inte accepterar rekursiva uppslagsfrågor, detta av prestandaskäl.

4.3.1 DNS-serverkonfiguration och Irix

I Irix hanteras namnservern av `bind`. Nu gällande version är 9.2 och denna konfigureras med filen `/etc/named.conf`.

Konfigurationsfilen är numera uppbyggd på ett c-liknande[17] sätt och använder sig av `{` och `}` för att gruppera posterna, eller stanzas som de numera heter. Posterna kan nästlas. Kommentarer kan skrivas med `//` alternativt `/* */`, `#` fungerar också bra. Det finns ett

[17] Med c-liknande syftas på programmeringsspråket c.

gratisprogram som följer med `bind` med vilket man kan kontrollera om syntaxen i conf-filen är korrekt. Detta program kan hittas under `/usr/freeware/sbin/named-checkconf`.

En `named.conf`-fil för en masterserver kan se ut enligt följande:

```
options {
#
#
       directory "/var/named":
       forwarders {
              172.16.10.100;
       };
};

# type domain source host/file
zone "." {
       type hint;
       file "root.cache";
};
zone "sim.foi.se" {
       type master;
       file "named.hosts.sim.foi.se";
};
zone "10.16.172.in-addr.arpa" {
       type master;
       file "named.rev.sim.foi.se";
};
zone "0.0.127.in-addr.arpa" {
       type master;
       file "localhost.rev";
};
```

En `named.conf`-fil för en slavserver kan se ut enligt följande:

```
options {
#
#
       directory "/var/named":
       forwarders {
              172.16.10.100;
       };
};

# type domain source host/file
zone "." {
       type hint;
       file "root.cache";
};
zone "sim.foi.se" {
       type slave;
       file "named.hosts.sim.foi.se.bak";
       masters {
              172.16.10.2;
       };
};
zone "10.16.172.in-addr.arpa" {
       type slave;
       file "named.rev.sim.foi.se.bak";
       masters {
              172.16.10.2;
       };

};
zone "0.0.127.in-addr.arpa" {
       type master;
       file "localhost.rev";
};
```

Den sista posten (stanza) i de båda filexemplen ovan angående localhost är numera överflödigt. För att ytterligare öka säkerheten kan restriktioner göras för vilka som får ladda ner själva databasen från servern. Detta görs med att lägga till följande rad

```
options {
      allow-transfer { 192.16/16: };
}
```

För slavserver kan ipet lämpligen bytas mot `none`. Ingen skall ändå ladda ner slavserverns databas.

	Masterserver	Slavserver	Klient
`/etc/nsswitch.conf`	x	x	x
`/etc/resolv.conf`	x	x	x
cache- eller hint-fil	x	x	
zonfil	x		
Reverse zone file[18]	x		
Reverse loopback file[19]	x	x	
`named`-demon igång	x	x	

Figur 3 Sammanfattande tabell över vad som krävs för att respektive funktion skall fungera som och med namnserver.

4.3.2 Klientkonfiguration

Skapa filen `/etc/resolv.conf` och fyll i domännamn respektive namnservrar som skall användas. I den ordning de skall användas.

Exempel på `/etc/resolv.conf`:

```
domain       sim.foi.se
nameserver   172.16.10.2
nameserver   172.16.10.3
```

Skapa även filen `/etc/nsswitch.conf`, om den inte redan finns, och konfigurera så att dns används genom att ändra raden `hosts: files nis dns` efter önskat funktion.

Starta därefter om `nsadmin` med `nsadmin restart` för att införa ändringarna. Lämpligen görs nu en test så att klienten fungerar. Detta genom att använda `nslookup`, vilket inte använder sig av eventuella `hosts`-filer och dylikt utan endast dns.

4.3.3 Masterserverkonfiguration

Konfigurera `/etc/named.conf` enligt tidigare beskrivning i kapitel 4.3.1. Skapa därefter katalog `/etc/named/` respektive filerna `root.cache`, `named.hosts`, `named.rev`, `localhost.rev`.

Konfigurera därefter systemet så att named startas vid uppstart med `chkconfig named on` och starta `named` manuellt med `named`.

Därefter konfigureras servern som klient enligt kapitel 4.3.2 ovan.

[18] Omvänd zonfil.
[19] Omvänd återkopplingsfil.

4.3.4 Slavserverkonfiguration

Konfigurera `/etc/named.conf` enligt tidigare beskrivning i kapitel 4.3.1. Skapa därefter katalog `/etc/named/` respektive filerna `root.cache`, `localhost.rev`.

Konfigurera därefter systemet så att named startas vid uppstart med `chkconfig named on` och starta `named` manuellt med `named`.

Övriga filer kopierar sig nu själv från masterservern. Kontrollera att så sker.

Därefter konfigureras servern som klient enligt kapitel 4.3.2 ovan.

4.4 DHCP

Dynamic host configuration protocol (DHCP) används för att dynamiskt allokera hostnamn[20] och nätverksadresser.

Att ansluta en klient till ett nätverk med en eller flera dhcp-servers sker genom att klienten skickar ut en `DHCPDISCOVER` på nätverket via UDP-broadcast innehållande viss klientinformation. Samtliga servers i nätverket kontrollerar mot sin egen konfiguration om den är tänkt att hantera klienten. Kontrollerar därefter om det finns förkonfigurerade ip-nummer mappade till hårdvaruadressen, om inte så genereras ip-nummer. Servern skickar sedan tillbaka ett `DHCPOFFER` till klienten som accepterar det första giltiga erbjudandet och skickar ett svar med `DHCPREQUEST` med alla konfigurationsparametrar inklusive vald dhcp-server via broadcast till alla. Vald server lägger till klienten i sin databas och skickar ett `DHCPPACK` tillbaka som svar på detta.

Det finns två typer av dhcp-servers, masterserver och relayserver[21]. De båda servrarna konfigureras via det grafiska gränssnittet som nås via `ProclaimServerMgr` respektive `ProclaimRelayMgr`. Masterserverprogrammet som körs är `dhcp_bootp` och för relayserverdemonen `dhcp_relay`. Att tänka på är att bootp inte bör köras parallellt med dhcp då dessa kommer att krocka.

För att aktivera dhcp för klienterna körs kommandot `chkconfig autoconfig_ipaddress on`. Starta därefter om datorn alternativt kör programmet `run-proclaim start`.

Förutom att tilldela ip-nummer kan dhcp användas för att ge information om NIS-servrar i nätverket och mycket mera.

5 Nätverksstyrd installation med RoboInst

RoboInst är ett verktyg för Irix med vilket nätverksstyrd installation av operativsystemet samt programvara i detta kan ske. Detta förenklar i många fall själva installationsförfarandet om systemet innehåller många liknande installationer och hårdvarutyper.

För att sätta upp RoboInst krävs fyra olika servers (dock ej nödvändigtvis fysiskt skilda datorer). Dessa är RoboInst-server, Mjukvaruserver, Bootserver, Konfigurationsserver.

[20] På engelska, hostname.
[21] Försvenskat översatt till vidarebefordrande server.

6 Nätverkstjänster

6.1 Apache som webbserver

Apache är den webbserver som standardmässigt medföljer Irix. Den version som medföljer är `sgi_apache` och medföljer på `irix-6.5-applications`-cdn. Apache kan också nås via freeware-cdn och heter då `fw_apache`.

Efter att ha installerat Apacheservern startas servern upp genom chkconfig `sgi_apache` on följt av `/etc/init.d/sgi_apache start`. Processen som skapas, och körs som `root`, heter `httpd` och har endast till uppgift att skapa childprocesser körandes som `nobody`. Observera att webbservern inte kommer att gå igång om Netscape fasttrack redan körs. Netscape Fasttrack stängs av med `chkconfig nss_fasttrack off`. Se dessutom till att inte både `sgi_apache` och `fw_apache` rullar samtidigt då de båda standardmässigt vill ha åtkomst till port 80.

Som standard konfigureras webbservern via `httpd.conf` i `/var/sgi_apache/httpd-outbox/`. Denna katalog kan också anses vara `root`-katalogen för webbservern. Själv webbsidorna ligger sedan standardmässigt i katalogen `/var/www/htdocs`. Denna kan konfigureras till annan valfri katalog genom konfigurationsfilen ovan.

För att tillåta användare att ha egna webbsidor på server konfigureras denna med följande alternativ

```
<IfModule mod_userdir.c>
        UserDir public_html
</IfModule>

<Directory /usr/people/*/public_html>
        AllowOverride FileInfo AuthConfig Limit
        Options Indexes Includes FollowSymLinks MultiViews
</Directory>
```

Hemmakatalog blir sedan `~username/public_html` och mappas i webbservern till `http://dns/~username`.

6.2 Samba för fildelning med Microsoft Windows

Samba är ett mjukvarupaket som körs under flera Unix- och Linuxdialekter, däribland även Irix. Samba möjliggör att dessa datorer kan prata med Microsofts protokoll för fil- och skrivardelning. Samba använder Server Message Block-protokollet (SMB) för sin kommunikation. SMB bygger på Network Basic Input Output System (NetBIOS) och går numera över TCP/IP.

För att Samba skall fungera skall Swat konfigureras i `/etc/inetd.conf` genom raden `swat stream tcp nowait root /usr/samba/bin/swat swat`. Starta sedan om inetd genom `killall -HUP inetd` för att aktivera. Konfigurera sedan Samba via webbgränssnittet som nås på port 901 på servern (`http://localhost:901/`). För att detta skall fungera måste `root`-användaren ha ett lösenord. För säkerhetsskull bör konfigurationsfilen säkerhetskopieras innan konfiguration sker (`cp /usr/samba/lib/smb.conf /usr/samba/lib/smb.conf.original`). Aktivera därefter Samba med `chkconfig samba on` och `/etc/init.d/samba start`.

Accesskontrollen i Samba sköts på vanligt vis via `/etc/passwd` och de vanliga accessrättigheterna.

7 Slutsatser

Stridssimulering, FLSCs datorpark består i dagsläget av mer än bara Irixdatorer. Idag finns även ett stort antal Windowsdatorer samt även ett betydande antal Linuxbaserade plattformar. Vad gäller Irix- tillsammans med Linuxdatorerna torde detta inte vara något som helst problem, dock ger Windowsdatorerna lite huvudbry.

Vad gäller valet av Internetprotokoll ses ingen betydande fördel med att byta från IPv4 till IPv6 i dagsläget. Detta då det nya protokollet inte är buggfritt på Irixmiljön och att IPv6 inte är testat med simulatorn. För FLSCs del ses inga fördelar med det nya protokollet i dagsläget.

MTU-paketens storlek sätts av begränsningar i simulatorn och bör styras av detta även i fortsättningen. Dock bör ett övervägande om att sätta `tcp_mtudisc on` göras då kommunikationen med andra system än Irix blir felaktig om inte mtu-storleken anpassas i kommunikationen. Kommer fler än ett nätverkskort anslutas i en dator, vilket inte verkar bli fallet initialt, bör `ipforwarding` slås av då detta förmodligen endast kommer störa nätverkstrafiken.

För att öka säkerheten i nätverket bör `tcpiss_md5`-flaggan slås på. Detta främst för att tcpiss_md5 ger den extra funktionen av att slumpvis vald port kommer användas. Detta minskar risken för snooping betydligt. Dessutom bör `rsh`, `Telnet` respektive `rlogin` bytas mot säkrare `ssh` i den mån detta är möjligt. En översyn av vilka processer som rullar på en viss dator bör tas. Samtliga tjänster som ej behövs bör stängas av, däribland tjänsten fingerd. `/etc/passwd` bör ”skuggas” genom användning av `/etc/shadow`.

För att lösenord ej skall skickas över nätverket i onödan skall de datorer och användare som används oftast ges access utan lösenord med `.rhosts` och i vissa fall via `.rlogin`. Användarna bör naturligtvis vara medvetna om vilka säkerhetsrisker detta kan medföra vid felaktigt användande.

Eventuellt bör en den inbyggda tcp-wrappern användas för att få kontroll över vilka gränssnitt som får accessa respektive port i `inetd`.

Vad gäller nätverksstyrd konfiguration kvarstår fortfarande ett antal oklarheter om vad Windowsdatorerna kan hantera. I nuläget är förslaget att NIS används även i fortsättningen på Irix och Linux för hanteringen av användare och lösenord. För översättning mellan ip-nummer och domännamn används en namnserver på Irixsystemet som hanterar samtliga datorers, inklusive Irixdatorerna, domännamn.

RoboInst för nätverksstyrd installation anses vara för komplicerat för FLSCs del och en enklare metod rekommenderas, förslagsvis en enkel diskkopia.

8 Ordlista och förkortningar

Broadcast	Ett sätt att kommunisera med alla på det lokala nätverket.
Cache/Caching	Att temporärt lagra information för att slippa hämta det nästa gång den behöver nås eller efterfrågas.
DHCP	Dynamic host configuration protocol
DNS	Domain Name Service
Forwarding	Vidarebefordrande
Header	I datapaketsammanhang den del av paketet som inte är egentlig data utan tilläggsinformation som behövs för att till exempel dirigera paketet rätt.
hostname	På svenska värdnamn, dock används ordet ofta i sin engelska form inom datorområdet.
IMAP	Internet Message Access Protocol, används för att läsning av e-post. Breven ligger normalt kvar på servern. Se även POP.
Interface	På svenska gränssnitt, dock används ordet ofta i sin engelska form inom datorområdet.
ISO	International Organization for Standardization
ITU-T	International Telecommunication Union- Telecommunication Standardization Sector
LDAP	Lightweight Directory Access Protocol
Localhost	Benämning på den lokala datorn.
Loopback	I nätverkssammanhang syftas återkoppling till den egna datorn. Det vill säga ingen kommunikation via nätverksgränssnittet. Vanligen använd för localhost.
MTU	Maximum transmission unit
Multicast	Ett sätt att ett paket till flera datorer samtidigt. Detta genom att grupper med medlemmar tilldelas och sedan sänds paketet till denna grupp.
Namnserver	Se DNS.
NetBIOS	Network basic input output system
NFS	Network File System
NIS	Network Information Service
OSI	Open System Interconnection
POP3	Post Office Protocol, version 3. Ett enkelt protokoll för att hämta e-post från sin e-postserver. Användarnamn och lösenord skickas okrypterat. Se även IMAP.
Port scanning	Att kolla av portar på en dator för att se vilka portar som är öppna och vad de används till.
RSH	Remote Shell, okrypterat shell. Se även SSH.
SMB	Server Message Block
Snooping	Avlyssning av nätverkstrafik.
SSH	Secure Shell, krypterad shelltrafik, till skillnad mot till exempel rsh och Telnet.
Stanza	Benämning på en grupp i `bind` som används vid konfiguration av DNS-server.
Subverting	Att utnyttja säkerhetsluckor i nätverksprogram för att ta sig in i systemet.
TCP	Transportprotokoll där, till skillnad från UDP, en bestående koppling mellan sändare och mottagare finns. Alla paket i sändningen tar därför samma väg och kommer fram i rätt ordning. Omsändning sker vid förlorade paket. Denna metod är ofta långsammare än UDP men tillförlitligare.

TTL	Time-To-Live. Används för att sätta hur många ”hopp” ett paket skall ta innan det ”dör”. Ett hopp kan till exempel vara en router.
UDP	Transportprotokoll där, till skillnad från TCP, ingen bestående koppling mellan sändare och mottagare finns utan varje paket tar den snabbaste vägen just då. Detta ger ofta högre prestanda än TCP. Inga omsändningar av förlorade paket sker vilket gör mindre tillförlitlig än TCP.

9 Referenser

1. IRIX® Advanced Network Administration – Student Workbook (AINA-3.0-I6.5-S-SD-W) [2004]
2. IRIX® Advanced Network Administration – Laboratory Workbook (AINA-3.0-I6.5-S-SD-L) [2004]
3. Internet Protocol (IP) Multicast, http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ipmulti.htm [2005-02-08]