FAGPRØVE

Fredrik R. Sætre

UKE 5, 2003

GJENNOMFØRING

HORDALAND
FYLKESKOMMUNE

- Innholdsfortegnelse -

- INNHOLDSFORTEGNELSE - 3

GENERELT 4

DEL 1, OPPGAVE 1.1 4

FREMDRIFT 4

DEL 1, OPPGAVE 1.2 6

A. IDS 6
B. DDOS 6
C. SNIFFING 7
D. "STEALTH" SCANNING 7

DEL 2 8

FREMDRIFT 8

DEL 3 10

DEL 4 11

1. SCSI 11
2. USB 12
2. USB 12
3. FIREWIRE 13

DEL 5 14

Generelt

Disponering av tiden for fagprøven.

Oppgave	Tid
Planlegging	Mandag 09.00 – 14.10
Montering av Firewall	Mandag 14.10 – 15.05
Installerte MRTG	Mandag 15.15 – 16.30
Konfigurerte og testet MRTG	Tirsdag 08.00 – 10.00
Monterte Firewall	Tirsdag 10.00 – 12.00
Installerte Smoothwall GPL 1.0	Tirsdag 12.30 – 13.00
Testet Firewall	Tirsdag 13.00 – 14.00
Oppgave 3a	Onsdag 09.00 – 09.30
Oppgave 4	Onsdag 09.30 – 15.30
Tegning for oppgave 5	Onsdag 15.30 – 17.00
Oppgave 5	Torsdag 09.00 – 12.30

DEL 1, oppgave 1.1

Alle deler som trengs til installasjonen var hos kunden

Fremdrift

MANDAG:

- Klargjorde arbeidsplass
- Festet ESD utstyr til jord og kroppen vha. ESD lenke.
- Sjekket deler
    - Hovedkort : ok
    - Nettverkskort : ok
    - Prosessor: ok
    - Vifte: ok
    - Harddisk: ok
    - RAM: ok
    - CDR-ROM: ok
    - Floppy: ok
    - Kabinett: ok
- Undersøkte monteringsmulighetene på kabinettet. Noe keitete. Bakplaten måtte fjernes for å kunne montere hovedkortet pga. en vertikaltstilt strømforsyning som tok mye plass.
- Oppdaget at prosessorviften var for høy og ville derfor komme i konflikt med strømforsyning.
- Ny vifte bestilt.
- Monterte det jeg kunne: Hovedkort til bakplaten, Floppy, Harddisk og CDR-ROM i tilegnede brønner.
    - Prosessor ikke påmontert for at den ikke skulle være eksponert for skade.
    - Ingen kort installert pga. de må fjernes når ny vifte kommer.
    - RAM ikke montert for å minske eksponering.
- Festet deksler for å gi minst mulig eksponering.

TIRSDAG:

- Mottatt 1 stk. Global Win CPU Cooler.
- Tok på meg ESD-lenke.
- Åpnet maskinen.
- Monterte prosessor og prosessorvifte. Passet på at kjernen på AMD prosessoren var intakt og uskadd. Fjernet beskyttelses tape for kjøle pasta og monterte viften på toppen.
- Installerte RAM i DIMM 1 slot. Passet på å ikke ta på connectorer.
- Festet bakplate med hovedkort og skrudde på deksel.

- Festet strøm til hovedkortet samt floppy, harddisk og CDR. Fordelte last jevnt over de forskjellige kabel strekkene.
- Satt i flatkabler til floppy og harddisk og CDR. Passet på at kablene ble satt riktig i på hovedkortet ved å konsultere manual og på enhetene ved å passe på å få merket side av kabelen mot strøm kontakten.
- Monterte nettverkskort og tilleggsporter for hovedkortet (Digital audio, USB og COM2).
- Koblet kontrollpanelet
- Klargjorde skjerm, tastatur og mus
- Startet maskin og åpnet BIOS settings.
- Maskinen fant automatisk harddisken og CDR spilleren.
- Slo av uønskede features på hovedkort og rettet klokken samt satte boot sequence til Floppy, CD, HD og Nettverk.
- Skrudde igjen kabinett

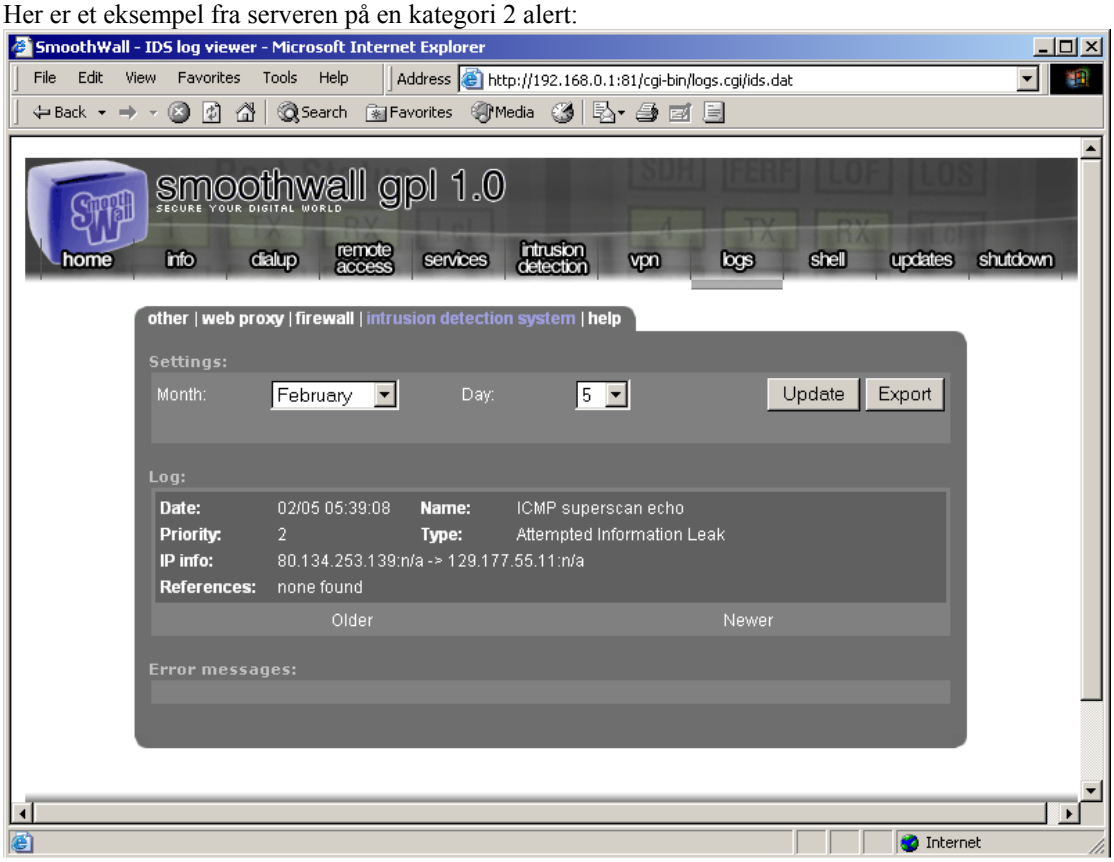
- Startet installasjonen av Smoothwall ved å boote fra installasjons CDen.
- Valgte:
    - Språk: Engelsk
    - Media: CD
      Maskinen partisjonerte og formaterte harddisken
    - Probet etter nettverkskort for grønn sone:
      Fant: 3-Com "Corkscrew" Etherlink PCI III/XL
    - IP: 192.168.0.1
    - Nettverksmaske: 255.255.255.0
    - Keyboard: "no" for norsk.
    - Timezone: Europa/Oslo
    - Hostname: Smoothwall
    - Disable ISDN
    - Disable USB ADSL
      Maskinen foreslo at jeg skulle ha en Grønn – Rød (Modem) konfigurasjon.
    - Network type: Green – Red
    - Probet etter nettverkskort for rød sone:
      Fant: 3-Com "Corkscrew" Etherlink PCI III/XL
    - Konfigurerte rød sone:
        - IP: 129.177.55.11
        - Primary DNS: 129.177.55.2
        - Secondary DNS: 129.177.55.4
        - Gateway: 129.177.55.1
    - Konfigurerte DHCP:
        - Start adresse: 192.168.0.100
        - Stop adresse: 192.168.0.150
        - Primary DNS: 129.177.55.2
        - Secondary DNS: 129.177.55.4
        - Gateway: 192.168.0.1
        - Name Suffix: ifjf.uib.no
    - Valgte passord for admin, setup og root
- Rebootet
- Koblet til nettverk
- Pinget gateway fra firewallen og byttet på kabler til jeg fikk fant riktig kort til riktig sone.
    - Øvre kort: Grønn sone
    - Nedre kort: Rød sone
- Koblet til klient maskin. Denne virket fint og kunne kjøre IP-trafikk ut uten problemer. Testet med Explorer.
- Logget meg på webinterface og så igjennom at alt var opp og gikk.
- Prøvde å pinge utenfra uten å få svar fra server. Brukte Nmap for å scanne porter på maskinen. Resultatet var negativt ergo virker firewallen.

DEL 1, oppgave 1.2

a. IDS

IDS står for Intrusion Detection System. Det detekterer upassende, uriktig eller abnormal aktivitet på en firewall, switch, router og lignende enheter. Firewallen som jeg har satt opp har en IDS som heter ”Snort”. Denne vil detektere uønsket aktivitet som for eksempel portscanning (brukes av hackere for å finne svakheter på serverere) eller uønsket aktivitet innenfor serveren som for eksempel MSN chat ol. En IDS kan både håndtere angrep utenfra og missbruk av nettet innenfra og rapportere om dette. På Snort kommer dette opp med forskjellige koder etter grad av prioritering. De går fra 1-3. Et DoS angrep ville blitt regisrert som 1, mens f.eks. MSN chat vil bli regisrert som en 3’er.

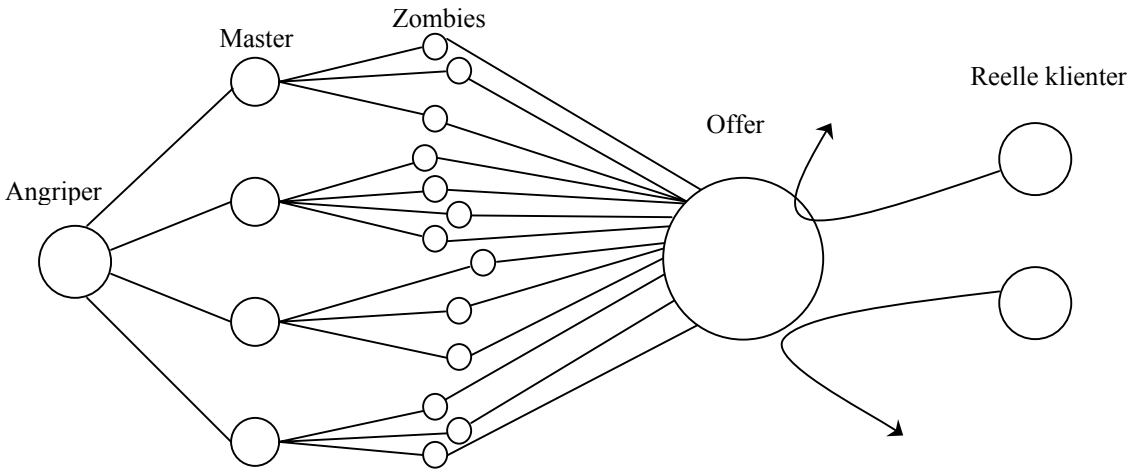
Her er et eksempel fra serveren på en kategori 2 alert:



Grunnen til at det ikke blir en klasse 1 er at Smoothwall har tatt høyde for dette og vil ikke bli påvirket. Like fullt er det regnet som et angrep, og kunne lett ha funnet seg veien inn hvis de hadde funnet hull i brannmuren. Denne forespørselen ble stoppet.

b. DDoS

DDoS står for Distributed Denial of Service. Dette er en teknikk som brukes for å sette ut tjenester på en maskin som for eksempel en webserver. Et ”denial of service” kan være at en som sitter på en 10 Mb linje setter en webserver som står på en 1 Mb linje ut av drift ved å overlaste serveren. På denne måten vil serveren måtte avvise en del legitime forespørsler. Serveren er ikke ”ute av drift”, men for opptatt til å svare. Når et DDoS angrep iverksettes er det flere maskiner som utfører en oppgave som til sammen har større båndbredde enn offeret.



Strukturen i et DDoS angrep noe forenklet

Et DDoS angrep gjøres i flere steg.

1. Scan et stort antall datamaskiner og bruk kjente svakheter for å få tilgang til disse maskinene.
2. De første maskinene blir kjent som ”master”-maskiner og brukes som et trinn for å lettere få flere maskiner under seg. Som i et pyramidespill.
3. Installer software på ”master”-klientene og få enda en undergruppe maskiner. Disse kalle ”Zombies”.
4. Når en har fått et ønsket antall maskiner kan man sette igang et angrep. Angriperen sier hvilken maskin som skal ”floodes” og resultatet blir at alle Zombiene vil gjøre som angriperen sier og dermed utfører kanskje tusner av maskiner samme operasjonen mot en server og på den måten oversvømmer offeret med forespørsler og dermed utestenger andre.

c. Sniffing

Sniffing er en passiv form for hacking og kan være farlig hvis passord og lignende blir overført ukryptert. Ved å ”lytte” på en eller flere porter på en ønsket maskin kan man fange opp all trafikk som går. Pakkene kan bli lest og disse kan inneholde informasjon om passord og brukernavn.
Når en ev. hacker får tak i disse er det ikke vanskelig å logge seg på serveren ved hjelp av informasjonen som ble mottatt og bruke den videre for å få tak i et stort nettverk maskiner for muligens å kjøre et stort DoS angrep eller få tilgang til sensitiv informasjon. Det er derfor viktig å alltid koble seg til med kypterte koblinger slik at data ikke kan bli lest åpent.
Det finnes en rekke software som gjør dette. Du kan for eksempel be programmet om å sniffe en hvilken som helst maskin og bare returnere passord og brukernavn.
Mailkontoer er ofte sårbare for slike angrep. Overføring skjer ofte ukryptert og passord og brukernavn kan lett bli snappet opp og missbrukt.
Sniffing begrenser seg til locale nettverk. En maskin må kunne fysisk være koblet til en hub eller et knutepunkt for å kunne lese data.

d. ”Stealth” scanning

Scanning er brukt for å finne hvilke porter som er åpen og hvilke tjenester som kjøres på en maskin. Dette kan gjøres av systemadministratorer for å se hvilke tjenester som kjøres og hvilke som kan være sårbare eller av hackere som bruker det for å finne svakheter i systemer og bryte seg inn på maskiner. Når man kjører en åpen scan til en maskin vil avsender også bli sendt med, og hvis man blir oppdaget er det ikke vanskelig å finne tilbake til hvem som scannet maskinen.
Når en scan gjennomføres vil maskinen starte en prosedyre som er kjent som ”handshake”. Til dette brukes SYN og ACK flaggene. Når all data er overført sendes et FIN signal, og det er FIN signalet som blir brukt for å kjøre en stealth scan. Istedet for å kjøre en handshake sender man en pakke med FIN flagget satt til en ønsket port. Hvis tjenesten på denne porten er aktiv vil den ikke svare. Hvis ikke vil den sende en feilmelding til avsender. Disse hendelsene blir ikke logget. ”Den som tier samtykker”. Når porten ikke svarer er tjenesten aktiv og på den måten kan man lett få en oversikt over hvilke tjenester som er aktiv og ikke.

DEL 2

Til denne oppgaven trenger jeg en maskin med Linux OS installert. Jeg velger å bruke en maskin med Red Hat 7.3 installert pga. denne er den mest stabile Red Hat distribusjonen. SNMP er en del av Red Hat pakken. Hvis ikke siste versjon er installert må denne hentes fra nettet og installeres.
Maskinen bør også ha en webserver i drift hvis ikke må dette installeres og konfigureres. SNMP skal brukes for å få informasjon om systemet og webserveren skal brukes for å få vist resultatet. MRTG er et program som kan overvåke og distibuere webdokumentasjon i form av grafer om alle funksjoner i en datamaskin. Alt fra CPU-temperatur til hvor mange brukere som er logget på. Jeg skal bruke det til å overvåke nettverksaktiviteten på en Linux server.

Fremdrift

- Fant en maskin med Red Hat 8.0 som er siste versjon av Red Hat. Jeg er kjent med at denne har litt bugs og ble sluppet litt tidlig, men en av mine kollegaer sier at han har gjort en lignende installasjon på 8.0 og lyktes. Jeg velger å prøve.
- Sjekket om maskinen hadde webserver og SNMP installert. Det hadde den.
- Testet SNMP
    - Gjorde backup av snmpd.conf og laget en ny.
    - Tilførte: "rocommunity test"
    - Startet SNMP daemon på nytt.
    - Kjørte en sjekk rutine for SNMP med kommandoen: "snmpwalk –V 1 –c test localhost system"

```
[root@seis snmp]# snmpwalk -v 1 -c fredrik localhost system
SNMPv2-MIB::sysDescr.0 = STRING: Linux seis 2.4.18-14 #1 Wed Sep 4 11:57:57 EDT 2002 i586
SNMPv2-MIB::sysObjectID.0 = OID: NET-SNMP-MIB::netSnmpAgentOIDs
SNMPv2-MIB::sysUpTime.0 = Timeticks: (392) 0:00:03.92
SNMPv2-MIB::sysContact.0 = STRING: root@localhost
SNMPv2-MIB::sysName.0 = STRING: seis
SNMPv2-MIB::sysLocation.0 = STRING: Unknown
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (20) 0:00:00.20
SNMPv2-MIB::sysORID.1 = OID: IF-MIB::ifMIB
SNMPv2-MIB::sysORID.2 = OID: SNMPv2-MIB::snmpMIB
SNMPv2-MIB::sysORID.3 = OID: TCP-MIB::tcpMIB
SNMPv2-MIB::sysORID.4 = OID: IP-MIB::ip
SNMPv2-MIB::sysORID.5 = OID: UDP-MIB::udpMIB
SNMPv2-MIB::sysORID.6 = OID: SNMP-VIEW-BASED-ACM-MIB::vacmBasicGroup
SNMPv2-MIB::sysORID.7 = OID: SNMP-FRAMEWORK-MIB::snmpFrameworkMIBCompliance
SNMPv2-MIB::sysORID.8 = OID: SNMP-MPD-MIB::snmpMPDCompliance
SNMPv2-MIB::sysORID.9 = OID: SNMP-USER-BASED-SM-MIB::usmMIBCompliance
SNMPv2-MIB::sysORDescr.1 = STRING: The MIB module to describe generic objects for network interface sub-layers
SNMPv2-MIB::sysORDescr.2 = STRING: The MIB module for SNMPv2 entities
SNMPv2-MIB::sysORDescr.3 = STRING: The MIB module for managing TCP implementations
SNMPv2-MIB::sysORDescr.4 = STRING: The MIB module for managing IP and ICMP implementations
SNMPv2-MIB::sysORDescr.5 = STRING: The MIB module for managing UDP implementations
SNMPv2-MIB::sysORDescr.6 = STRING: View-based Access Control Model for SNMP.
SNMPv2-MIB::sysORDescr.7 = STRING: The SNMP Management Architecture MIB.
SNMPv2-MIB::sysORDescr.8 = STRING: The MIB for Message Processing and Dispatching.
SNMPv2-MIB::sysORDescr.9 = STRING: The management information definitions for the SNMP User-based Security Model.
SNMPv2-MIB::sysORUpTime.1 = Timeticks: (4) 0:00:00.04
SNMPv2-MIB::sysORUpTime.2 = Timeticks: (7) 0:00:00.07
SNMPv2-MIB::sysORUpTime.3 = Timeticks: (8) 0:00:00.08
SNMPv2-MIB::sysORUpTime.4 = Timeticks: (9) 0:00:00.09
SNMPv2-MIB::sysORUpTime.5 = Timeticks: (10) 0:00:00.10
SNMPv2-MIB::sysORUpTime.6 = Timeticks: (12) 0:00:00.12
SNMPv2-MIB::sysORUpTime.7 = Timeticks: (19) 0:00:00.19
SNMPv2-MIB::sysORUpTime.8 = Timeticks: (20) 0:00:00.20
SNMPv2-MIB::sysORUpTime.9 = Timeticks: (20) 0:00:00.20
[root@seis snmp]#
```

    - Testen viser at jeg kan hente ut system informasjon fra SNMP serveren. Testen var godkjent.

- La leserettigheter til for localhost i snmpd.conf.

```
com2sec local     localhost   local
```

  - Gir sikkerhetsgodkjenning for localhost identifisert av "local".

```
group MyROGroup  v1        local
group MyROGroup  v2c       local
group MyROGroup  usm       local
```

  - Lager grupper i de forskjellige versjonene og legger til sikkerhetsgodkjenningen "local".

```
view all    included  .1                      80
```

  - Gir muligheter for å lese alle undergruppene

```
access MyROGroup ""      any       noauth    exact     all    none   none
```

  - Gir "MyROGroup" tilgang å lese alle subgruppene.

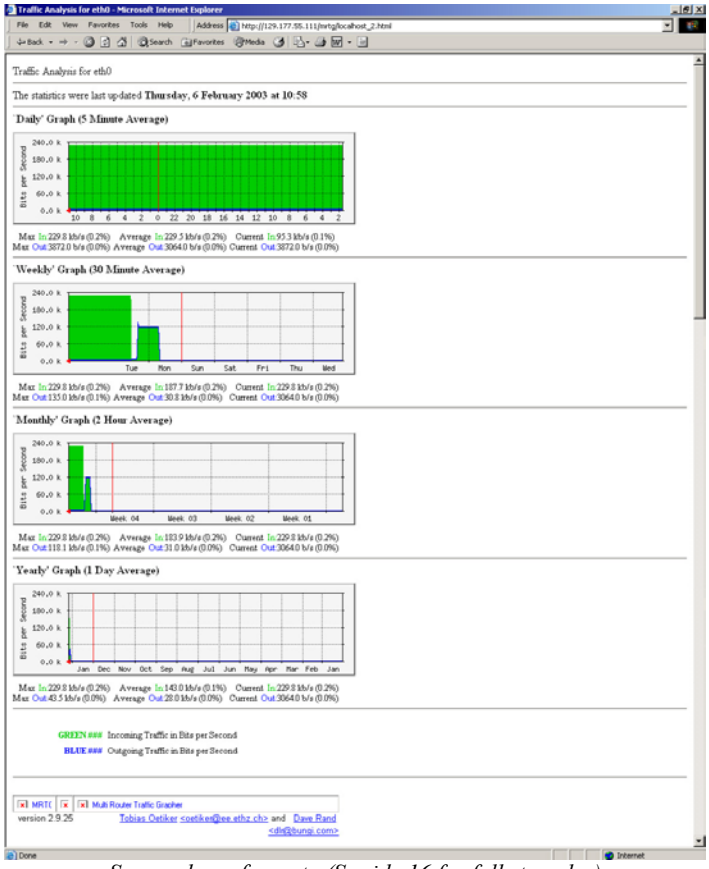
- La til script i mrtg.cfg i web mappen. Denne config filen hentet informasjon om IP trafikk.

```
WorkDir: /var/www/html/mrtg/
ThreshDir: /var/www/html/mrtg/
Options[^]: noinfo,

Target[index]: 2:local@localhost:
SetEnv[index]: MRTG_INT_IP="192.168.0.109"
MRTG_INT_DESCR="eth0"
MaxBytes[index]: 12500000
Title[index]: Traffic Analysis for eth0
PageTop[index]: Traffic Analysis for eth0
Options[index]: bits
```

  - WorkDir — Mappen som reultatet skrives til
  - ThreshDir — Midlertidig arbeidmappe
  - Options[^]: noinfo, — Gir ikke informasjon om tid oppe og navne på enheten
  - Target — Kilden informasjon skal hentes fra
  - MaxBytes — Maximum størrelse på bytes på grafen
  - Title — Titel for siden. Kommer i <title> tags.
  - PageTop — Kommer som header på siden.
  - Options:bits — Anngir måleenhet

- Lastet ned, pakket ut og installerte siste versjon av MRTG.
  - MRTG finnes på www.mrtg.org.
  - Herfra lastet jeg ned siste versjon.
  - Pakket ut med untar: "tar –xzvf mrtg------.tar.gz"
  - Gikk inn i mappen og kjørte configure scriptet med "-prefix=/usr/local/mrtg-2"
  - Kjørte make og make install.
- La til stien til mrtg programmet i PATH ved legge til "PATH=$PATH:/usr/local/mrtg-2/bin" i /etc/profile.
- Restartet snmpd.
  #service snmpd restart
- Kjørte MRTG på mrtg.cfg filen som ble laget.
- Kontrollerte websiden som ble generert. Data var registrert og alle grafer ble tegnet.



Screen dump fra mrtg (Se side 16 for full størrelse)

- Redigerte crontab og la til mrtg i rotasjonen slik at siden blir automatisk oppdatert.
  *0-59/5 * * * * mrtg /var/www/html/mrtg/mrtg.cfg*

DEL 3

Liste over komponenter i Firewall.

Komponent	Beskrivelse	Merknad
Kabinett	Q-Tec Midi tower ATX 6021CMD	
Hovedkort	Asus A7N266-VM Hovedkort Socket A	
Prosessor	AMD Athlon XP2100+ 1.733 GHz	
Vifte	GlobalWin CPU-Vifte Socket A/7/370/FCPGA	
RAM	DDR-DIMM PC2100 256MB DDR CL2.5	
Nettverkskort	3Com Etherlink XL 10/100Mbit PCI	2 stk
CDR	Plextor CD-brenner IDE 48x/24x/48x	

Oppgave 3b og 3c er beskrevet under fremgangs delene på oppgave 1 og 2.

DEL 4

1. SCSI

Eksterne SCSI-plugger

SCSI står for Small Computer Standard Interface og har eksistert siden 1986. De har kommet i mange variasjoner og størrelser og er hovedsakelig brukt som et bedre alternativ til IDE. Både interne og eksterne media kan tilkobles og det er mest lagringsmediaer som SCSI blir brukt til. De tidligere har støtte for 8 enheter, men i realiteten bare 7 fordi busskotrolleren tar opp en av adressene. De nye har plass til 16(15). Bussen er ”Daisy-chained” og må være terminert i begge ender.

Det er to forskjellige buss typer.

- Singel-ended SCSI
- Differential SCSI
    - Low Voltage Differential
    - High Voltage Differential

Disse to typene er forskjellig og en kan ikke bruke en singel-ended enhet på en differential buss uten en konverter eller lignende.
Differential koblet SCSI har større støy immunitet spessielt når den er brukt med ”twisted pair”-kabel.
En singel-ended SCSI-buss kan ha en lengde på mindre enn seks meter, mens en HVD differential buss kan være opptil 24 meter(12 for LVD). Det er ingen softwaremessig forskjell på de to bussene, kun fysiske forskjeller på disse to.
I forhold til mange andre busser har SCSI ”hot-swap” mulighet. Dette betyr at du kan fjerne og legge til haddisker mens maskinen fremdeles går.

Her er en oversikt over de forskjellige SCSI versjonene: *(kilde: http://www.scsita.org)*

STA Terms	Bus Speed, MBytes/Sec. Max.	Bus Width, bits	Max. Bus Lengths, Meters (1)			Max. Device Support
			Single- ended	LVD	HVD	
SCSI 1 (2)	5	8	6	(3)	25	8
Fast SCSI (2)	10	8	3	(3)	25	8
Fast Wide SCSI	20	16	3	(3)	25	16
Ultra SCSI (2)	20	8	1.5	(3)	25	8
Ultra SCSI (2)	20	8	3	-	-	4
Wide Ultra SCSI	40	16	-	(3)	25	16
Wide Ultra SCSI	40	16	1.5	-	-	8
Wide Ultra SCSI	40	16	3	-	-	4
Ultra2 SCSI (2,4)	40	8	(4)	12	25	8
Wide Ultra2 SCSI (4)	80	16	(4)	12	25	16
Ultra3 SCSI or Ultra160 SCSI (6)	160	16	(4)	12	(5)	16
Ultra320 SCSI (6)	320	16	(4)	12	(5)	16
Ultra640	640	16	(4)	(7)	(5)	16

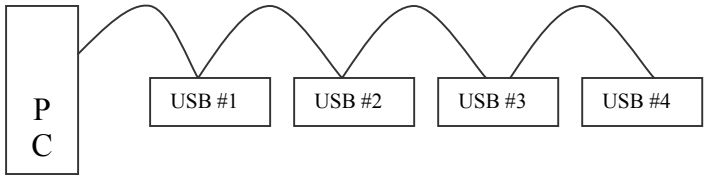
(1) The listed maximum bus lengths may be exceeded in Point-to-Point and engineered applications.
(2) Use of the word "Narrow", preceding SCSI, Ultra SCSI, or Ultra2 SCSI is optional.
(3) LVD was not defined in the original SCSI standards for this speed. If all devices on the bus support LVD, then 12-meters operation is possible at this speed. However, if any device on the bus is singled-ended only, then the entire bus switches to single-ended mode and the distances in the single-ended column apply.
(4) Single-ended is not defined for speeds beyond Ultra.
(5) HVD (Differential) is not defined for speeds beyond Ultra2.
(6) After Ultra2 all new speeds are wide only.

2. USB

USB står for Universal Serial Bus og er en av de mest brukervennlige bussene idag. Alle bærbare maskiner har eller bør ha en eller flere USB porter. USB er en veldig fleksibel buss og er brukt til alt fra digitale kamera, scannere, mus, tastatur til lagringsmedia, lydkort, DSL-modemer og PC-telefoner. Den kan støtte 127 enheter på en gang per buss, selv om den ville blitt veldig treg da. Hovedkortene idag kommer stort sett med 2 USB busser som standard og har ofte utvidelser til både fire og seks porter i tillegg.

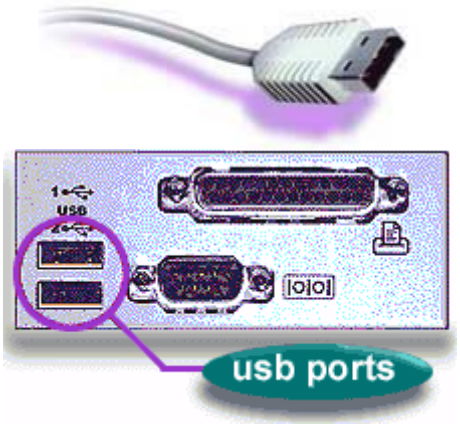
Båndbredden på vanlig USB ligger på 12 Mb/sec. USB2 derimot har hele 480 Mb/sec å sparke fra med og kommer godt opp på listen sammen med IEEE-1394(FireWire) og senere SCSI versjoner. En USB kabel kan max være 5 meter.

USB har også muligheten for ”hot-swapping” og kan ”daisy-chain” kobles slik som SCSI. Windows er meget kompatibelt med USB og installerer ofte drivere av seg selv når det har detektert en kjent enhet. USB har ikke bruk for terminering slik SCSI har.



Daisy-chaining

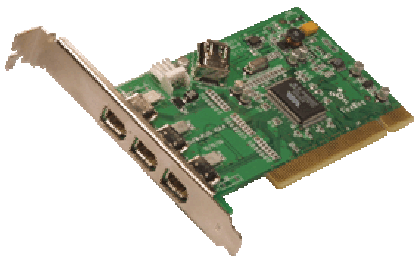
USB har en rekke adaptere for å kunne være kompatibel med andre interfacer. Jeg kan blandt annet nevne at det finnes adaptere til EIDE og SCSI-2. USB og IEEE-1394(FireWire) er komplimentære busser og gjør samme nytte. USB er billigere og enklere teknologi enn FireWire. USB er beregnet på applikasjoner med medium til lav-båndbredde.



Dette er USB porten og pluggen

3. FireWire

IEEE 1394 er en høyhastighetsbus for overføring av store mengder data. Det var Apple som gav standarden døpenavnet FireWire. Apple er også kjent som storbruker av standarden og får mye skryt for deres gode kompabilitet med standarden.
FireWire kan brukes til Video overføring, printere, scannere, eksterne lagringsmedia og mye mye mer. Personlig har jeg vært borti systemet i bruk som en del av harddisk-recording systemet i studioet vårt og er imponert over datamengden og kvaliteten på dataen som overføres. Lengden per strekk mellom hver enhet kan ikke overskride 4,5 meter, men kan være lengre på lavere hastigheter. Ved å bruke en fiber variant kan lengder komme opp i 70 meter.



Firewire kort for PCI

Standard Firewire kabel

De tidligere FireWire bussene gikk på hastigheter fra 98 Mb/sec til 400 Mb/sec, mens den nye 1394b går på svimlende 1,2 Gb/sec. Den har også det som heter isochronous dataoverføring som garanterer at et minimum med data blir overført. Dette er ypperlig for live streaming av video og audio.

Man trenger ikke PC eller Macintosh for å bruke FireWire. Det brukes også for kommunikasjon og dataoverføring mellom perifere enheter som f.eks. en videomaskin og et film kamera.

Det er ikke så farlig hvordan man kobler til enheter på en 1394 bus. Det er det samme, så lenge alt holder sammen. Daisy-chaining, tre- eller stjernekoblinger. Alt går.

Det finners to typer FireWire connectorer en 4 pinns og en 6 pinns. 4 pinns connectoren er hovedsakelig brukt på batteridrevne enheter som for eksempel digitale-videokameraer og håndholdte audiorecordere.

DEL 5

Dette er en beskrivelse av tegningen på side 15. Den beskriver den xerografiske prosessen steg for steg.

1. **Main charge:**
   Fototrommelen blir ladet statisk opp ved hjelp av hoved-corona. Hoved-corona yter store spenninger, men lite strøm. Det kan finnes en lampe for å lade ut områder på trommelen som ikke skal brukes. Dette er ofte i maskiner som har støtte for A3. Og fototrommelen vil bli eksponert med lys på steder hvor det ikke vil bli kopiert.

2. **Belysning:**
   Fra en lampe blir originalen belyst gjennom glassplaten på toppen. Den beveger seg over originalen synkronisert med fototrommelen.

3. **Optikk:**
   Lyset går gjennom optikken med speil og linser. Den har som hovedoppgave å lede riktig størrelse på kopien til eksponering. Her kan kopier forstørres og forminskes eller bare være 1:1.

4. **Eksponering:**
   Når lyset har gått gjennom optikken kommer den til fototrommelen. Der hvor det lyses på trommelen vil spenningen forsvinne. Der hvor originalen er hvit vil det sendes 100% av lyset tilbake. Hvor det er blekk(sort) vil det sende tilbake tilnærmet lik 0% av lyset og dermed vil fotorullen på disse stedene fremdeles være ladet. Det lages da en ladet kopi av originalen på glassplaten på fototrommelen.

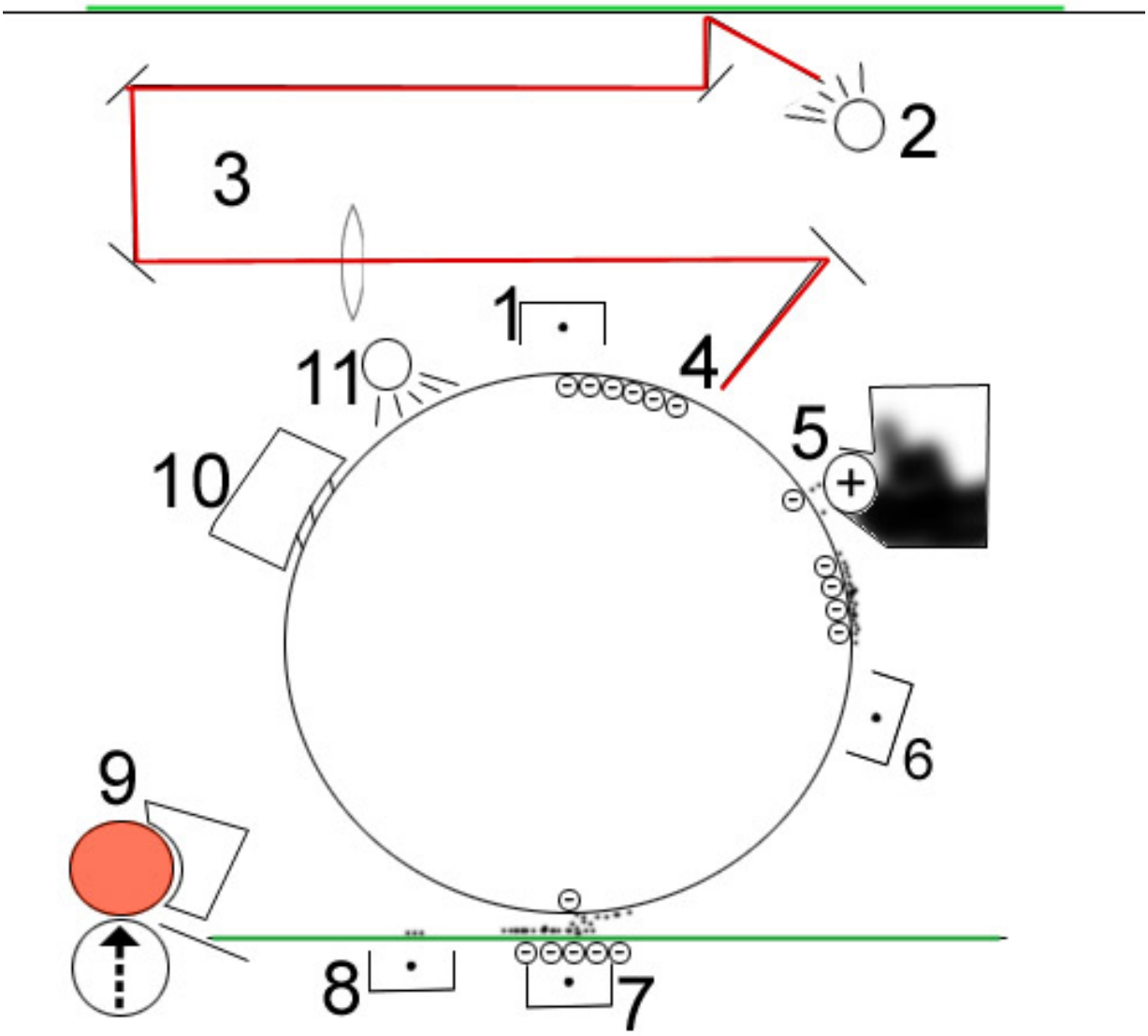
5. **Developer:**
   Dette er enheten som overfører og inneholder toner. Den er ladet med et annet potensiale enn trommelen og når toneren blir ladet vil den overføres til de stedene hvor fototrommelen er ladet. Det finnes forskjellige typer toner. Noen er magnetisk og legge seg som en børste på en magnetisk developer. Det kan også være en diaz i bunnen av trommelen for å ”riste” toneren ned til rolleren som overfører toneren til fototrommelen.

6. **Støtteprosess, decharge:**
   Denne støtteprosessen minsker potensialet i trommelen. Det vil si at den gjør ladingen på trommelen mindre slik at overføringen blir lettere.

7. **Overføring:**
   I dette steget blir toner overført til arket. Dette gjøres ved at en overførings-corona som har sterkere ladning enn trommelen har ”trekker” toneren ned fra undersiden av arket. Arket blir ladet.

8. **Diaz:**
   Dette er en vekselspenning for å lade ut arket. Slik at det ikke følger med trommelen eller andre deler.

9. **Fixing:**
   For å feste toner til arket brukes det trykk og varme. Trykkroller befinner seg på undersiden av arket og er fjæret og noe myk. Heatrolleren er hard, meget varm, består stort sett av teflon og befinner seg på samme side som toneren. Alltid på toppen og renses kontinuerlig. For at ikke toner og arket skal feste seg til trommelen brukes det enten voks, silikon eller lignende for å gjøre overflaten glatt. Det er denne enheten som gjør at maskinen har så lang oppstarts tid.
   For å gi heatrolleren sin varme brukes en lampe som ligger inni rolleren.



Den xerografiske prosess.

Traffic Analysis for eth0 - Microsoft Internet Explorer

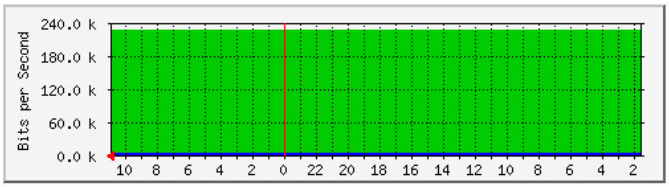
File Edit View Favorites Tools Help Address http://129.177.55.111/mrtg/localhost_2.html

Back Search Favorites Media

Traffic Analysis for eth0

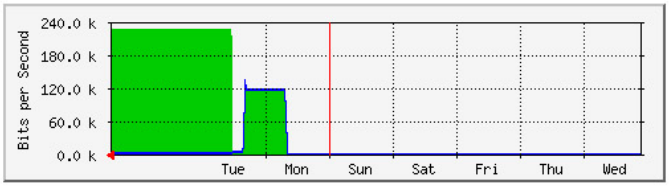
The statistics were last updated **Thursday, 6 February 2003 at 10:58**

`Daily' Graph (5 Minute Average)



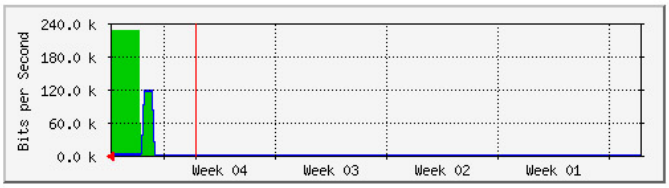
Max In:229.8 kb/s (0.2%) Average In:229.5 kb/s (0.2%) Current In:95.3 kb/s (0.1%)
Max Out:3872.0 b/s (0.0%) Average Out:3064.0 b/s (0.0%) Current Out:3872.0 b/s (0.0%)

`Weekly' Graph (30 Minute Average)



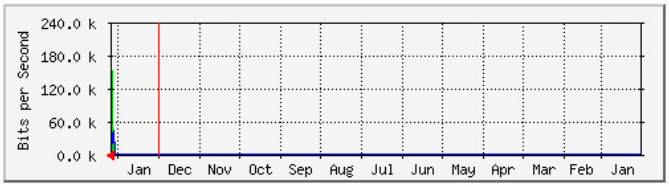
Max In:229.8 kb/s (0.2%) Average In:187.7 kb/s (0.2%) Current In:229.8 kb/s (0.2%)
Max Out:135.0 kb/s (0.1%) Average Out:30.8 kb/s (0.0%) Current Out:3064.0 b/s (0.0%)

`Monthly' Graph (2 Hour Average)



Max In:229.8 kb/s (0.2%) Average In:183.9 kb/s (0.2%) Current In:229.8 kb/s (0.2%)
Max Out:118.1 kb/s (0.1%) Average Out:31.0 kb/s (0.0%) Current Out:3064.0 b/s (0.0%)

`Yearly' Graph (1 Day Average)



Max In:229.8 kb/s (0.2%) Average In:143.0 kb/s (0.1%) Current In:229.8 kb/s (0.2%)
Max Out:43.5 kb/s (0.0%) Average Out:28.0 kb/s (0.0%) Current Out:3064.0 b/s (0.0%)

GREEN ### Incoming Traffic in Bits per Second
BLUE ### Outgoing Traffic in Bits per Second

MRTG Multi Router Traffic Grapher
version 2.9.25 Tobias Oetiker <oetiker@ee.ethz.ch> and Dave Rand <dlr@bungi.com>

Done Internet