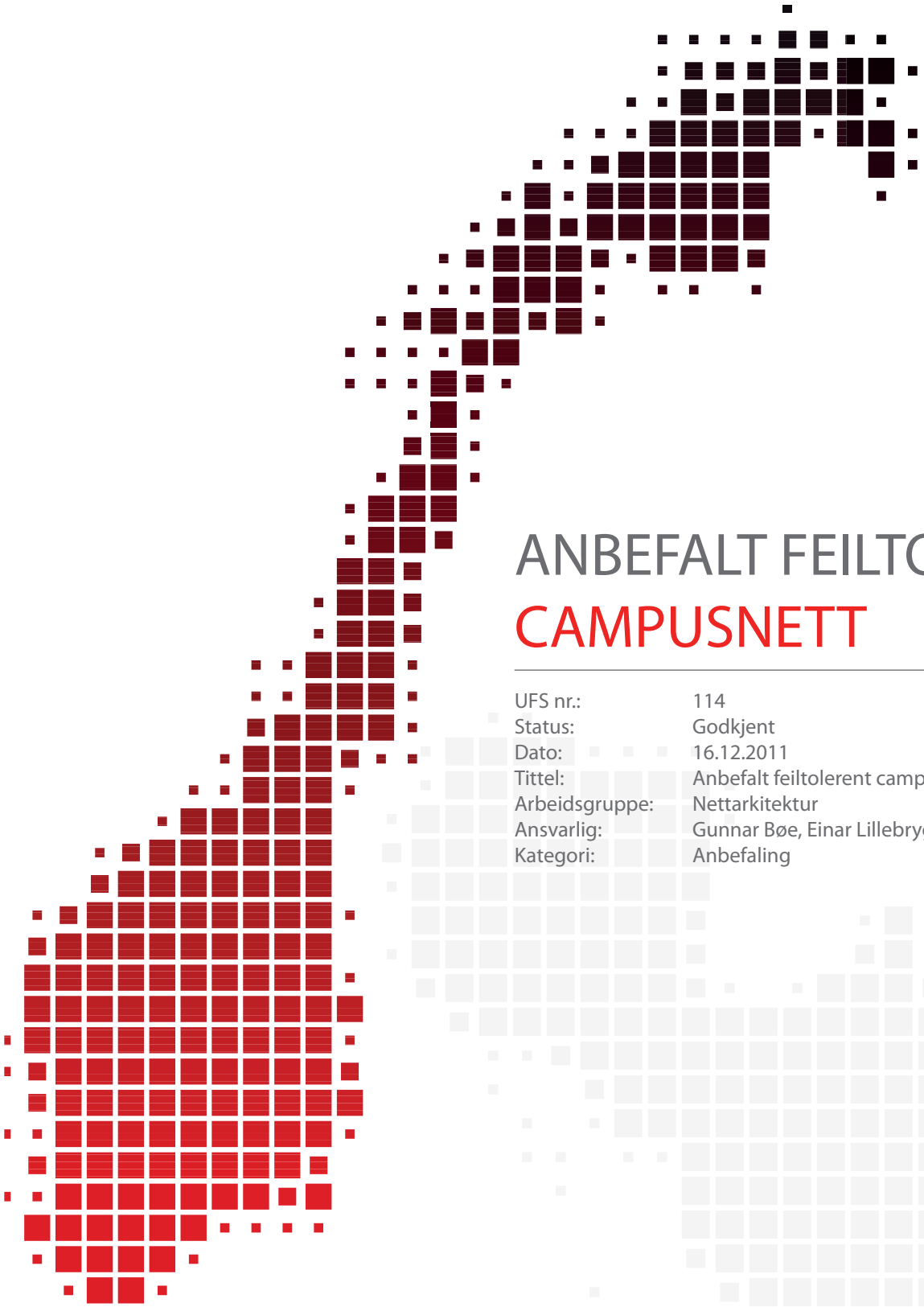
UNINETT Fagspesifikasjon



ANBEFALT FEILTOLERENT CAMPUSNETT

UFS nr.:	114
Status:	Godkjent
Dato:	16.12.2011
Tittel:	Anbefalt feiltolerent campusnett
Arbeidsgruppe:	Nettarkitektur
Ansvarlig:	Gunnar Bøe, Einar Lillebrygfjeld, Vidar Faltinsen
Kategori:	Anbefaling

Innhold

Sammendrag

Dette dokumentet anbefaler oppsett av et feiltolerent campusnett. Man kan dele campusnettet i tre deler; kjernenett, distribusjonsnett og aksessnett. Anbefalinger gis for hver del av campusnettet.

For kjernenettet anbefales en struktur med to hovedkommunikasjonsrom med godt adskilte strøm- og kjøleløsninger. Det anbefales videre en redundant fiberstruktur ut fra disse to hovedkommunikasjons-rommene. Kjernenettet bør bestå av (minst) to kjernesvitsjer som er satt opp med redundant BGP-tilkobling mot Forskningsnettet. Det bør videre være redundante forbindelser til distribusjons- og/eller kantsvitsjer. Man kan velge en tradisjonell løsning basert på IS-IS/OSPF, VRRP/HSRP og Rapid Spanning Tree (RSTP), eller man kan satse på en virtuell og proprietær kjerne med bruk av link aggregering (IEEE 802.3ad) mot distribusjon-/kantsvitsjer. Det er fordeler og ulemper med begge løsningene.

I distribusjonsnettet anbefales bruk av Rapid Spanning Tree (RSTP, IEEE 802.1w). Nyere og bedre protokoller er nå standardisert, slik som TRILL og IEEE 802.aq, men utstyrsstøtten er per dato mangelfull. Dersom man har mange vlan bør MSTP (IEEE 802.1s) vurderes, da dette gir gode muligheter for å lastdele trafikk i grupper av vlan (der hver gruppe kjører RSTP). Alternativt kan proprietære løsninger for per vlan spanning tree benyttes.

Strukturen i distribusjonsnettet kan realiseres på ulike måter. Anbefalingen går inn på fire ulike scenarioer og gjør rede for fordeler og ulemper i hvert tilfelle.

I aksessnettet er det viktig å konfigurere kantsvitsjporter som ”edge porter”. Dette gjør at av- og påkobling av endeutstyr ikke trigger nye spanning tree beregninger.

Tjenere bør kobles redundant inn i nettverket mot to ulike svitsjer (typisk såkalte top-of-rack svitsjer). En mye brukt og anbefalt løsning er å benytte ethernet bonding på linux-tjenere og ethernet teaming på windowstjenere. Disse bør settes opp i aktiv/aktiv modus slik at avsendertrafikk fra serverne benytter begge forbindelsene og man således vet at feiltoleransen virker. Et alternativ til bonding/ teaming er å benytte multichassis link aggregering, men dette vil kreve proprietære løsinger på svitsjesiden. For virtuelle tjenere tilkoblet svitsjer i bladsystem må man sørge for redundant tilkobling til bladsystemet.

1 Introduksjon

Dette dokumentet anbefaler oppsett av et feiltolerent campusnett. Vi kan i hovedsak dele campusnettet i tre lag:

- Kjernenett
- Distribusjonsnett
- Aksessnett

De ulike lagene behandles i de påfølgende kapitlene.

Dokumentet tar ikke stilling til kapasiteten på de ulike linkene i campusnettverket. Det er uavhengig av design og kan gradvis oppgraderes ved behov. Generelt kan vi si at nyimplementasjoner i dag bør støtte 10 Gbps ethernett i kjernenettet og 1 Gbps ut mot kanten i nettet.

Anbefalingen er forsøkt holdt generell og leverandøruavhengig, men det gis allikevel en del konkrete referanser til utstyr. Det skal bemerkes at leverandørreferansene ikke er komplette. Siden utstyr fra Cisco og HP er dominerende i UH-sektoren i dag er disse bredest omtalt.

2 Kjernenettet

I kjernenettet gjøres ruting, i større campusnett med en dynamisk rutingprotokoll som IS-IS eller OSPF. Redundant ruting fra campusnettet mot omverden gjøres med BGP. For mindre campusnett uten redundant kjernenett benyttes kun statisk ruting.

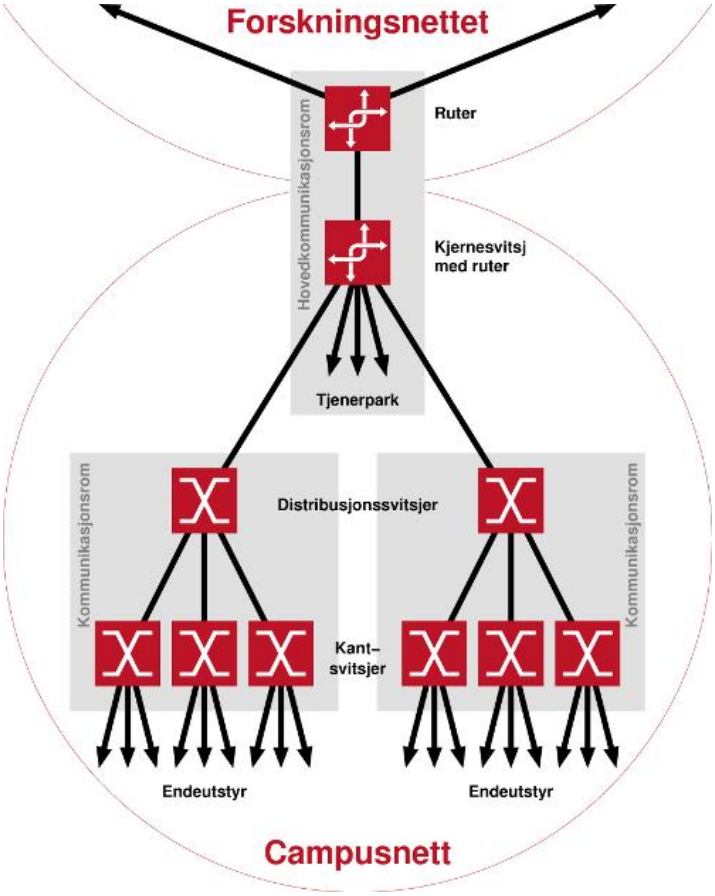
2.1 Ett hovedkommunikasjonsrom

Dette scenarioet er typisk for mange høgskoler i sektoren (de største universitetene og noen høgskoler har en mer redundant struktur på plass). Figur 1 viser typisk nettstruktur i dette tilfellet.

Det er her kun **ett** hovedkommunikasjonsrom på campus:

- Dette er UNINETTs leveransepunkt. I mange tilfeller er leveransen redundant, dvs det er to adskilte veier ut av campus. Feiltolerant Forskningsnett er viktig, men ikke fokus i denne anbefaling.
- Kjernesvitsjen til institusjonen står i dette rommet. Dette er en L3 svitsj som i tillegg til L2 svitsjing også gjør lokalnettruting og pakkefiltrering. Rutingoppsettet er enkelt basert på statisk ruting.
- I mange tilfeller er sentral tjenerpark direkte tilkoblet denne svitsjen. Svitsjen danner videre roten i en fibermessig trestruktur med fiberforbindelser ut til andre kommunikasjonsrom der distribusjon- og/eller kantsvitsjer er plassert.

En styrke med dette oppsettet er at det har **lav kompleksitet**. Det krever ikke spesialist-kompetanse på ruting m.m. for å drive løsningen.



Figur 1: Ikke-redundant campusnett

2.1.1 Svakheter

Ut i fra et feiltoleranseperspektiv gir løsningen mange svakheter. Ideelt sett bør man ikke ha noe sted med "single point of failure". Det er tilfelle her:

- Kun en kjernesvitsj. Dersom denne ryker/går ned er det meste av campusnettet nede.
- Kun en vei ut av et gitt kommunikasjonsrom. Ryker forbindelsen er miljøet som sokner til rommet nettløst.

2.1.2 Foreslåtte forbedringstiltak

Dersom ikke-redundant struktur beholdes kan man bøte på svakhetene med en del tiltak.

Fokuser særlig på strøm og kjøling

Erfaringsmessig er det problemer med strømforsyning eller kjølesystemer som skaper de fleste driftsproblemene i sektoren. Management modul i kjernerutere og svitsjer er i seg selv pålitelige og det er sjeldent at de svikter (men det kan skje). Prioriter derfor:

1. Sørg for redundant strømforsyning i kjernesvitsj. Potensielt også i andre svitsjer.
2. Sørg for redundant strømmating i hovedkommunikasjonsrom, med bruk av UPS og evt dieselaggregat. UFS 107 [1] gir anbefalinger her.
3. Sørg for feiltolerant kjøleløsning, jfr. UFS 108 [2].
4. Sørg for en velfungerende overvåkningsløsning, der utfall av sentrale komponenter gir øyeblikkelig varsel på SMS. Jfr. UFS 128 [3] for krav til overvåkning. Utover generell overvåkning av nettverksutstyret må følgende også dekkes:
    - Overvåk strøm- og kjølingssituasjonen. Ved redundant strømforsyning, der den ene forsyningen feiler, må dette varsles.
    - UPSene må støtte SNMP slik at man kan få umiddelbar alarm ved bortfall av bystrøm og således kan rekke å gjøre ekstraordinære tiltak.
    - Tilsvarende må en rask temperaturstigning i kommunikasjonsrommet varsles. Her anbefales bruk av Weathergoose [4] eller tilsvarende.

I tillegg, men dette er heldigvis mer sjeldent forekommende:

- Sørg for tidligdeteksjon i forbindelse med branntilløp, jfr. UFS 104 [5].

Hold reservedelslager

- Hold eget reservedelsutstyr av distribusjon og kantsvitsjer.
- Vurder også reservedeler for fiber og TP-moduler i kjernesvitsj. Ha i det minste alternative svitsjer som kan fases inn dersom en kjernesvitsjmodul feiler.
- Alternativt, eller som et supplement, benytt UNINETTs reservedelslager for komponenter i kjernesvitsj.
- Vurder evt særskilt avtale med leverandør om kort leveranse av reservedeler (dette blir en kostnadsvurdering).

Redundant management modul

Med tiltakene over er situasjonen fortsatt sårbar, særlig i forhold til om management modul i kjernesvitsjen går i stykker. Da vil i praksis hele campusnettverket gå ned og det kan ta tid å få tilsendt en erstatning. Følgende ekstratiltak kan bedre på dette:

- Anskaff en redundant management modul. Denne settes i drift som en hot standby.
- I tillegg ha en ekstra viftemodul på lager (av riktig type). Dersom viftemodul ryker går hele svitsjen ned.

Dersom replikering av tilstand mellom rutingprosessene på de to kortene er støttet, har man den ekstra fordelen at programvareoppgradering kan gjennomføres sømløst, uten nedetid. Når den ene management modulen oppgraderes er det andre i drift og motsatt.

Redundant kjernesvitsj i samme rom

Ytterligere redundans kan bygges ved å innføre en ekstra kjernesvitsj i samme rom. Man har fortsatt ikke alle de fordelene som to separate hovedkommunikasjonsrom gir, men kan ellers bygge et nettmessig fullgodt redundant kjernenett. Redundant kjernenett behandles videre i kapittel 2.2.

2.2 To hovedkommunikasjonsrom

Å ta steget opp til to hovedkommunikasjonsrom gir en vesentlig styrket feiltoleranse. Det muliggjør:

- To adskilte UNINETT-leveransepunkter med to adskilte UNINETT-rutere.
- To adskilte kjernesvitsjer med mulighet for redundant struktur til distribusjon-/og kantsvitsjer.

Viktige poeng med adskilte hovedkommunikasjonsrom:

- Lav sannsynlighet for strømutfall i begge rom samtidig. NB: Det ideelle her er at hvert rom er matet fra to forskjellige transformatorer.
- Lav sannsynlighet for at fiberskade tar ned nettet. Redundant fiberstruktur gjør at minst to adskilte brudd må til for å få utfall.
- Svært lav sannsynlighet for kjøleproblem i begge rom samtidig.
- Svært lav sannsynlighet for branntilløp eller andre katastrofer i begge rom samtidig.

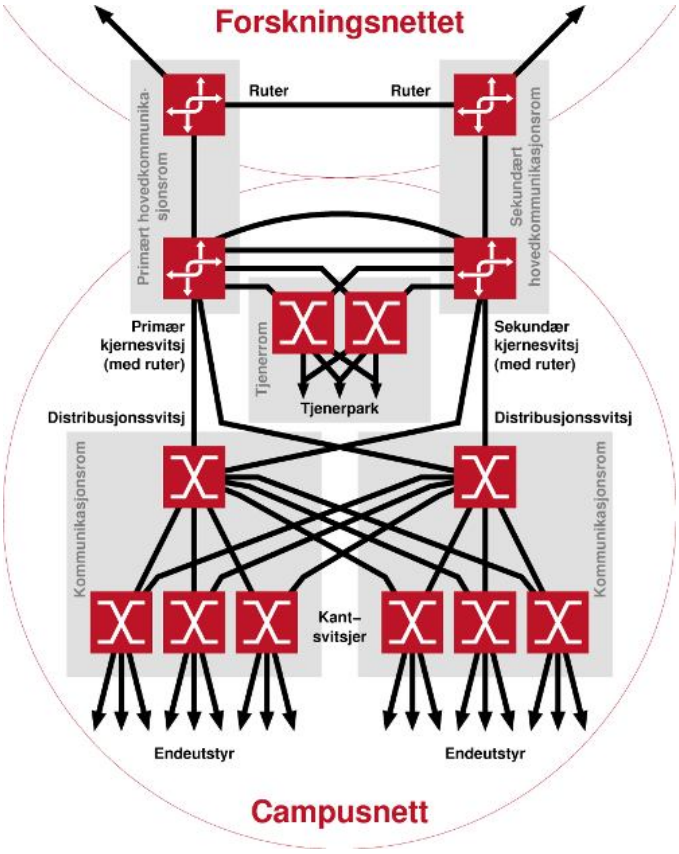
2.2.1 Fullredundant modell

Figur 2 viser eksempel på en fullredundant modell med to adskilte hovedkommunikasjonsrom, dedikert tjenerrom og redundante fibertrasseer til alle kommunikasjonsrom. Løsningen er ideell sett ut i fra et feiltoleranseperspektiv. Den er imidlertid svært kostbar. Ut i fra en kost/nytte-vurdering vil vi ikke anbefale en slik løsning i UH-sektoren, men modellen tas med for å illustrere hvordan potensialet kan tas helt ut.

Følgende elementer er her dekket:

- To adskilte hovedtelematikkrom.

- To adskilte UNINETT-rutere og to adskilte kjerne-svitsjer. Feiltolerant ruting med BGP anbefales, for detaljer se UFS132 [6].
- Redundant L3 design på/mellom campus med IS-IS eller OSFP.
- Redundant aksessruter for alle lokalnett med bruk av VRRP, HSRP eller tilsvarende.
- Redundant L2 struktur med lav konvergenstid (RSTP).
- Distribusjon- og kantsvitsjer har tilkobling til to andre "uplink"-svitsjer. Dette medfører omfattende bruk av fiber.
- Adskilt tjenerpark i separat rom. Redundant aksess fra hver tjener mot to adskilte tjenersvitsjer. Hver tjenersvitsj har to adskilte forbindelser til hver kjernesvitsj.

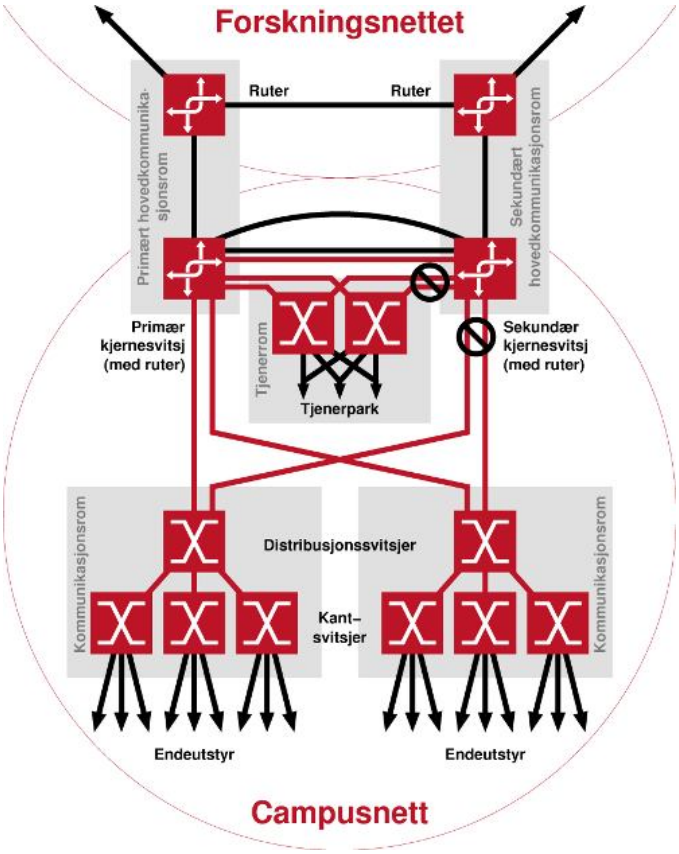


Figur 2: fullredundant kjernenett

2.2.2 Anbefalt tradisjonell løsning

I løsningen vi anbefaler er graden av redundans tonet noe ned, se figur 3. Her er følgende elementer på plass:

- To adskilte hovedkommunikasjonsrom.
- To adskilte UNINETT-rutere og to adskilte kjernesvitsjer med ruting. To redundante BGP sesjoner settes opp, en på primær og en på sekundær tilkobling. Man kan enten kjøre aktiv/passiv eller kjøre trafikk på begge linkene. Sistnevnte er bedre for å forsikre seg om at begge forbindelsene virker, men vil gi en del ekstra lag 2 trafikk mellom de to kjerneruterne på campus.
- I tillegg settes det opp en intern BGP forbindelse mellom de to kjernesvitsjene. Det er svært viktig at denne BGP-sesjonen alltid er oppe, ellers får man problemer med trafikk inn og ut av campus. For å ivareta dette må det etableres to redundante lag 3 linkforbindelser mellom primær og sekundær kjerneruter. Disse bør helst følge ulike føringsveier.



Figur 3: Anbefalt kjernenett

- Videre er det nødvendig å kjøre en dynamisk intern ruting protokoll på campus. Her kan man enten velge IS-IS eller OSFP. OSPF har bredere utstyrsstøtte, mens IS-IS har den fordelen at samme rutingprosess kan håndtere både IPv4 og IPv6.

- Redundant aksessruter for alle lokalnett med bruk av VRRP, HSRP eller tilsvarende. Primær aksessruter bør være samme ruter som primær BGP-ruter (når BGP er satt opp aktiv/passiv)

- L2 ringer bygges mellom distribusjonsvitsj og kjernesvitsj for redundans. Dette krever spanning tree (RSTP) for å bryte løkker. Rot i spanning-tree bør settes til primær kjernesvitsj. Spanning-tree blokkeringer blir da i normaltilfellet som vist på figur 3. Les mer om distribusjonsløsninger i kap. 3.

- Løsningen krever to fiberforbindelser ut av hvert kommunikasjonsrom:

    - En ideell løsning medfører adskilt og redundant fiberføring ut av alle kommunikasjonsrom, da mot de to respektive hovedkommunikasjonsrommene.

    - En mer nøktern løsning er å føre fiber mot kun et av hovedkommunikasjonsrom og i stedet besørge rikelig med fiber mellom de to hovedrommene. Da føres både hovedvei og redundant veil til tilsoknende hovedrom, mens redundant vei patches videre til det andre hovedrommet. I et slikt oppsett vil nettverket overleve strømstans eller kjøleproblemer i det ene hovedrommet, sannsynligvis også et mindre branntilløp. Men løsningen er sårbar i forhold til fiberbrudd.

- Adskilt tjenerpark i separat rom. Redundant aksess fra hver tjener mot to adskilte tjenersvitsjer. Hver tjenersvitsj har to adskilte forbindelser til hver kjernesvitsj. Dersom bladsystem benyttes gjelder det samme her. Tjeneroppsett behandles i mer detalj i kapittel 4.

Ved overgang fra en arkitektur med en kjernesvitsj (jfr. kap 2.1) til en arkitektur med to kjernesvitsjer er det viktig å få skilt ut alle tjenere på egne svitsjer ellers oppnår man ikke reell redundans for disse. Det samme gjelder sluttbrukere. De bør ikke kobles rett mot kjernesvitsj. Kjernesvitsjene bør ikke ha annet enn fiberforbindelser til andre svitsjer, samt til UNINETT sine rutere.

Vi anbefaler heller ikke bruk av servicemoduler i kjernesvitsjen, som for eksempel trådløs controller, da dette gjør design mer låst til aktuelle plattform for fremtiden. Det er generelt sett ikke lurt å blande for mange funksjoner i samme enhet.

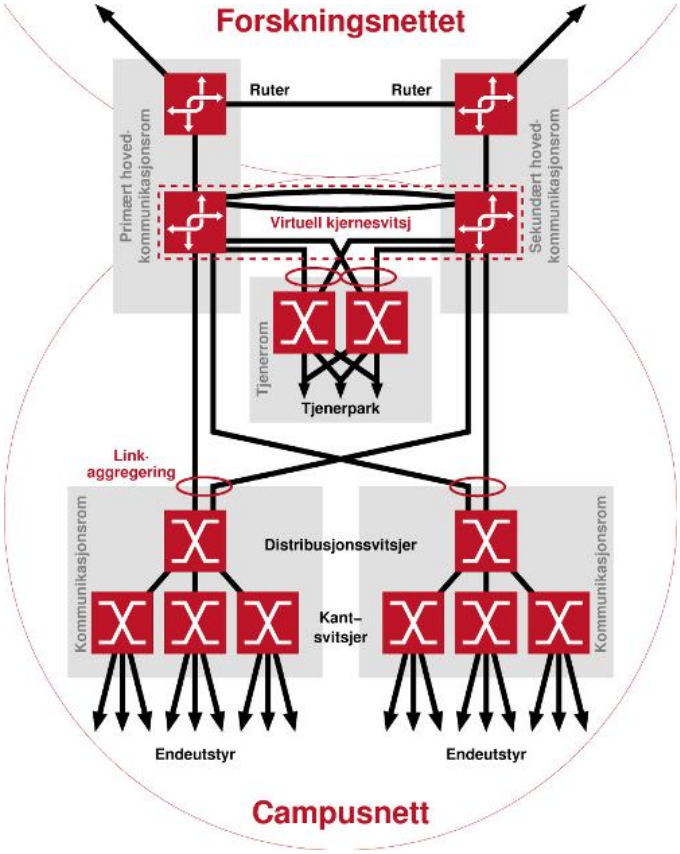
2.2.3 Anbefalt virtuell løsning

Dersom man har to kjernesvitsjer på campus som skissert i 2.2.2 (og ikke flere enn to), så kan samme fysiske topologi realiseres på en annen logisk måte ved å benytte seg av virtualisering i kjernen. Prinsippet er da at to fysiske chassis kobles sammen med et sett med dedikerte fiberforbindelser som danner et proprietært bakplan mellom disse og de to boksene vil logisk sett opptre som en enhet. Dette har en rekke fordeler da det forenkler design uten å miste ønsket grad av redundans.

Figur 4 skisserer hvordan design med virtuelle kjernesvitsjer blir.

Alle leverandørene (Cisco, HP, Juniper) som er inne på UH-sektorens felles avtale for nettelektronikk kan levere dette. Eksempler på implementasjoner er Cisco VSS (Catalyst 6500-plattform), Cisco vPC (Nexus-plattform), Juniper virtual chassis (EX4200, EX4500, EX8200), HP IRF (A-serien, tidligere H3C). Det er så langt liten erfaring med dette i sektoren utover noen implementasjoner av Cisco VSS (Virtual Switching System).

En utfordring med slik virtualisering er at det blir proprietært til leverandøren. Det settes begrensninger i hva slags hardware man kan velge, herunder hva slags management modul og linjekort. Eksempelvis krever Cisco VSS nyere Sup720 management modul og 67xx serien linjekort. Virtualiseringen kan også gi skjult kompleksitet og således en mer komplisert omgivelse for feilsøking. Virtualiseringsteknologiene er relativt ferske og ikke like velprøvd som IS-IS/OSPF, VRRP og spanning tree protokollene.



Figur 4: Løsning med virtuell kjernesvitsj

En virtuell kjerneløsning har imidlertid mange interessante fordeler:

- Behovet for dynamisk internruting med IS-IS eller OSPF opphører da man nå bare har en logisk kjernesvitsj. Man må ha BGP som før mot UNINETT.
- Behovet for å sette opp redundant aksessruter med VRRP eller tilsvarende opphører. Man har implisitt denne redundansen uansett.
- Man realiserer enkelt redundans mot andre svitsjer ved å bruke link aggregering (IEEE 802.3ad). Øvrige svitsjer vil fysisk kables mot hver sin fysiske kjernesvitsj, men logisk tror de dette er samme svitsj og man kan uten problem implementere link aggregering for lastdeling og redundans. Man blir således ikke prisgitt spanning tree for redundans i distribusjonsnettet. Spanning tree bør fortsatt være påskrudd, men da kun for å detektere utilsiktede løkker i topologien og i tilfelle blokkere disse. Mer om spanning tree i kapittel 3.

3 Distribusjonsnettet

Distribusjonsnettet forbinder aksessnettet med kjernenettet, dvs forbindelsen fra aksess-svitsjene til stamnettet. Det er ingen ruting i distribusjonsnettet. IEEE 802.1q benyttes på trunkene i distribusjonsnettet for å transportere flere vlan over samme forbindelse.

3.1 Spanning tree

Spanning tree protokollen benyttes i distribusjonsnettet for å detektere eventuelle løkker og da bryte disse. Den opprinnelige spanning tree standarden av 1990, IEEE 802.1D (STP), er veldig konservativ og har meget treg konvergens. Cisco introduserte tidlig proprietære mekanismer for å bøte på dette (port fast, uplink fast, backbone fast). I 2001 kom Rapid Spanning Tree Protocol (RSTP), IEEE 802.1w, som radikalt forbedrer konvergenstiden ved topologiendringer i distribusjonsnettet (typisk fra 30-50 sekunder ned til noen få sekunder).

Det er pågående arbeid for å utarbeide standarder som går helt bort fra spanning tree algoritmen og i stedet gå over til en link state algoritme basert på IS-IS. IEEE 802.1aq er en slik løsning, en annen er IETF sin RFC 5556, TRILL (Transparent Interconnection of Lots of Links). Det er for tidlig å anbefale slike løsninger, da støtten hos leverandørene er mangelfull.

Pr dato anbefales å bruke IEEE 802.1w, Rapid Spanning Tree (RSTP)

Men bilde er mer sammensatt enn dette. Bør man kjøre en felles spanning tree instans for alle vlan, eller bør man kjøre en instans pr vlan, eller en hydrid av dette? Eller kan man klare seg helt uten spanning tree i noen tilfeller?

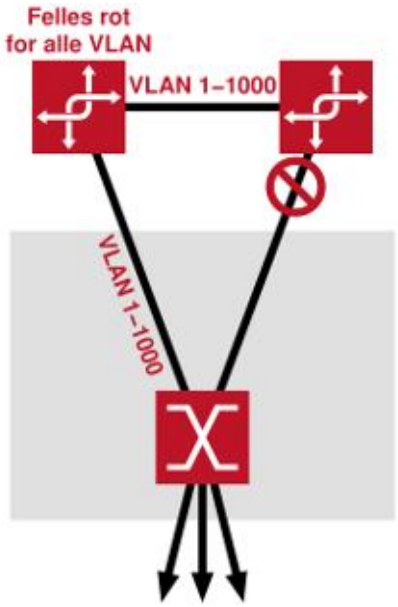
For å ta det siste først. Dersom distribusjonsnettet ikke inneholder noen løkker, men er en ren trestruktur, så er spanning tree i prinsippet overflødig. Vi sier i prinsippet for løkker kan oppstå ved utilsiktede feilkoblinger og da ønsker du mekanismer som detekterer og løser opp i dette. Her finnes dog implementasjoner som sørger for dette pr kantsvitsj (f.eks. loop protect på HP) som er mindre CPU-krevende enn spanning tree. Generelt sett vil det være en fordel om du kan melde ut alle aksessporter av spanning tree og detektere loop her på annet vis.

CPU-prosessering kan nemlig være en problemstilling med spanning tree dersom du har veldig mange vlan og hvert vlan har sin egen spanning tree instans som topologi må beregnes for. Merk at det finnes ingen standard for per vlan spanning tree, kun proprietære implementasjoner. Cisco sin løsning heter PVST+[1], mens for Juniper heter det VSTP. HP A-serien har også PVST+. Motstykket er Common Spanning Tree (CST) der man har en felles spanning tree instans for alle vlan. Den opprinnelige IEEE 802.1q standarden la opp til CST. CST har åpenbare svakheter ut i fra et lastdelingsperspektiv.

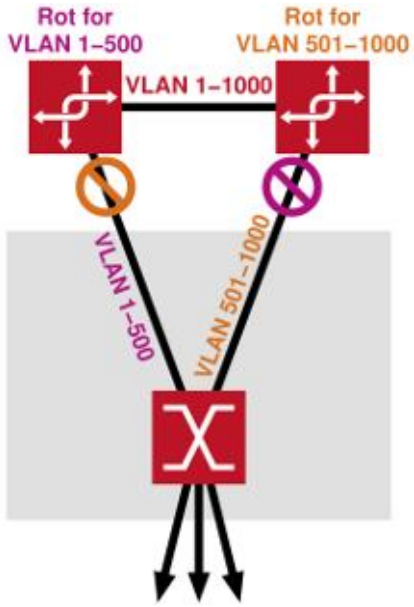
[1] Forgjenger til PVST+ er PVST. PVST støttes bare over ISL-trunker, mens PVST+ støttes over 802.1q trunk.

La oss ta et realistisk eksempel der en kant-/distribusjonssvitsj har to ulinker til hver sin kjernesvitsj som vist på figur 5. Det er 1000 vlan som går over trunkene. Med CST vil alle vlan gå over samme trunk, mens den andre står helt ubrukt.

Implementerer du derimot per vlan spanning tree kan du selv balansere hvilke vlan som skal gå på hvilken trunk. Ulempen er som nevnt CPU-prosessering. Rasjonelt sett er det jo svært unødvendig å gjøre 1000 topologiberegninger for 2 ulike scenarioer. Nettopp dette er bakgrunnen for MSTP, Multiple Spanning Tree (IEEE 802.1s)[2]. MSTP tillater at du kan gruppere dine vlan i grupper av spanning tree instanser. Innen hver gruppe kjører rapid spanning tree. I eksemplet i figur 6 er to grupper satt opp.



Figur 5: CST gir ingen lastdeling



Figur 6: MSTP grupperer vlan i grupper

Ulempen med MSTP er at den krever en del planlegging og konfigurering. Det vil være mer arbeid å innføre et nytt vlan. Det er ingen større implementasjoner av MSTP i sektoren i dag til tross for at MSTP er godt egnet i hybride miljøer.

Dersom man har et oppsett med kjerne/distribusjon basert på Cisco eller Juniper og HP på kant, kan man kjøre PVST+/VSTP i kjernen og CST på kant.

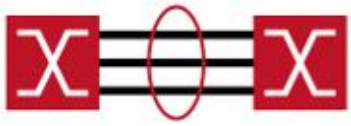
3.1.1 Link Fault Management / Unidirectional Link Detection

IEEE 802.3ah definerer Link Fault Management (LFM) som en protokoll for å detektere linker som ikke er bidireksjonelle. Juniper har implementert LFM, mens Cisco og HP har en proprietær implementasjon av det samme, Unidirectional Link Detection (UDLD). Det anbefales at man konfigurerer LFM / UDLD på linkene i distribusjonsnettet. Hensikten er å ta ned porter der man har link i bare en retning. Tilfeller av link i en retning kan gi svært uheldige følger, da de ulike svitsjene vil få et ulikt bilde av topologien og dette kan gi utilsiktede løkker eller uønsket trafikkmønster.

[2] MSTP er også inkorporert i IEEE 802.1q fra 2005.

3.2 Link aggregering

Link aggregering er definert i standarden IEEE 802.3ad[3] og er en metode som brukes for å bundle flere linker mellom to svitsjer (eller en svitsj og en tjener) til en logisk forbindelse, se figur 7. Dette gir økt kapasitet på denne forbindelsen. Man kan eksempelvis bundle 4 stykk 1 Gbps forbindelser for å få 4 Gbps kapasitet. Lastdeling blir gjort i begge retninger og baseres seg algoritmen for trafikkfordeling baserer seg på kilde-destinasjon IP-adresse (og evt. port). LACP, Link Aggregation Control Protocol, benyttes for å dynamisk forhandle størrelsen på bundelen. LACP gjør bl.a. at brutte linker automatisk blir tatt ut av bundelen i feilsituasjoner, noe som er viktig.



Figur 7: Link aggregering

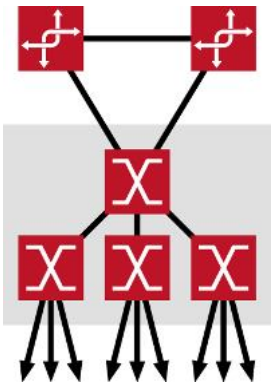
Link aggregering har ingen påvirking på topologien. Det er bare en bundling av flere linker til en større forbindelse mellom to svitsjer. Hver imidlertid oppmerksom på at STP port kostnad kanskje må justeres manuelt slik at balansering blir riktig (Cisco og Juniper gjør dette automatisk, men ikke HP).

Les mer om proprietære, multichassis utvidelser til link aggregering i kapittel 4.1.2.

3.3 Redundant distribusjon

Vi går her inn på fire ulike måter å implementere et redundant distribusjonsnett. Det er fordeler og ulemper med hvert design.

3.3.1 Redundant distribusjonssvitsj i hvert rom



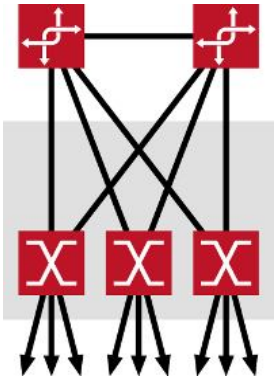
Her er det en sentral distribusjonssvitsj i hvert kommunikasjonsrom som har redundant fiberforbindelse mot to ulike kjernesvitsjer. Kantsvitsjene har kun en uplink, da på TP (1 Gbps eller 10 Gbps) lokalt i rommet.

- Fordeler:
  - Kantsvitsjer trenger ikke annet enn TP-porter, hvilket gir mulighet for billige kantsvitsjer.
  - Løsningen krever bare to fibre ut av rommet.
- Ulemper:
  - Det er ingen redundans på kantsvitsj nivå.
  - Det kan ofte bli sløsing med porter på distribusjonssvitsj.

[3] IEEE 802.3ad kom i 2000. Senere har IEEE 802.1 arbeidsgruppen overtatt arbeidet med link aggregering og i 2008 kom 802.1ax som definerer en medieuavhengig link aggregering (ikke lenger avgrenset til ethernet). Merk også at forløperen til IEEE 802.3ad var Cisco sin proprietære EtherChannel.

3.3.2 Redundante kantsvitsjer direkte mot kjerne

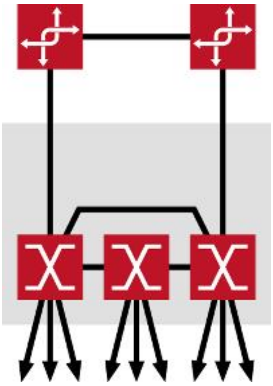
Her er det ingen distribusjonsvitsj involvert, men hver enkelt kantsvitsj er koblet redundant direkte mot kjernesvitsjene.



- Fordeler:
  - Krever ikke dedikert distribusjonssvitsj.
  - Gir redundans på kantsvitsjnivå.
- Ulemper:
  - Koster mye i fiber. Svært mye fiber ut av rommet.
  - Krever to fiberporter i hver kantsvitsj og større beslag på fiberporter i kjernesvitsjene.

3.3.3 Redundant kantsvitsj-stack

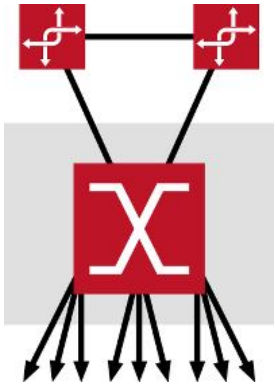
I dette tilfellet er kantsvitsjene stacket og stacken har en redundant kobling mot kjernenettet.



- Fordeler:
  - Krever ikke dedikert distribusjonssvitsj.
  - Krever bare to fibre ut av rommet.
- Ulemper:
  - Gir suboptimalt trafikkmønster fra kantsvitsj til kantsvitsj. Dvs at trafikk går i kjede gjennom flere svitsjer.

3.3.4 Redundant, modulær kantsvitsj

Her benyttes en større chassisbasert kantsvitsj som betjener hele kommunikasjonsrommet.



- Fordeler:
  - Ryddig implementasjon og forenklet administrasjon (en svitsj for hele rommet).
  - Krever ikke dedikert distribusjonssvitsj.
  - Krever bare to fibre ut av rommet.
- Ulemper:
  - Stort utfall dersom hele svitsjen (management modulen) ryker. Man bør definitiv ha reservedeler og god overvåkning.

4 Aksessnettet

Aksessnettet omfatter tilkobling fra endemaskinene mot kantsvitsjer. Tjener- og klienttilkobling inngår her. Vi gir naturlig nok tjenertilkobling bredest omtale.

Redundans i forhold til tjenestene kan naturligvis bygges uten at hver enkelt tjener er redundant. Dette kan være en god nok løsning. Vi anbefaler uansett å vurdere et tjenernett-oppsett som gir redundant nett tilkobling pr tjener.

4.1 Edge porter

Det er ikke ønskelig at link transisjoner mot endemaskiner, som typisk forekommer hyppig, medfører en ny beregning av RSTP-topologien. Måten å løse dette på er å konfigurere alle slike porter til ”edge port”[4]. En ”edge port” vil gå rett i forwarding state etter link og ikke vente på en spanning tree beregning, noe som i seg selv er en stor fordel (man får nett med en gang). Som en ekstra beskyttelse mot utilsiktede løkker skapt av endeutstyr skal en ”edge port” enten bli tatt ned eller miste sin ”edge port” status og bli med i spanning tree topologien dersom BPDU-pakker detekteres bak porten. Dette kan skje dersom en svitsj blir tilkoblet, eller en utilsiktet løkke blir laget.

HP har en tilleggsfunksjon som heter ”loop protect” som anbefales. Den vil hjelpe i scenarioer der det er dannet utilsiktede løkker og *ingen* BPDU-pakker blir detektert. Det kan eksempelvis skje i tilfeller der billige svitsjer som feilaktig forkaster BPDU-pakker kobles inn i nettverket. HP sin ”loop protect” sender ut noen andre kontrollpakker som vil komme gjennom slikt utstyr og svitsjen kan da detektere om de samme pakkene kommer inn på en annen port.

4.2 Redundante top-of-the-rack svitsjer

En god arkitektur i tjenerrommet er å sette opp to uavhengige top-of-the rack svitsjer rundt omkring i maskinrommet. Hver tjener kables alltid mot to slike svitsjer. Tilsvarende gjelder for bladsystem; to uplinker mot to ulike svitsjer. Kapasiteten kan være 1 Gbps eller 10 Gbps avhengig av behov.

Fysisk sett er dette prinsippet. Så spørs det hvordan lastdeling og/eller redundans skal implementeres. Her er det flere veier til mål. En fundamental utfordring er at link aggregering standarden (IEEE 802.3ad, jfr. kap 3.2) for bundling av flere ethernett linker krever at hver ende av ”bundelen” er samme device (svitsj/tjener). Det er *ikke* dette vi vil ha. Da må vi pr dato dessverre over på proprietære løsninger og de fleste leverandørene har et mulig opplegg. Vi kan dele det i tre hovedkategorier; bonding/teaming, multichassis link aggregering og stacking.

[4] Cisco benevner dette ”port fast”.

4.2.1 Bonding/Teaming

Det enkleste oppsettet sett fra nettverkssiden er å bruke “ethernet bonding” på linux-tjenere eller “ethernet teaming” på windows-tjenere. I begge tilfeller kobles de to tjenernettverkskortene mot svitsjeporter på to ulike svitsjer (på same vlan naturligvis). Det er ikke noe spesiell konfigurasjon på tjenersvitsjeportene (med et unntak, nevnt under). Trafikk *ut* av tjenerne kan kjøres lastdelt over de to nettkortene (aktiv/aktiv). Man kan også kjøre i såkalt aktiv/passiv modus, men da får man i realiteten ikke testet redundansen i en normalsituasjon, så dette anbefales ikke.

Hvert nettkort har hver sin unike mac-adresse, men kommuniserer med samme avsender IP-adresse. Trafikk ned til tjeneren vil i realiteten *ikke* bli lastdelt, men vil avhengig av hva som ligger i ARP-cache til ruter (eller andre avsendere på subnettet) følge veien til denne mac-adressen.

Et problem som kan oppstå i dette oppsettet er dersom en av top-of-the-rack svitsjene mister sin(e) uplink(er), dvs sin kontakt med omverden. Tjeneren vil ikke forstå dette og fortsatt lastdelt sende trafikk, da også til dette sorte hullet. For å motvirke et slikt scenario har Cisco og Juniper en feature som heter ”uplink-tracking”. Her vil linken ned til tjeneren/tjenerne blir tatt ned hvis uplinken(e) går ned. Dette bør konfigureres. Det er forventet at andre leverandører også vil få denne funksjonaliteten.

4.2.2 Multichassis link aggregering

Flere leverandører tilbyr en proprietær utvidelse av link aggregering (IEEE 802.3ad, LACP) for å muliggjøre en distribuert kobling fra tjener til nettverk. En måte å omgå dette på er å benytte en virtuell kjerneløsning som forklart i 2.2.3, men ofte har man ikke denne typen utstyr og løsning i tjenerrommet.

Andre implementasjoner som ikke medfører å bygge et virtuelt chassis med beholde de to svitsjene som adskilte administrative enheter er:

- Cisco sin multichassis LACP (mLACP). Dessverre kun støttet på Cat6500 plattform.
- Cisco har også en implementasjon for Nexus-plattformen som benytter virtual port channel (vPC).
- Juniper sin MC-LAG som er støttet på EX-plattformen.
- HP har for sin E-serie en løsning som kalles distributed trunking (dt-lacp). Distributed trunking har den fordelen at den kjører på mange hardware plattformer, også de mindre.[5]
- Andre leverandører har også løsninger.

Hovedprinsippet er den samme i alle tilfellene. Det settes opp en dedikert proprietær forbindelse mellom de to svitsjene som skal danne multichassis link aggregeringsoppsettet. I tilfelle Cisco 6500 heter denne tverrforbindelsen interchassis communication channel (ICC), mens Juniper kaller den Internet Chassis Control (ICCP). HP kaller forbindelsen InterSwitch-Connect (ISC).

Merk at Cisco og Juniper sine løsninger gir et active-standby oppsett ut mot tjener, dvs at kun den ene tjener-forbindelsen blir brukt til enhver tid. HP sin ”distributed trunking” er lastdelt og med spanning tree. I mange scenarioer vil da trafikk inn og ut mot tjenerne i realiteten gå over ISC trunken.

4.2.3 Stacking

Man kan også realisere redundant top-of-the rack løsning i tjenerrom ved hjelp av svitsj-stacking. De fleste leverandørene tilbyr en løsning for stacking av flere enkeltstående svitsjer. Som regel kan disse da administreres som en enhet, noe som i seg selv kan være en fordel. En ulempe kan være trafikkmønsteret. En stor chassisbasert svitsj vil som regel ha et bakplan som er bedre skalert enn det

[5] dt-lapc er støttet på 3500-, 5400-, 6200- og 8200-plattformene i HP sin E-serie (tidligere Procurve).

en stacking-løsning gir. Men sett ut i fra et feiltoleranseperspektiv er de ulike svitsjene i svitsj-stacken helt separate og gir således en økt grad av redundans sett fra tjeneren.

Alle leverandørene som er inne på UH-sektorens felles avtale for nettelektronikk støtter stacking på noen av produktene i sin portefølje.

4.3 Virtuelle svitsjer

Ved bruk av virtualisering kobles de virtuelle tjenerne mot en virtuell svitsj. Det forenkler oppsettet fra tjeneren. Dersom man videre sørger for at den virtuelle svitsjen har redundante uplinker er god feiltoleranse ivaretatt. Dersom det virtuelle tjenermiljøet er på et bladsystem sørger man her for to redundant uplinker ut av bladsystemet.

4.4 Redundante tjenester

Utover tjenerredundans bør alle kritiske tjenester i seg selv være redundante. Vi går ikke i detalj på dette, men nevner to nettnære og basalt viktige tjenestene:

- DNS: Sørg for å ha mist to, redundant plasserte resolvere. Dersom maskinene i nettverket mister tilgang på DNS så blir nettverket i realiteten ubrukelig.
- DHCP: Det anbefales naturligvis DHCP-tjeneste for alle klientnett. Det er rasjonelt og formålstjenlig å sentralisere DHCP-tjenesten, men den bør også implementeres redundant.

4.5 Klienttilkobling

Klienter kobles mot kantsvitsjer. Det er naturligvis ingen redundans på denne tilkoblingen, det lar seg ikke forsvare. For å spare på kablingen kan man velge å legge opp til et TP-punkt pr kontor og benytte en lokal kontorsvitsj. Ulempen med dette er at man får en ekstra, typisk billig komponent i nettverket som kanskje heller ikke kan overvåkes. Det anbefales å kun benytte svitsjer i kommunikasjonsrom som har rikere funksjonalitet, er mer pålitelige og kan overvåkes godt. Vi viser til UFS 105 [5] som tar for seg anbefalt konfigurasjon av svitsjer i campusnett. Aktuell funksjonalitet ut mot sluttbrukerne er bl.a.:

- IEEE 802.1X som krever pålogning for å få nettverksaksess, se UFS133 [8] for detaljer.
- DHCP snooping som hindrer falske DHCP-tjenere å ødelegge oppsett på lokalnettet

Se ellers omtale i kapittel 4.1 om edge porter.

Referanser

[1] UFS 107: Krav til strømforsyning av IKT-rom
https://ow.feide.no/_media/gigacampus:ufs107.pdf

[2] UFS 108: Krav til ventilasjon og kjøling i IKT-rom
https://ow.feide.no/_media/gigacampus:ufs108.pdf

[3] UFS 128: Rammebetingelser og krav til nettverksovervåking av campusnett
https://ow.feide.no/gigacampus:ufs#overvakning

[4] Weathergoose
http://www.itwatchdogs.com/product-detail-weathergoose_ii-1.html

[5] UFS 104: Krav til brannsikring av IKT-rom
https://ow.feide.no/_media/gigacampus:ufs104.pdf

[6] UFS 132: BGP oppsett på campus
https://ow.feide.no/gigacampus:ufs#nett

[7] UFS 105: Anbefalt konfigurasjon for svitsjer i campusnett
https://ow.feide.no/_media/gigacampus:ufs105.pdf

[8] UFS133: Anbefalt oppsett for 802.1X i fastnett
https://ow.feide.no/gigacampus:ufs#nett

Definisjoner

L2: Lag 2 i OSI-stakken. På L2 forstår ikke svitsjen IP-adresser, men forholder seg til mac-adresser.

L2+: Noen svitsjer har evnen til å forstå ulike egenskaper ved IP-header og høyere lag. Et eksempel er DHCP snooping. Slik funksjonalitet benevnes L2+.

L3: Vi er da på lag 3, nettverkslaget, med forståelse av IP-adresser hvor ruterne opererer. Noen svitsjer kan også gjøre ruting. Vi omtaler dem som L3-svitsjer.

Kantsvitsj: En svitsj som står i periferien av nettverket, nærmest brukerne.

Distribusjons-svitsj: En svitsj som håndterer aggregert trafikk fra en rekke kantsvitsjer og forbinder dette med kjernesvitsjer.

Kjernesvitsj: Svitsjer som står i kjernen av nettet og i hovedsak ikke har brukere direkte tilkoblet, primært høykapasitets forbindelse til andre svitsjer og tjenere.

Klientporter: Porter på svitsjen som er rettet mot klientmaskiner i nettverket. Dette inkluderer også tjenere, skrivere og annet endeutstyr. Slike porter har en del andre egenskaper enn nettverksporter, altså porter som har forbindelse til andre nettverkskomponenter (rutere, svitsjer, basestasjoner).

Ved spørsmål omkring denne eller andre UFSer – kontakt campus@uninett.no
Andre UFSer er tilgjengelige på www.uninett.no/ufs