Reductie van de convergentietijd klantaansluitingen SURFnet

Andree Toonk
Leendert van Doesburg

28 juni 2004

Samenvatting

Klanten van SURFnet kunnen op verschillende manieren aansluiten op SURFnet5. Een groot deel van de klanten is aangesloten via Gigabit Ethernet technologie over eigen wide-area glasvezelverbindingen. SURFnet hanteert daarbij afhankelijk van de beschikbare glasvezelinfrastructuur verschillende aansluitmodellen. Bij twee van deze aansluitmodellen, de GigaMAN en de multi-POP (eBGP) aansluiting, ligt de convergentietijd relatief hoog. Deze zijn 30 tot 50 seconden voor de GigaMAN aansluiting en 2 tot 3 minuten voor de 'multi-POP' aansluiting. De wens van SURFnet is om deze tijden te reduceren.

Voor de GigaMAN aansluitingen waarbij de klanten op een stadsring zijn aangesloten, wordt in de huidige situatie gebruik gemaakt van het Spanning Tree Protocol (STP). Dit protocol convergeert met de default configuratie relatief langzaam (maximaal 50 seconden). Er zijn verschillende methoden om de convergentietijd voor deze aansluitingen te reduceren. Mogelijkheden zijn om de timers aan te passen, gebruik te maken van (Cisco proprietary) enhancements of migratie naar het Rapid Spanning Tree Protocol(RSTP). Na onderzoek blijkt een migratie van STP naar RSTP de beste oplossing te ziin. Hiermee wordt de convergentietijd gereduceerd naar 0.5 tot 2 seconden. Een migratie naar RSTP is relatief eenvoudig en brengt geen problemen met zich mee. Er zijn tijdens dit project diverse testen en metingen gedaan.

De klanten die een 'multi-POP' aansluiting hebben, zijn verbonden met twee POP routers. Met beide POP routers is een (e)BGP sessie opgezet. Ook bij deze aansluiting wordt gebruik gemaakt van de default paramaters (althans, aan de SURFnet zijde). De convergentietijd ligt daarmee tussen de 2 en 3 minuten. Door de parameters voor de betreffende klant-verbinding aan te passen kan de convergentietijd worden gereduceerd tot enkele seconden. De parameter wijziging hoeft slecht op één van de routers te worden doorgevoerd. Tijdens het opzetten van de BGP sessie wordt onderhandeld over de timer-waarden. Hierbij wordt de laagste waarde gekozen. Klanten hebben in de huidige situatie dus al de mogelijkheid de convergentietijd te reduceren. Het blijkt dat er slechts één klant is, die dit nu al heeft aangepast. Om eventuele nadelige gevolgen van veel klanten met een te lage timer keuze te beperken, is het noodzaak dat SURFnet zichzelf hier tegen beschermd.

Inhoudsopgave

Hoofdstuk 1

Inleiding

Ter afsluiting van de studie Systeem en Netwerkbeheer aan de UvA, is een project van 4 weken uitgevoerd. Dit 'analytisch network project' (ANP), heeft plaatsgevonden bij SURFnet. Er is tijdens dit onderzoek getracht de convergentietijd van de klantaansluitingen van SURFnet te verbeteren. De tijd die het duurt voordat een klant na een fiber-cut weer verbinding heeft duurt nu maximaal 3 minuten. Dit vindt SURFnet te lang, de opdracht is om deze tijd te reduceren. Er is onderzoek gedaan naar 2 verschillende aansluit methoden. Per aansluit methode is de aanpak hetzelfde. Allereerst is gekeken naar de verschillende theoretische mogelijkheden. Nadat één of meerdere technologiën zijn gekozen, zijn deze getest in het communicatie lab van de Hogeschool van Utrecht. Hier is de beschikking over allerlei soorten netwerk apparatuur, waarop diverse zaken getest kunnen worden. Nadat in het communicatie lab op de HvU getest is, zijn soortgelijke testen gedaan op het test netwerk van SURFnet. Dit is voor de verificatie van de gemeten resultaten in eerdere metingen en om te controleren of dit ook op de SURFnet apparatuur werkt. Eén van de wensen van SURFnet is bovendien om een werkende test-opstelling op te leveren wat hiermee wordt gerealiseerd.

Na het doorlezen van dit document heeft de lezer een goed beeld gekregen van de huidige SURFnet klantaansluitingen. Daarnaast wordt stap voor stap een beschrijving gegeven van hoe de convergentietijd van de klantaansluitingen gereduceerd kan worden.

Over SURFnet

SURFnet is het Nederlandse computernetwerk voor hoger onderwijs en onderzoek. SURFnet verbindt de netwerken van universiteiten, hogescholen, onderzoekscentra, academische ziekenhuizen en wetenschappelijke bibliotheken met elkaar en met andere netwerken in Europa en de rest van de wereld. SURFnet maakt onderdeel uit van het wereldwijde internet. Via het SURFnet-netwerk kunnen gebruikers vanaf hun werkplek, of via de PC thuis, communiceren met andere netwerkgebruikers en informatie raadplegen op andere computers aangesloten op SURFnet of elders op het internet. SURFnet zorgt voor goede koppelingen met buitenlandse onderwijs- en onderzoeksinstellingen. SURFnet maakt daarbij gebruik van een hoogwaardige infrastructuur en geavanceerde technologien. Instellingen voor hoger onderwijs en onderzoek kunnen volledig gebruik maken van SURFnet.

In het kader van het nationale GigaPort-project kunnen nu ook bedrijven en andere instellingen SURFnet gebruiken voor het testen en ontwikkelen van nieuwe, geavanceerde diensten. Dankzij voortdurende innovatie en de exclusiviteit van het netwerk beschikken de gebruikers van SURFnet

over één van de snelste en meest geavanceerde netwerken ter wereld. Naast snelheid staan ook de betrouwbaarheid en veiligheid van het netwerk hoog in het vaandel. Ongeveer 750.000 studenten en medewerkers van de ruim 150 op SURFnet aangesloten instellingen maken vrijwel dagelijks van het netwerk gebruik.

Hoofdstuk 2

Opdrachtomschrijving

2.1 De opdracht

De opdracht bestaat uit het onderzoek doen naar de verbetering van de convergentietijden van klantaansluitingen bij SURFnet. Het doel is om het gevolg van een glasvezelbreuk waarbij het netwerk nu enkele tientallen seconden niet voor de betreffende klant beschikbaar is, terug te brengen naar een aanvaardbare waarde zodat de breuk door de klant niet of nauwelijks opgemerkt zal worden. Hierbij gaat het om op twee verschillende manieren aangesloten klanten welke gebruik maken van de glasvezelinfrastructuur:

- Klanten welke door infrastructurele redenen niet zoals de meeste klanten direct op één van de vijftien POP's (main connection points) van SURFnet zijn aangesloten, maar indirect via een connection point op twee verschillende POP's van SURFnet zijn aangesloten. Hierbij zorgt (e)BGP voor de redundantie wanneer een van de POP's onbereikbaar wordt.
- Klanten welke aangesloten zijn op een GigaMAN, een stadsring. Ook een POP van SURFnet is op deze ring aangesloten. Hierbij zorgt het spanning-tree protocol (STP) bij onderbreking van de ring voor de redundantie.

Door niet te veel gedetailleerde technische informatie te vergaren van de inrichting van het huidige netwerk en zo met een geheel nieuwe kijk tegen het netwerk aan te kijken, wordt gestimuleerd om (mogelijk) met nieuwe inzichten naar een oplossing te zoeken in plaats van door de gaan op de reeds eerder ingeslagen weg. Anderzijds zal toch naar de huidige situatie gekeken moeten worden om te voorkomen dat een mogelijke oplossing teveel veranderingen van het totale netwerk teweeg zal brengen.

Allereerst zal worden gekeken naar de opzet van de geleverde netwerkfunctionaliteiten. Kennis hiervan is van belang om de samenhang tussen de verschillende deelaspecten te kunnen plaatsen.

Een voorwaarde welke gesteld wordt is dat de oplossing gerealiseerd moet kunnen worden op de nu gebruikte netwerkcomponenten.

De klanten welke direct op de POP van SURFnet zijn aangesloten hebben geen last van hoge convergentietijden. De twee soorten klantaansluitingen 'multi-POP klantaansluiting' en 'GigaMAN klantenaansluiting' welke wel last hebben van hoge convergentietijden, nemen toe in aantal. Voor deze twee soorten zal er gekeken worden hoe deze tijden gereduceerd kunnen worden.

2.2 De GigaMAN klantaansluiting:

- De mogelijke, gangbare technologieën op laag 2 welke als toevoeging, vervanger of opvolger van het huidige STP kunnen fungeren (RSTP,Cisco's *fast enhancements), zullen onderzocht en naast elkaar gezet worden.
- Gekeken zal worden naar het 'fine-tunen' van de huidige STP-parameters
- Na keuze van de beste oplossing zal er een simpele configuratie opgezet worden in een test-omgeving voor verificatie van de theorie

2.3 De multi-POP klantaansluiting:

- Er zal onderzocht worden of BGP is toe te passen met een voldoende korte convergentie-tijd. Dit houdt in dat er voor het reeds gebruikte (e)BGP gekeken zal gaan worden naar aanpassingen van parameters.
- Er zal gekeken worden naar de mogelijke toepassing van VRRP/HSRP
- Na onderzoek van het gebruik van een BGP of VRRP/HSRP, zal er een simpele configuratie opgezet worden in een test-omgeving voor verificatie van de theorie.

Hoofdstuk 3

Huidige situatie

Klanten van SURFnet kunnen op verschillende manieren aansluiten op SURFnet5. Een groot deel van de klanten is aangesloten op het SURFnet5 netwerk via Gigabit Ethernet technologie over eigen wide-area glasvezelverbindingen. SURFnet hanteert daarbij afhankelijk van de beschikbare glasvezelinfrastructuur drie aansluitmodellen:

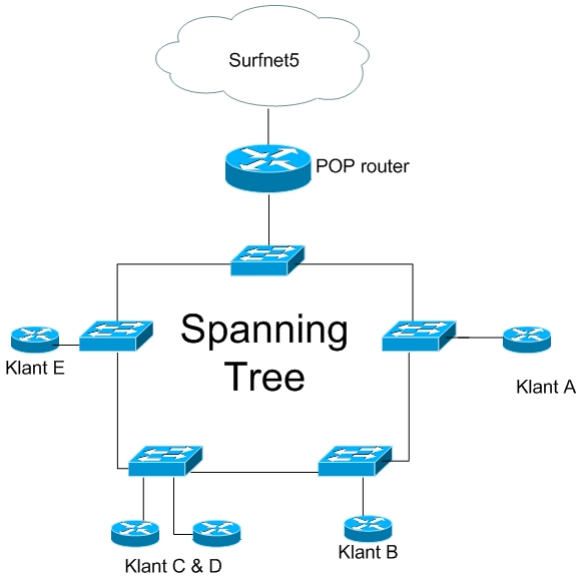
1. één poort op een Gigabit Ethernet switch op een POP-locatie
2. één poort op een Gigabit Ethernet ring (GigaMAN)
3. twee afzonderlijke Gigabit Ethernet verbindingen (multi-POP).

De meeste klanten van SURFnet bevinden zich in of nabij de directe omgeving van een POP-locatie. Deze klanten sluiten aan op een Gigabit Ethernet switch op een POP-locatie. De (korte) fibers welke voor de verbinding gebruikt worden, zijn hierbij in eigen beheer. Een breuk kan daardoor relatief snel worden verholpen. Anders is het voor langere fibers welke niet in eigen beheer zijn en ondergronds langere afstanden afleggen. Hierbij is het van belang dat er een andere fiber voor redundantie aanwezig is omdat het wachten op reparatie te lang zal gaan duren en daar niet op vertrouwd mag worden. Bij voorkeur hebben de fibers een verschillende topologische route om te voorkomen dat beide fibers tegelijk door bijvoorbeeld graafwerkzaamheden kunnen worden verbroken. Vanaf de vijftien verschillende POP's zijn deze redundante verbindingen naar Amsterdam aanwezig. Beide verbindingen worden tegelijk gebruikt. ISIS zorgt voor de convergentie in het geval van een breuk.

3.1 Ring topologie (GigaMAN)

In veel grote steden wordt de glasvezel-infrastructuur steeds verder uitgebreid. Hierdoor wordt het mogelijk om (nieuwe) klanten van SURFnet welke zich niet in de nabijheid van een POP-locatie bevinden toch met Gigabit of hoger op het netwerk aan te sluiten. Doordat het ook hier fibers betreft welke niet in eigen beheer zijn, zal er een redundante fiber beschikbaar moeten zijn tussen de klant naar de POP-locatie. Door gebruik te maken van aanwezige stadsringen is het mogelijk om (financieel) efficiënt meerdere klanten in een ring op te nemen in plaats van één ring voor elke klant afzonderlijk (zie figuur 3.1).

De ringen worden door SURFnet GigaMAN's genoemd en zijn gerealiseerd met ethernet-switches. Van nature zijn ethernet netwerken niet geschikt voor ringstructuren. Wanneer een ring niet wordt



Figuur 3.1: *GigaMAN-ring*

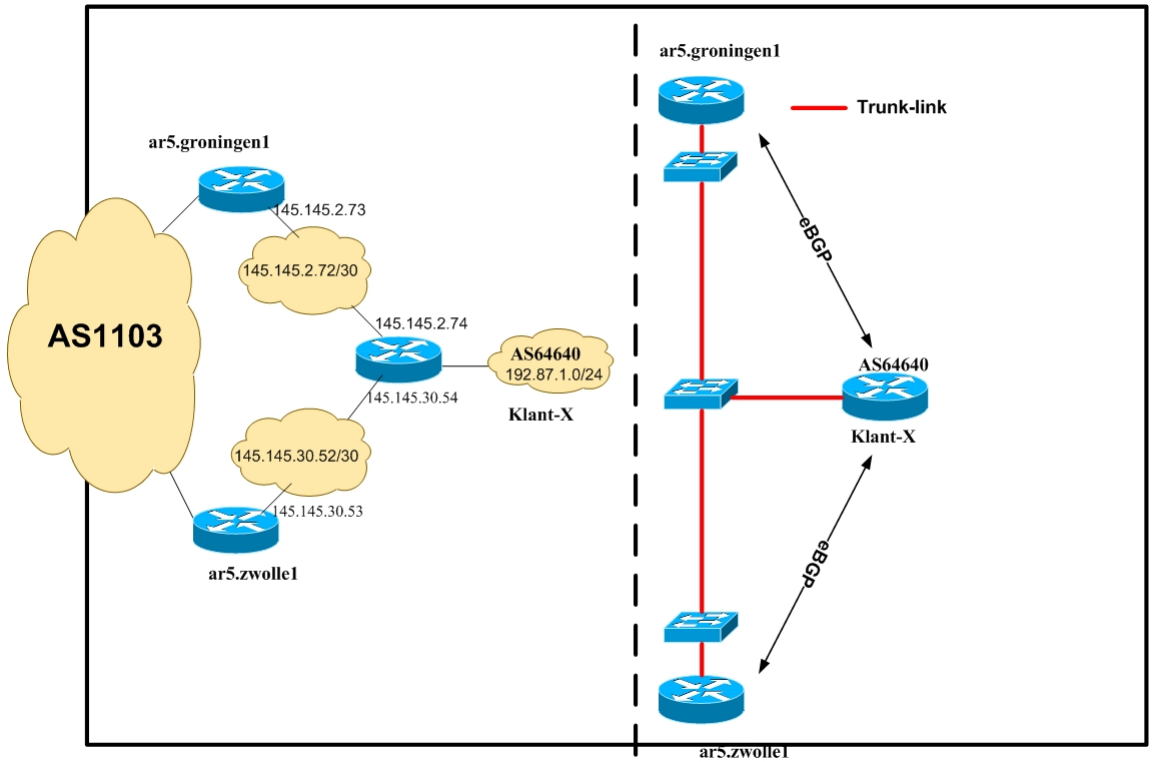
onderbroken, zal er een 'loop' in het netwerk optreden met als gevolg dat het netwerk niet goed meer performt of zelfs onbruikbaar wordt. Om toch van een ring gebruik te kunnen maken en redundancy te realiseren, wordt gebruik gemaakt van het Spanning Tree protocol. Dit protocol zorgt ervoor dat de ring op éń plaats wordt onderbroken. In het geval van een link failure ergens in de ring zal het Spanning Tree Protocol ervoor zorgen dat de redundante link wordt geactiveerd door de onderbreking op te heffen. De tijd die het kost voor het netwerk om te stabiliseren (convergentie) duurt in de huidige situatie enige tientallen seconden.

3.2 Point-to-Point verbindingen (multi-POP)

Als laatste zijn er klanten welke zich geografisch tussen twee POP's van SURFnet bevinden en de mogelijkheid hebben om met twee afzonderlijke fibers op deze twee verschillende POP's aangesloten te worden. Deze klant-aansluiting met een koppeling naar twee POP's, worden verder in dit document de multi-POP aansluiting genoemd. Redundantie wordt gerealiseerd door terug te kunnen vallen op de secundaire verbinding van de tweede POP als de primaire verbinding naar de eerste POP faalt. Waar bij andere aansluitingen volstaan kan worden met een default (statische) route op de klant-router, moet bij deze aansluiting door de klant-router met beide POP-routers (e)BGP worden gesproken. De klant-router adverteert hierbij zijn prefix(en) naar beide routers op de POP-locatie. De POP routers adverteren naar de klant router een default route (0.0.0.0/0) terug.

Om een beter inzicht te krijgen in de huidige situatie, is de multi-POP aansluiting van een willekeurige SURFnet-klant (Klant-X) bekeken. In het voorbeeld van figuur 3.2 is te zien dat de lokatie in twee uplinks heeft. Eén naar Groningen en één naar Zwolle. Links is het laag 3 ontwerp van deze topologie weergegeven en rechts het laag 2 ontwerp.

Ar5.groningen1 en ar5.zwolle1 adverteren beide een default route naar AS64640/Klant-X. Deze worden verzonden met een metric van 10 vanuit Groningen en 20 vanuit Zwolle. Zodoende zal de



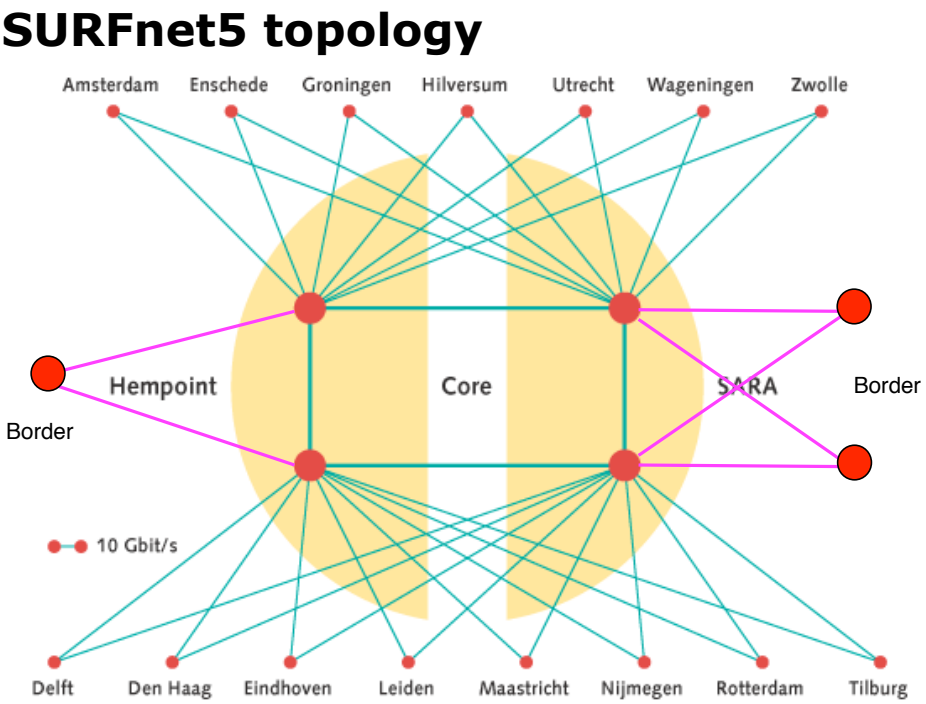
Figuur 3.2: *Multi-POP aansluiting*

router van Klant-X altijd Groningen prefereren voor zijn uitgaande verkeer. Klant-X accepteert alleen de prefix 0.0.0.0./0 (de default route). Bij de Point-to-Point verbindingen wordt dus gebruik gemaakt van het BGP protocol voor redundantie.

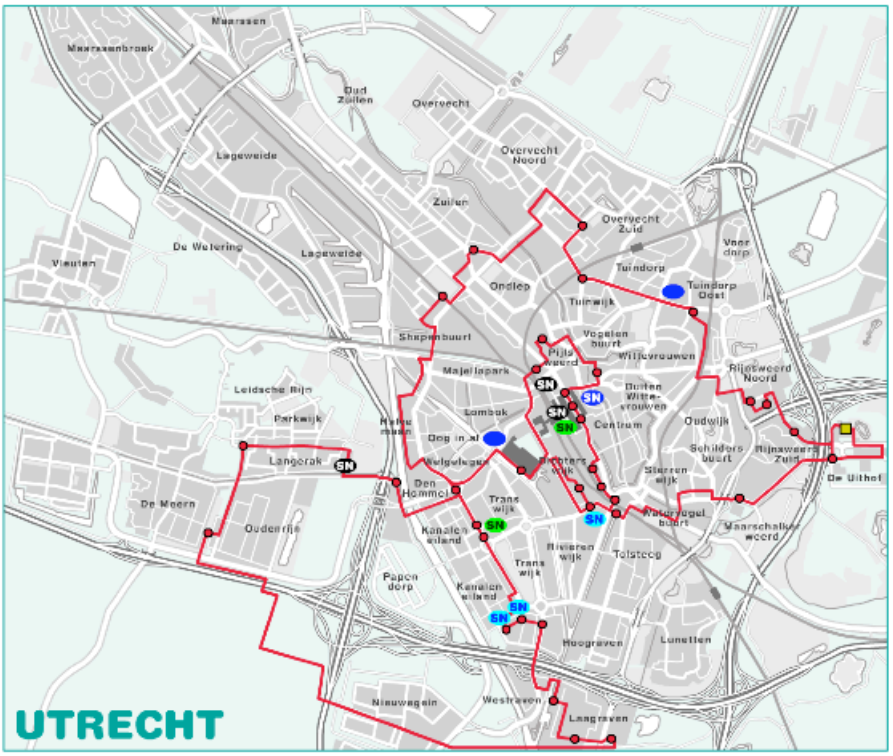
Klant-X (AS64640) adverteert met BGP de eigen prefixen naar zowel de POP router in Zwolle als in Groningen. Zwolle accepteert deze met een localpreference van 195 en Groningen met een localpref van 200[1]. Voor verkeer naar Klant-X, heeft Groningen de 'prefered route'. Groningen en Zwolle adverteren de prefixen van Klant-X vervolgens weer door met iBGP naar de vier Core-Routers CR1 t/m CR4 van het netwerk van SURFnet5. Doordat de localpreference wordt meegegeven in iBGP zullen ook de Core Routers Groningen prefereren als het primaire pad naar Klant-X (AS64640).

Figuur 3.3 laat een globaal overzicht zien van het SURFnet5 netwerk met de vier Core-Routers en de vijftien POP-routers. Figuur 3.4 laat een voorbeeld van een stadsring in de stad Utrecht zien.

[1]hoogste localpreference wordt geprefereerd



Figuur 3.3: *SURFnet5 topologie*



Figuur 3.4: *Stadsring Utrecht*

Hoofdstuk 4

Gigaman

In dit hoofdstuk zal als eerste Spanning Tree en met name RSTP onderzocht worden. Hierbij wordt de basiskennis van STP als bekend verondersteld. Daarna wordt gekeken naar mogelijke oplossingen om de convergentietijd van de huidige situatie te reduceren. Verder zullen metingen van de technologieën in het HvU communicatie lab bekeken worden. Hieruit zal een technologie gekozen worden, welke vervolgens getest is op het testnetwerk van SURFnet. Als laatste zullen er aanbevelingen voor implementatie worden gedaan.

4.1 Spanning Tree

Netwerken moeten steeds robuuster en betrouwbaarder worden om aan de wensen van gebruikers te blijven voldoen. Hiervoor is het belangrijk dat een netwerk redundant en zelf herstellend wordt opgezet. Op laag 3 van het OSI-model worden met routerings protocollen redundante routes naar een bestemming mogelijk gemaakt. Als een primair pad faalt, wordt er een secundair pad naar de bestemming gekozen. Load-balancing is hierbij mogelijk door meerdere paden tegelijk te gebruiken. De redundantie op laag 3 wordt gerealiseerd door routers. Redundantie op laag 2 van het OSI-model is lange tijd op de achtergrond gebleven. Om ook op deze laag redundantie te bieden, is door de IEEE het 802.1D Spanning-Tree Protocol (STP ; 802.1d) ontwikkeld. Als een verbinding op laag 2 faalt, zorgt het algoritme ervoor dat een redundante verbinding geactiveerd wordt. Een andere belangrijke funktie van STP is het voorkomen van een loop op laag 2 welke gecreëerd wordt als verbindingen redundant uitgevoerd worden of wanneer er een verkeerde topologie-wijziging wordt gemaakt. Loops op laag 2 veroorzaken ongewenst veel dataverkeer waardoor het netwerk onbeschikbaar wordt. De redundantie op laag 2 wordt gerealiseerd met behulp van switches. Een ringvormig Ethernet netwerk wordt met STP tolerant tegen bekabelingsperikelen zoals kabelbreuken en mispatches. Met STP is het mogelijk om een netwerk automatisch te laten herconfigureren na een kabelbreuk

4.1.1 STP belangrijker

Door de toenemende gebruik van routers van de afgelopen decennia is STP lange tijd op de achtergrond gebleven en werd gezien als 'a less-important protocol that just worked'. De switching technologie ontwikkelde zicht echter eveneens. Switching gebeurde later in hardware en deze hardware werd steeds goedkoper. Mede door het feit dat switches een betere performance bieden dan routers, werden daar waar mogelijk switches ingezet. Door deze tendens werd STP belangrijker

voor het realiseren van redundantie en het voorkomen van loops. Een groot probleem hierbij is dat het (verouderde) STP 50 seconden [max. convergentietijd = (2 x Forward-Delay) + Max-Age] nodig heeft voor het detecteren en actief maken van een backup verbinding. Dit is trager dan veel routerings protocollen.

4.1.2 Convergentietijd verbeteringen STP

Door de timers van STP aan te passen (te 'fine-tunen'), kan de convergentietijd gereduceerd worden. Dit heeft beperkingen en kan een instabiel netwerk tot gevolg hebben. Door switch-fabrikanten werden proprietary oplossingen ontwikkeld om de convergentietijd te verbeteren. Zo ontstonden features als Portfast, Uplinkfast, en Backbonefast. Deze mechanismen zijn aanvullingen op de originele IEEE 802.1d specificaties welke extra configuratie vereisen en niet altijd onderling tussen apparatuur van verschillende switch-fabrikanten werkt. De mechanismen zijn een grote verbetering ten opzichte van de originele STP en worden veel gebruikt.

4.1.3 VLAN-technologie en STP

Met virtual LAN's (VLAN's) werd het mogelijk om op switches meerdere laag 2 broadcast domeinen te realiseren (IEEE 802.1q). De aanpak van IEEE is om hierbij een enkele loop-vrije topologie te hanteren. Sommige switch-fabrikanten zijn overgegaan op een logische topologie voor elk afzonderlijk VLAN zoals Cisco's Per VLAN Spanning Tree+ (PVST+). Het voordeel van de IEEE aanpak met een Common Spanning Tree (CST) is dat dit niet reken-intensief en wel schaalbaar is bij gebruikt van veel VLAN's. Het is hierbij niet mogelijk om load-balancing van verkeer te realiseren door logische verbindingen van de verschillende VLAN's te verdelen over verschillende redundante verbindingen. Dit laatste is met Per-VLAN spanning tree (PVST) van Cisco wel mogelijk. PVST is echter reken-intensiever en minder schaalbaar. Om dit op te lossen is door Cisco het Multi-Instance Spanning-Tree Protocol (MISTP) ontwikkeld welke een enkele spanning tree (instantie) koppelt aan meerdere VLAN's.

4.1.4 RSTP en MST

Om de convergentietijd beperkingen van STP en de schaalbaarheid van meerdere spanning tree's te verbeteren, zijn door de IEEE recent twee nieuwe standaarden ontwikkeld:

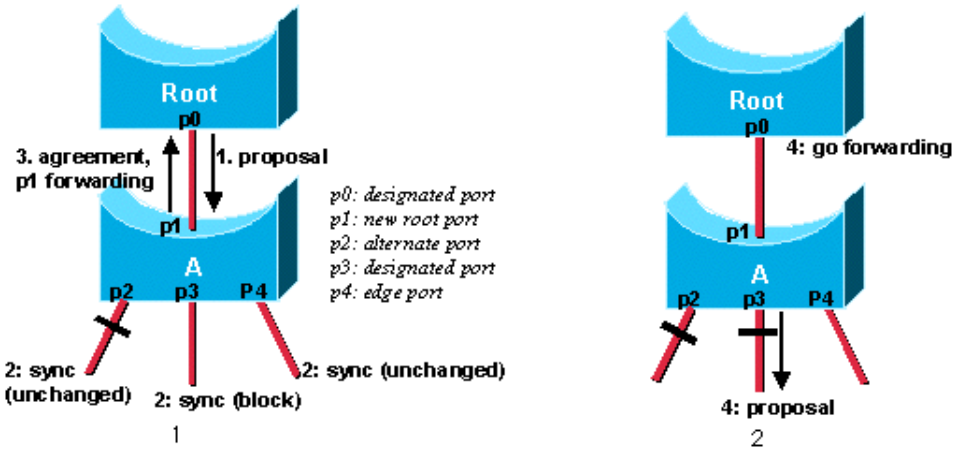
- Rapid Spanning-Tree Protocol (RSTP; IEEE 802.1w)
- Multiple Spanning Trees (MST; IEEE 802.1s)

4.1.5 RSTP

Het Rapid Spanning-Tree Protocol kan gezien worden als een doorontwikkeling op het oudere Spanning-Tree Protocol. RSTP berekent de uiteindelijke topologie voor de spanning tree met hetzelfde algoritme als het oude STP. Hierdoor is het mogelijk om ook aan oude switches te koppelen welke alleen STP ondersteunen. RSTP is dus backwards compatible met STP. De portfast, uplinkfast en backbonefast mechanismen zijn in RSTP verder verbeterd en geïmplementeerd.

Verbeteringen in RSTP t.o.v. STP

RSTP convergeert sneller door het gebruik van het 'active bridge-to-bridge handshake mechanism' in plaats van de in STP gebruikte manier van passief wachten op het convergerende netwerk met de bijbehorende timers. Dit in RSTP snel (rapid) in forwarding mode brengen van de switch-poorten in plaats van passief timers af te wachten, kan als de belangrijkste feature worden gezien. Het hierbij gebruikte handshake mechanisme bestaat uit een 'proposal sent' en een 'agreement reply' op de (blocked) gekoppelde poorten tussen twee switches (zie figuur 4.1.1). Na de agreement gaan de twee poorten meteen over naar de forwarding mode terwijl poorten welke met andere switches verbonden zijn (p2 en p3), op dat moment blocked blijven om loops te voorkomen (zie figuur 4.1.2). Deze blocked poorten doorlopen daarna eveneens het handshake mechanisme. Het mechanisme 'hopt' zo sequentieel van de root-switch naar de 'edge' van RSTP-domein waarbij loops tijdens en uiteraard na het proces niet mogelijk zijn.



Figuur 4.1: *Handshake-mechanisme in RSTP*

De disabled, blocking en de listening modes in STP zijn in RSTP samengevoegd tot één status en wordt de discarding mode genoemd. In deze mode worden frames weggegooid en worden er geen MAC-adressen geleerd. Het aantal port states wordt zo bij RSTP teruggebracht van vijf naar drie: Discarding, Learning en Forwarding.

De rol van de poorten welke los staat van de status, kan zijn: Root, Designated of Blocking. Deze laatste rol wordt opgedeeld in backup of alternate waarvan het uplinkfast mechanisme gebruik maakt. Zo wordt het voor een switch mogelijk om meteen van een backup of alternate poort een root poort te maken als de huidige root poort geen verbinding meer heeft. Het is dus niet nodig om op het moment dat een uplink faalt, een nieuwe root poort te berekenen omdat een alternatief pad al bekend is.

De gebruikte BPDU's in RSTP zijn type 2, versie 2 pakketten en bevatten een aantal kleine wijzigingen t.o.v. de oude STP-BPDU's. Voor het handshake mechanisme en voor de rol en status van de poort (waar de betreffende BPDU vandaan komt), zijn de zes resterende bits het flag-byte gebruikt. Legacy switches met STP negeren deze nieuwere BPDU's, zodat RSTP aan de hand van het niet terugkrijgen van een agreement reply weet dit een STP-switch is. RSTP zal dan STP-BPDU's gaan verzenden en de timers van STP in acht nemen. Het spreekt voor zich dat de voordelen van RSTP hierbij niet gelden. In tegenstelling tot STP worden BPDU's bij RSTP niet vanuit de root-switch

doorgegeven maar onvoorwaardelijk met een frequentie van de hello-time verzonden. Naast het doorgeven van informatie hebben de BPDU nu de funktie van een keep-alive mechanisme voor de betreffende verbinding tussen de twee switches. Na drie gemiste BPDU's wordt aangenomen dat de verbinding verbroken is (port fast aging). In de meeste gevallen zal de switch echter zien dat een fysieke verbinding wegvalt, en hoeft er niet gewacht worden op drie gemiste BPDU's.

Voor het in RSTP snel kunnen overgaan naar de forwarding mode en zo het netwerk snel te laten convergeren, is het belangrijk dat de verbindingstype van de switch-poorten juist geconfigureerd word. Er wordt onderscheid gemaakt tussen 'edge ports' en 'link ports'.

Op edge ports worden alleen eindstations aangesloten. Voor dit type wordt het portfast mechanisme gebruikt waarbij de poort met het aktief worden meteen zonder de listening en learning in de forwarding mode komt. Als er op die poort toch BPDU's ontvangen worden, wordt dit een normale link port.

Link ports zijn de poorten welke de switches onderling verbinden. Hierbij is onderscheid te maken tussen point-to-point en shared links. Als een poort op full-duplex geconfigureerd is, wordt aangenomen dat dit een point-to-point verbinding is. Een half-duplex poort wordt dan met een shared verbinding geassocieerd, dit kan handmatig gewijzigd worden. Alleen als bekend is dat een poort aan slechts een ander device gekoppeld is (wat tegenwoordig meestal het geval is), kan het handshake mechanisme snel werken.

Wanneer er meerdere devices aan een poort gekoppeld zijn (d.m.v. een hub of een switch zonder STP), valt het handshake mechanisme terug op de timers van STP omdat niet bekend is hoeveel switches er gekoppeld zijn welke moeten antwoorden op een proposal. Hier kunnen STP-switches aanwezig zijn. Een eindstation welke door misconfiguratie niet op een edge port maar op een link port is aangesloten, zal door RSTP na een proposal gezien worden als een mogelijke STP-switch en zal na twee maal de forward-delay in forwarding komen. Het nauwkeurig configureren van RSTP is dus van belang om te voorkomen dat er gebruik gemaakt gaat worden van de timers en RSTP zich als STP gaat gedragen.

4.1.6 MST

Zoals Rapid Spanning-Tree Protocol gezien kan worden als een doorontwikkeling op het oudere Spanning-Tree Protocol, kan Multiple Spanning Tree gezien worden als een standaard welke geïnspireerd is door het Multi-Instance Spanning-Tree Protocol (MISTP). MST is een complex protocol dat kan omgaan met zowel STP als RSTP.

Om (R)STP schaalbaar te maken kunnen er met MST VLAN's gegroepeerd worden tot één of meerdere instantie's. Een instantie bestaat uit een spanning tree en bevindt zich in een regio met meerdere switches met dezelfde configuratie-instellingen. Door met een regio te werken kunnen daarbinnen meerdere instanties geconfigureerd worden, terwijl de regio naar buiten toe één instantie (topologie) is en zich zo voor compabiliteit kan voordoen als een virtuele switch. Een regio bevat een IST (Internal Spanning Tree), welke altijd aanwezig is en zich hetzelfde gedraagt als een Common Spanning Tree (CST). Daarnaast kunnen Multiple Spanning Tree Instances (MSTI's) geconfigureerd worden welke alleen binnen de regio bestaan. In tegenstelling tot MSTI's, kan een IST wel communiceren met (R)STP-switches buiten de regio. In de regio worden BPDU's niet per VLAN (zoals bij PVST+) of per instantie (zoals bij MISTP) verstuurd, maar als een enkele BPDU welke alle informatie van de verschillende instantie's (Mrecords) bevat. Deze type 2, versie 3 BPDU's worden op elke link port verzonden, zowel op de designated poort als de root poort.

Het koppelen van meerdere regio's, regio's met STP of (R)PVST+ switches, vereist een grondige kennis van MST en (R)STP.

RSTP kan op Cisco-switches op twee manieren gerealiseerd worden:

- Rapid PVST+ (RSTP geïmplementeerd in PVST+)
- MST (RSTP geïmplementeerd in de opvolger van MISTP)

4.2 Mogelijke oplossingen

- Door voor de huidige situatie de timers van STP aan te passen, kan met weinig moeite een reductie van de convergentietijd gerealiseerd worden.
- Er kan gekeken naar de *fast verbeteringen die door Cisco zijn aangebracht op het 802.1d protocol. Deze verbeteringen zijn echter Cisco proprietary. Het streven is om zo veel mogelijk open standaarden te gebruiken. Verder moeten deze verbeteringen per switch (of zelfs per poort) geconfigureerd worden en vergt daardoor een hogere beheerslast.
- In RSTP zijn de verbeteringen van Cisco opgenomen en verder geoptimaliseerd. Cisco adviseert RSTP te gebruiken boven de (oudere) verbeteringen. RSTP is een standaard en kan eenvoudig worden geconfigureerd.

4.3 Spanning Tree metingen in het HvU communicatie lab

Om een beter inzicht te krijgen in de convergentietijden van het Spanning Tree Protocol zijn een aantal metingen verricht. Tijdens deze metingen worden het traditionele Spaning Tree Protocol (802.1d) en het nieuwere Rapid Spaning Tree Protocol (802.1w) onderzocht op convergentietijd.

Om de convergentietijden van spanning tree te meten zijn een aantal metingen gedaan. Deze metingen zijn in drie delen opgesplitst.

1. Meting 1: STP met default parameters
2. Meting 2,3 en 4: STP met aangepaste parameters
3. Meting 5: RSTP met default parameters

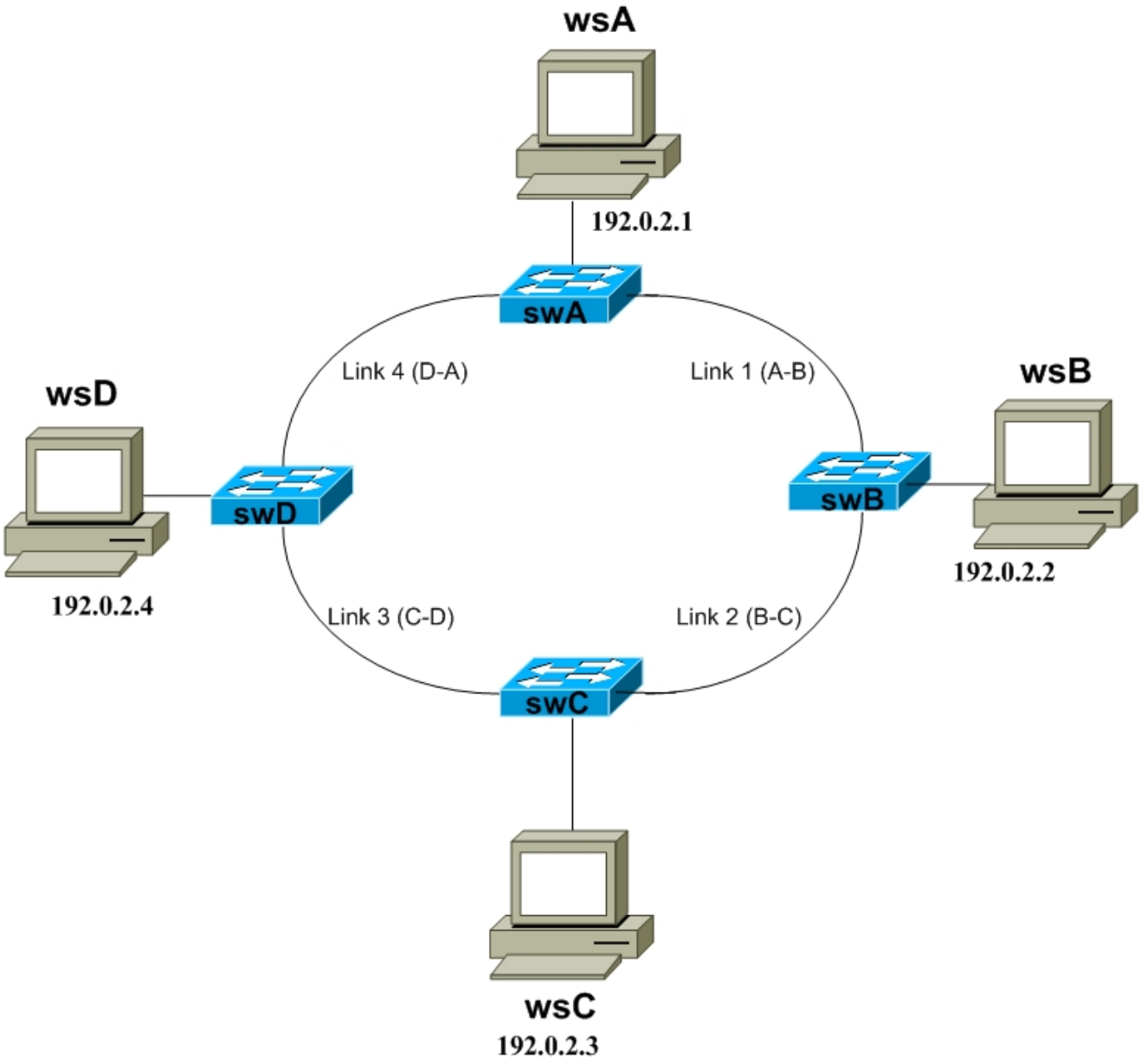
De test opstelling is weergegeven in fig. 4.2.

4.3.1 Hulpmiddelen

De switches welke zijn gebruikt in deze opstelling zijn Cisco 3550 switches. De onderlinge verbindingen zijn fast ethernet. SwA is root bridge (laagste mac adres). Vanaf wsA wordt fping (www.fping.com) gebruikt om te testen hoelang het duurt voordat de werkstations weer bereikbaar zijn na een gesimuleerde kabelbreuk. Alle metingen worden dus gedaan tov. WsA.

Fping wordt met de volgende parameters gestart:

```
fping -c 60 -g 192.0.2.1 192.0.2.4 -s
```



Figuur 4.2: *Meet opstelling*

De output ziet er alsvolgt uit:

```
192.0.2.1: xmt/rcv/%loss = 60/60/0%, min/avg/max = 0.54/0.67/3.42
192.0.2.2: xmt/rcv/%loss = 60/0/100%
192.0.2.3: xmt/rcv/%loss = 60/60/0%, min/avg/max = 0.99/1.10/1.34
192.0.2.4: xmt/rcv/%loss = 60/60/0%, min/avg/max = 0.98/1.08/2.26
```

Op deze manier kan eenvoudig achterhaald worden hoelang het duurt voordat een link weer aktief is.

4.3.2 Stap voor stap

Per meting worden om de beurt verschillende links onderbroken. In de eerste meting wordt als eerste bekeken wat de topologie is (welke port is blocking). Vervolgens wordt fping gestart, nadat de eerste replies zijn ontvangen wordt link1 onderbroken. Er wordt gemeten wat de tijd is voor dat alle werkstations weer bereikbaar zijn. Daarna zal de de orginele topology weer worden hersteld (kabel er weer in). Wederom wordt er een nieuwe fping gestart en er zal weer gemeten worden hoelang het duurt voordat alles weer gestabiliseerd is. Deze meting dient voor iedere link te worden herhaald.

Meting 1

In deze meting worden de switches geconfigureerd met STP met de default waarden.

Meting 2,3 en 4

In deze meting wordt wederom STP gebruikt. Echter nu met aangepaste parameters (timers).

Meting 5

In deze meting wordt RSTP gebruikt door MST te configureren.

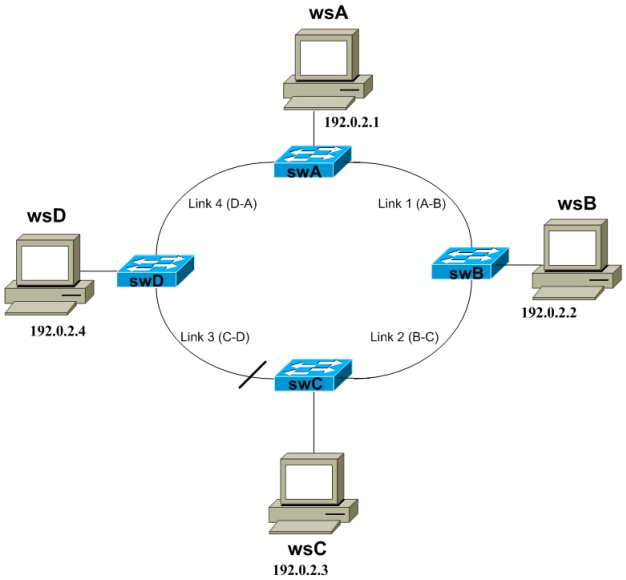
Resultaten

De metingen zijn met succes uitgevoerd in het communicatie lab van de HvU. Hieronder zullen de resultaten worden besproken.

4.3.3 Meting 1

In deze meting werd er niets aan de default STP parameters gewijzigd. Dit betekent dat de hello-time op 2sec staat, de forward delay op 15 sec en de max age time op 20sec. Switch A was steeds de root bridge in deze meting. De poort op swC naar swD was steeds in blocking mode in het geval dat de ring gesloten was. zie ook 4.3.

De tijden welke in tabel 4.2 staan weergegeven, representeren de tijd in seconden dat het duurde voordat het betreffende werkstation weer bereikbaar was. Een ”x”betekent dat de onderbreking



Figuur 4.3: *Meet opstelling, swC is blocking op link c-d*

802.1d	wsA	wsB	wsC	wsD
link 1 (a-b) (broken link)	x	47sec	47sec	x
link 1 (a-b) (restore link)	x	28sec	29sec	x
link 2 (b-c) (broken link)	x	x	30sec	x
link 2 (b-c) (restore link)	x	x	28sec	x
link 3 (c-d) (broken link)	x	x	x	x
link 3 (c-d) (restore link)	x	x	x	x
link 4 (d-a) (broken link)	x	x	x	49sec
link 4 (d-a) (restore link)	x	x	x	30sec

Tabel 4.1: *resultaten stp test 1*

geen invloed had op de bereikbaarheid van dat werkstation. Te zien is dat de convergentietijd tussen de 30 seconden (2 keer de forward-delay) en de 50 seconden (2 keer de forward-delay + max-age) ligt. Deze tijden zijn vergelijkbaar met de huidige situatie op de SURFnet GigaMAN netwerken.

4.3.4 Meting 2

In de tweede meting zijn een aantal STP (802.1d) parameters aangepast met het volgende commando op switch A (de root bridge):

```
swA# configure terminal
swA(config)# spanning-tree vlan 1 root primary diameter 4
swA(config)# end
```

Dit is een macro in het Cisco IOS waarbij het maximaal aantal 'hops' dat een tree kan omvatten als diameter-parameter (4 in dit geval) wordt meegegeven. De timers werden daardoor automatisch alsvolgt aangepast:

- max-age: 14 seconden
- Hello: 2 seconden (default)
- Forward-delay: 10 seconden

De blocking poort tijdens een gesloten ring zat weer op de link tussen C en D op swC. De resultaten zijn te zien in tabel 4.2:

802.1d (2)	**wsA**	**wsB**	**wsC**	**wsD**
link 1 (a-b) (broken link)	**x**	**33sec**	**33sec**	**x**
link 1 (a-b) (restore link)	**x**	**19sec**	**19sec**	**x**
link 2 (b-c) (broken link)	**x**	**x**	**20sec**	**x**
link 2 (b-c) (restore link)	**x**	**x**	**19sec**	**x**
link 3 (c-d) (broken link)	**x**	**x**	**x**	**x**
link 3 (c-d) (restore link)	**x**	**x**	**x**	**x**
link 4 (d-a) (broken link)	**x**	**x**	**x**	**32sec**
link 4 (d-a) (restore link)	**x**	**x**	**x**	**20sec**

Tabel 4.2: *resultaten stp test 2*

Te zien is dat de convergentietijd tussen de 19 seconden (2 keer de forward-delay, zou dus maximaal 20 seconden kunnen zijn) en de 33 seconden (2 keer de forward-delay + max-age, zou dus maximaal 38 seconden kunnen worden)ligt.

4.3.5 Meting 3

In de derde meting zijn wederom een aantal STP (802.1d) parameters aangepast. Dit is gedaan met het volgende commando op switch A (de root bridge):

```
swA# configure terminal
```

```
swA(config)# spanning-tree vlan 1 root primary diameter 4 helo-time 1
swA(config)# end
```

met als gevolg dat de volgende timers alsvolgt werden aangepast:

- max-age: 8 seconden
- Hello: 1 seconde
- Forward-delay: 6 seconden

De blocking poort tijdens een gesloten ring zat weer op de link tussen C en D op swC. De resultaten zijn te zien in tabel 4.3:

802.1d (3)	**wsA**	**wsB**	**wsC**	**wsD**
link 1 (a-b) (broken link)	**x**	**19sec**	**19sec**	**x**
link 1 (a-b) (restore link)	**x**	**12sec**	**12sec**	**x**
link 2 (b-c) (broken link)	**x**	**x**	**12sec**	**x**
link 2 (b-c) (restore link)	**x**	**x**	**11sec**	**x**
link 3 (c-d) (broken link)	**x**	**x**	**x**	**x**
link 3 (c-d) (restore link)	**x**	**x**	**x**	**x**
link 4 (d-a) (broken link)	**x**	**x**	**x**	**18sec**
link 4 (d-a) (restore link)	**x**	**x**	**x**	**12sec**

Tabel 4.3: *resultaten stp test 3*

Te zien is dat de convergentietijd tussen de 12 seconden (2 keer de forward-delay, 2x6sec) en de 19 seconden (2 keer de forward-delay + max-age, zou dus maximaal 20 seconden kunnen worden)ligt.

4.3.6 Meting 4

In de vierde meting zijn wederom een aantal STP (802.1d) parameters aangepast. Echter dit keer niet met de 'veilige' macro-functie maar zijn alle timers handmatig ingesteld. De timers zijn alsvolgt ingesteld:

- max-age: 6 seconden
- Hello: 1 seconde
- Forward-delay: 4 seconden

Dit zijn de kleinste waarden die ingesteld kunnen worden op het Cisco IOS. Theoretisch wordt op deze manier de snelst haalbare convergentie met het 802.1d protocol gerealiseerd. De resultaten zijn te zien in tabel 4.4:

De snelst haalbare convergentietijd van het spanning-tree protocol (zonder Cisco enhancements) ligt dus tussen de 8 seconden (2 keer de forward-delay, 2x4sec) en de 13 seconden (2 keer de forward-delay + max-age, zou dus maximaal 14 seconden kunnen worden). Wij hebben in bovenstaande meting ook een keer 7 seconden gemeten, dit is waarschijnlijk het gevolg van een meet afwijking.

802.1d (4)	wsA	wsB	wsC	wsD
link 1 (a-b) (broken link)	x	13sec	13sec	x
link 1 (a-b) (restore link)	x	8sec	8sec	x
link 2 (b-c) (broken link)	x	x	8sec	x
link 2 (b-c) (restore link)	x	x	7sec	x
link 3 (c-d) (broken link)	x	x	x	x
link 3 (c-d) (restore link)	x	x	x	x
link 4 (d-a) (broken link)	x	x	x	13sec
link 4 (d-a) (restore link)	x	x	x	7sec

Tabel 4.4: *resultaten stp test 4*

4.3.7 Meting 5

In de vijfde meting is dezelfde topology gebruikt, echter nu zijn de switches geconfigureerd met RSTP. Door de oudere versies van de IOS-software, werd rapid pvst+ niet door alle switches ondersteund. Daarom is voor deze meting MST gebruikt. Dit is alsvolgt op de IOS switches geconfigureerd :

```
spanning-tree mode mst
spanning-tree mst configuration
 name SURF
 revision 1
!op de edge poort (waar het werkstation mee verbonden is)
 interface FastEthernet0/3
 spanning-tree portfast
```

Er zijn geen timers aangepast, dus er wordt gebruik gemaakt van de default timers. De resultaten van de convergentietijden van 802.1w zijn te zien in tabel 4.5:

802.1w (5)	wsA	wsB	wsC	wsD
link 1 (a-b) (broken link)	x	48ms	80ms	x
link 1 (a-b) (restore link)	x	16msec	16ms	x
link 2 (b-c) (broken link)	x	x	48ms	x
link 2 (b-c) (restore link)	x	x	16ms	x
link 3 (c-d) (broken link)	x	x	x	x
link 3 (c-d) (restore link)	x	x	x	x
link 4 (d-a) (broken link)	x	x	x	96ms
link 4 (d-a) (restore link)	x	x	x	0ms

Tabel 4.5: *resultaten RSTP*

Zoals te zien is, zijn de convergentietijden welke met behulp van RSTP bereikt kunnen worden aanzienlijk beter. Om tot in milli-seconden te kunnen meten is de meetmethode voor deze meting wat aangepast. De bereikbaarheid van iedere host is nu afzonderlijk getest met behulp van fping.

```
/fping -c 1000 -p 1 -i1 -t 1 192.0.2.2
```

Alle fping variabelen zijn geconfigureerd op 1ms. Er werden tijdens de meting 1000 pakketten verzonden binnen 16 seconden. Daarvan gingen tijdens een meting 3 pakketten verloren (geen icmp echo reply ontvangen). Daaruit volgt dat de tijd dat er geen verbinding was naar de betreffende host bijvoorbeeld 48 ms is:

$$\frac{16sec}{1000packets} \times 3 = 48ms$$

Aanvullingen op de RSTP meting

Er is ook getest wat het gevolg was van het verdwijnen van de root bridge:

$$\frac{16sec}{1000packets} \times 77 = 1,2seconde$$

Daarna is de root bridge weer terug gezet in het netwerk, deze werd onmiddelijk weer root bridge. De tijd die dit koste was minimaal:

$$\frac{16sec}{1000packets} \times 4 = 64ms$$

Alle metingen zijn uitgevoerd met Cisco 3550smi switches. Later is één van de 3550smi switches vervangen door een Catalyst-switch met CatOS 7.5. Opvallend was dat de convergentietijd ineens een stuk hoger lag en erg per keer varieerde. De tijd dat (een gedeelte van de ring) niet beschikbaar was liep op tot:

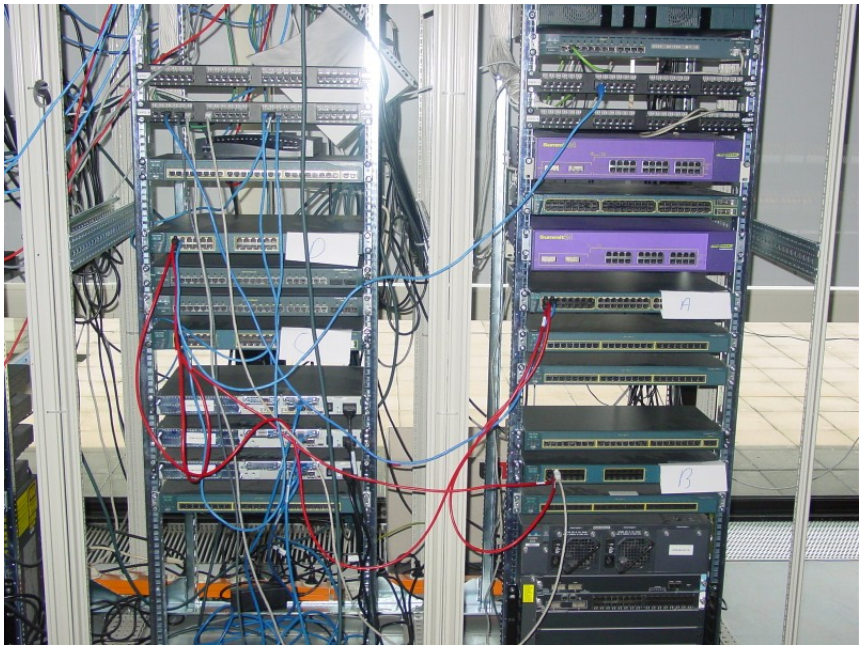
$$\frac{16sec}{1000packets} \times 60 = 1seconde$$

4.4 Gekozen technologie

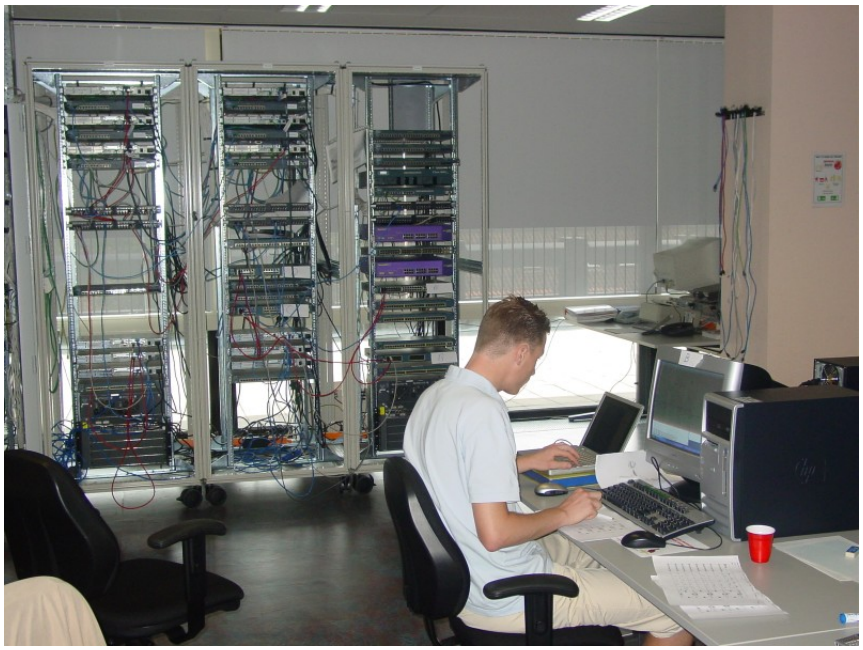
Het veranderen van de Spanning Tree (802.1d) timers heeft een voorspelbare invloed op de convergentietijd van het netwerk. Echter de door ons gebruikte switches (met IOS), konden niet geconfigureerd worden met de timers korter dan zoals die in meting 4 zijn gebruikt. Dit betekent dat Spanning Tree niet sneller kan convergeren dan 8 tot 14 seconden. Dit is al een hele verbetering ten opzichte van het gebruik van de default parameters. Hierbij ligt de convergentietijd tussen de 30 en 50 seconden, dit zijn ook de waarden zoals die nu binnen GigaMAN worden toegespast.

Duidelijk is geworden uit test 5, dat de resultaten met rapid spanning tree (802.1w) aanzienlijk beter zijn. De convergentietijd van het netwerk wordt terug gebracht van enkele (tientallen) seconden naar enkele (tientallen) milli-seconden.

Geconcludeerd kan worden dat wanneer de huidige configuratie van de GigaMAN's aangepast zal worden, het beste gekozen kan worden voor de implementatie van het Rapid Spanning Tree Protocol (RSTP). De convergentietijd die hiermee bereikt kan worden is aanzienlijk beter. Aan het aanpassen van de timers van het 802.1d protocol zit nog een nadeel. Volgens diverse bronnen op het Internet is de kans groot dat de stabiliteit van het netwerk hiermee omlaag 'kan' gaan terwijl er juist getracht wordt om dit te verbeteren. RSTP heeft zich in de praktijk al bewezen en verdient dan ook de voorkeur boven alternatieven zoals het aanpassen van timers of het gebruik van de Cisco proprietary enhancements in een 802.1d omgeving.



Figuur 4.4: *Testopstelling meting in het HvU communicatie lab*



Figuur 4.5: *Totaalbeeld meting in het HvU communicatie lab*

4.5 RSTP testen in het SURFnet testnetwerk

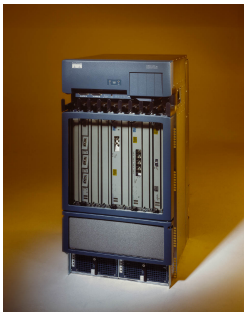
Uit de vorige metingen is gebleken dat Rapid Spanning Tree absoluut de voorkeur verdient wat betreft de convergetietijd. Na de metingen in het HvU communicatie lab, zijn er RSTP metingen gedaan op het testnetwerk van SURFnet bij SARA. Hierop is een GigaMAN ring nagebouwd. Het doel van deze test was om te bepalen of met deze apparatuur dezelfde convergentietijden konden worden gehaald.

4.5.1 Architectuur test opstelling

Voor het nabouwen van een GigaMAN netwerk, wordt gebruik gemaakt van de test apparatuur van SURFnet. Dit netwerk bevat dezelfde switches als het SURFnet5 netwerk. De switches welke gebruikt zijn om de ring te creëren (ts1 t/m ts3) zijn Cisco Catalyst 4912 switches, zie figuur4.6. Deze switches draaien CatOS 7.6(6) als OS.

Figuur 4.6: *cisco-WS-C4912G*

Als POP-router wordt test-router 11 (tr11) gebruikt. Dit is een Cisco 12410 GSR. Deze routers worden ook in het SURFnet5 netwerk op de POP-locatie gebruikt, zie figuur 4.7.



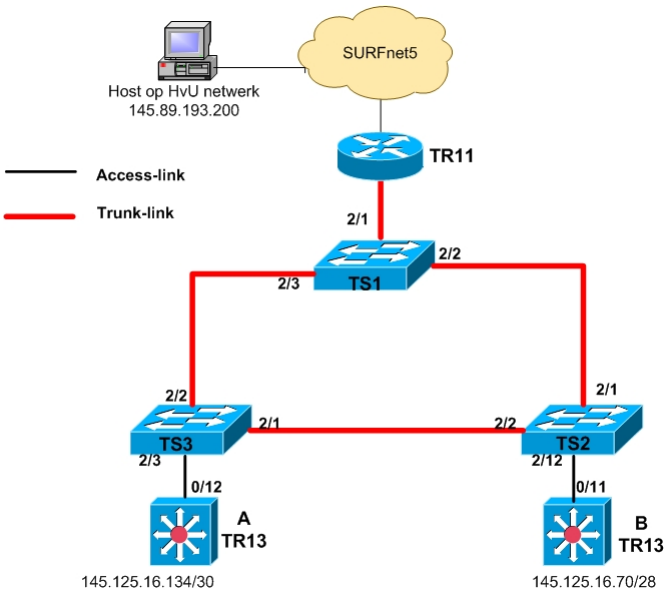
Figuur 4.7: *GSR-12410*

De klant routers worden gesimuleerd met behulp van een Cisco 3550emi. Op deze Layer3 switch worden 2 poorten geconfigureerd als 'router poort'. Deze krijgen een IP adres en gedragen zich verder niet als switch. Beide poorten gedragen zich als een host (A en B), ieder in een ander VLAN (VLAN 1 en 5).

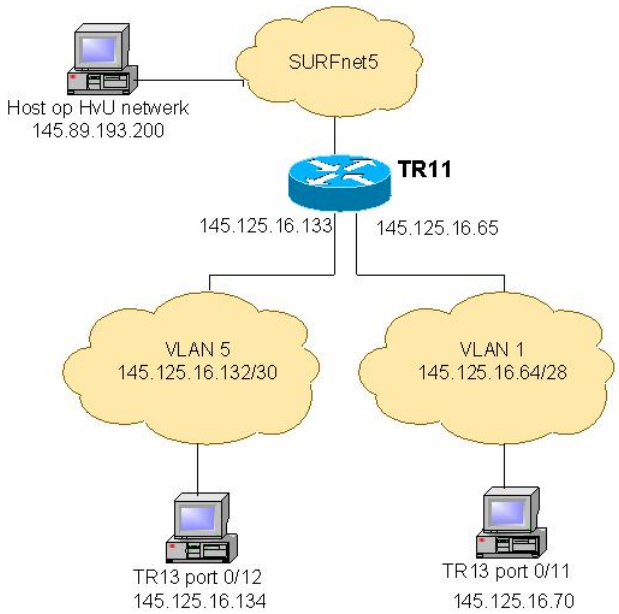
De test opstelling ziet eruit zoals in figuur 4.8

Net als in de huidige situatie heeft iedere klant een eigen VLAN. Deze VLANs dienen dus bekend te zijn op de switches TS1 t/m TS3. TR11 heeft een q-tagged verbinding met TS1, hetzelfde geldt voor de TS switches onderling.

De laag 3 topologie is weergegeven in figuur 4.9



Figuur 4.8: *testopstelling*



Figuur 4.9: *testopstelling-l3*

4.5.2 Configuratie test opstelling

De switches TS1 t/m TS3 zijn geconfigureerd om Rapid Spanning Tree te gebruiken voor loop-prevention. Er is allereerst gemeten door rapid-pvst+ te configureren. De benodigde configuratie hiervoor is alsvolgt:

```
TS1
! create qtaged trunk
set trunk 2/1  on dot1q 1-1005,1025-4094
set trunk 2/2  on dot1q 1-1005,1025-4094
set trunk 2/3  on dot1q 1-1005,1025-4094
!enable rapid-pvst
set spantree mode rapid-pvst+
! maak TS1 voor vlan 1 en 5 de root-bridge
set spantree priority 4096   1
set spantree priority 4096   5
! uplink naar TR11 is een edge port.
set spantree portfast    2/1 enable trunk
!
```

```
TS2
! create qtaged trunk
set trunk 2/1  on dot1q 1-1005,1025-4094
set trunk 2/2  on dot1q 1-1005,1025-4094
! configureer de klant-poort op vlan 1
set vlan 1    2/12
!enable rapid-pvst
set spantree mode rapid-pvst+
! link naar TR13 is een edge port.
set spantree portfast    2/12 enable
!
```

```
TS3
! create qtaged trunk
set trunk 2/1  on dot1q 1-1005,1025-4094
set trunk 2/2  on dot1q 1-1005,1025-4094
! configureer de klant-poort op vlan 5
set vlan 5    2/3
!enable rapid-pvst
set spantree mode rapid-pvst+
! link naar TR13 is een edge port.
set spantree portfast    2/3 enable
```

Voor het aanmaken van de VLANs is VTP gebruikt, dit is niet in bovenstaande configuratie opgenomen.

4.5.3 Test resultaten per VLAN rapid spanning tree

Om de convergentietijd van de ring te testen, wordt fping gebruikt[1]. Vanaf een host op SURFnet5 (145.89.193.200) wordt met behulp van fping getest hoelang het duurt voordat de interfaces van TR13 (145.125.16.70 of 145.125.16.134) weer bereikbaar zijn nadat er een link is onderbroken. Alle metingen worden dus gedaan tov. 145.89.193.200.

```
fping -c 1500 -e -i 20 -p 1 -r 1 -s -t 20 145.125.16.70
de output ziet er alsvolgt uit:
145.125.16.70 : xmt/rcv/%loss = 1500/1465/97%, min/avg/max = 1.40/1.58/19.5
```

Op deze manier kan eenvoudig achterhaald worden hoelang het duurt voordat een link weer aktief is. Iedere ping duurt 23ms, dus wanneer er 35 packets verloren gaan betekent dit een convergentietijd van ongeveer 805ms.

De test resultaten van de meting zoals die is gedaan met de test apparatuur van SURFnet staan in tabel 4.6

802.1w	**host A**	**host B**
Link TS3-TS1 down (2/2 op TS3 shut)	0,92 sec	X
Link TS3-TS1 up (2/2 op TS3 no shut)	0,49 sec	X
Link TS1-TS2 down (2/2 op TS1 shut)	X	0,81 sec
Link TS1-TS2 up (2/2 op TS1 no shut)	x	0,49 sec

Tabel 4.6: *resultaten RSTP test in test netwerk van SURFnet*

De resultaten zoals die in tabel4.6 staan, zijn gemiddelde van 4 metingen. In deze meting waren er op de switches twee VLANs geconfigureerd en er hoefde dus ook maar twee keer een spanning tree te worden berekend. Later is dezelfde test met meerdere VLANs gedaan. Hieruit bleek dat de convergentietijd daarmee langer werd. Bijvoorbeeld voor meting 3, waarbij link TS1-TS2 down werd gebracht. Met 2 VLANs was hierbij de convergentietijd gemiddeld 0,81 sec. Dezelfde meting met 10 VLANs duurde het 1.26 sec. Dit is verklaren door het feit dat het spanning tree algoritme voor meerdere VLANs moet worden berekend.

4.5.4 Test resultaten Multiple Spanning Tree

In de testen is ook gekeken naar verschillen in convergentietijd tussen 'per vlan (rapid) spanning tree' en 'Multiple (rapid) Spanning Tree'. De test switches (ts1 t/m ts3) zijn voor deze meting alsvolgt geconfigureerd:

```
set spantree mode mst
set spantree MST config name SURF
set spantree MST config revision 1
!optioneel, VLANs aan een instantie koppellen.
set spantree MST 0 vlan 1-100
!port fast voor de juiste poorten aanzetten.
```

[1] www.fping.com

```
set spantree portfast 2/12 enable
set spantree mst config commit
```

In een situatie met bovenstaande configuratie, wordt er slechts één spanning tree voor alle VLANs berekend. Deze ziet er dus voor alle VLANs het zelfde uit. Theoretisch gezien, zou deze configuratie sneller kunnen zijn, doordat er door de switches slechts één spanning tree berekend hoeft te worden. De meetmethode is wederom hetzelfde, met behulp van fping wordt gemeten hoeveel pakketten er verloren gaan. De resultaten staan weergegeven in tabel4.7

802.1w	**host A**	**host B**
Link TS3-TS1 down (2/2 op TS3 shut)	0,80sec	X
Link TS3-TS1 up (2/2 op TS3 no shut)	0,58sec	X

Tabel 4.7: *Resulaten multiple instance rapid spanning tree test in test netwerk van SURFnet*

De resultaten van deze meting zijn, vergeleken met per VLAN spanning tree (met 2 VLANs) ongeveer gelijk. Dit verschil zal echter groter worden bij het gebruik van meerdere VLANs.

4.5.5 Conclusie van de RSTP-metingen

In de test omgeving van SURFnet, is met soortgelijke apparatuur als in SURFnet5 een GigaMAN ring nagebouwd. De gemiddelde convergentietijd van het netwerk ligt tussen de 0.5 en 1 seconde. Dit zijn, zeker vergeleken met de huidige situatie, zeer snelle tijden. De verschillen tussen MST en rapid-pvst zijn minimaal. De verschillen zullen echter oplopen al naar gelang er meer VLANs zijn en er dus meer spanning-tree instanties berekend moeten worden. Uit de ervaring die is opgedaan door de meting, is gebleken dat het belangrijk is om de klant-aansluiting te definiëren als edge port (portfast). Wanneer dit niet wordt gedaan, zal in het geval van een link-failure de tijd die het duurt voordat de klantpoort in forwarding mode komt 2 maal de forward delay zijn. Alle voordelen die RSTP biedt, zijn dan voor de klant aansluiting teniet gedaan.
Een opvallend gegeven is dat wanneer de gegevens van de RSTP meting op het SURFnet test netwerk worden vergeleken met de RSTP metingen die zijn gedaan in het HvU communicatie lab, er toch een vrij groot verschil in convergentietijden is (tientallen milli-seconden vs honderden milli-seconden). Dit blijkt na nader onderzoek te worden verzoorzaakt door de gebruikte apparatuur. In het lab op de HvU is gebruik gemaakt van 3550smi switches waarop IOS draait. De Switches in het SURFnet netwerk zijn Catalyst 4912 switches welke CatOS draaien. Het lijkt er op dat de switches welke CatOS draaien langzamer zijn met het convergeren dan de switches die IOS draaien. Dit werd ook meteen duidelijk toen bij een latere meting in het HvU lab, één van de IOS switches is vervangen door een CatOS switch. Meteen liep de convergentietijd op naar een halve seconde. Of dit ligt aan het gebruikte Operating System of de hardware is onduidelijk[2].

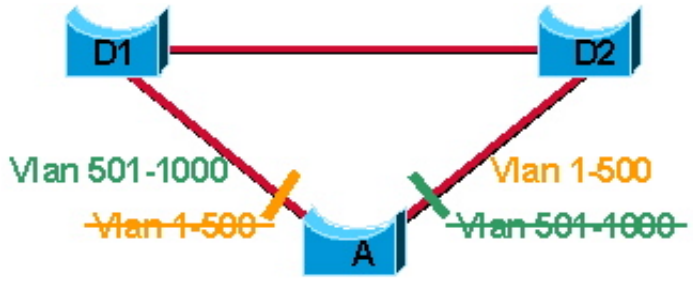
Rapid-pvst+ verdient de voorkeur boven MST omdat het configureren hiervan eenvoudig is. Vergeleken met de huidige situatie, waar ook per VLAN spanning tree wordt gebruikt, zijn de wijzigingen minimaal waardoor het beheer ervan relatief eenvoudig blijft: Keep It Simple (KIS).

[2]De gebruikte Catalyst switches zijn gebouwd met oudere hardware dan de IOS switches

4.6 Aanbevelingen voor implementatie

4.6.1 Loadbalancing in een ring

Voor loadbalancing wordt er in de praktijk veel gebruik gemaakt van de Spanning Tree cost op de poorten. Zoals een Spanning Tree met access-link-ports (een link met één VLAN) wordt berekend op basis van de port-cost welke standaard is afgeleid van de bandbreedte van de betreffende poort, wordt een Spanning Tree met trunk-ports berekend op basis van de VLAN-cost welke standaard eveneens afgeleid is van de bandbreedte van de betreffende trunk-port.



Figuur 4.10: *Loadbalancing in (R)STP*

Door de VLAN-cost te manipuleren, kunnen de VLAN's over verschillende trunk-ports verdeeld worden (zie figuur 4.10). In plaats van een Common Spanning Tree voor alle VLAN's waarbij de verbinding A-D1 niet gebruikt wordt omdat deze blocked is, kan met rapid-pvst+ en MST gezorgd worden dat alleen voor VLAN's 1-500 de verbinding A-D1 blocked is en voor de VLAN's 501-1000 de verbinding A-D2 dit is. Zo wordt loadbalancing gerealiseerd door het gebruik van beide verbindingen door verschillende VLAN's.

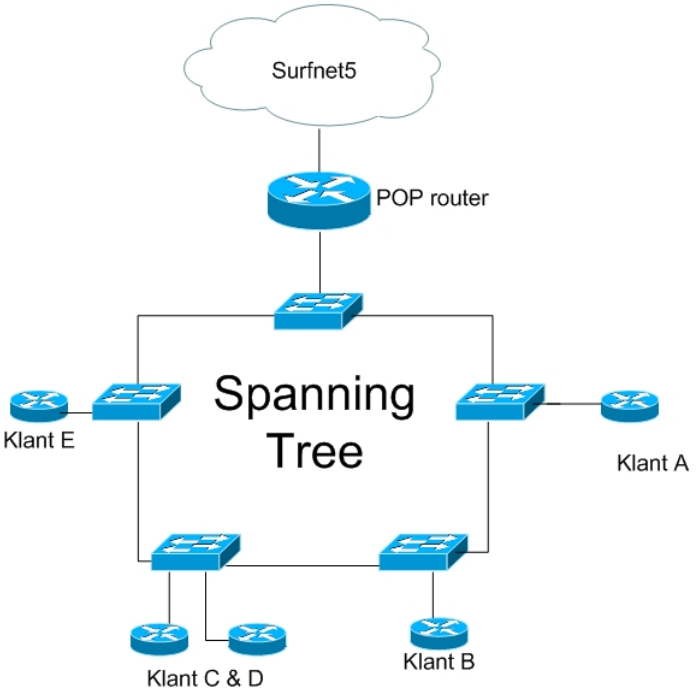
Door meerdere klanten in een ring op te nemen (zie figuur 4.11) welke (ongeveer) evenveel data-verkeer genereren, wordt load-balancing vanzelf gerealiseerd. De klanten C,D en E krijgen voor een optimale loadbalancing linksom een pad naar de POP-router en de klanten A en B rechtsom. De verbinding tussen switch B en C moet dus geblokt worden. Door van de switch op de POP-locatie een root-switch te maken, wordt deze situatie vanzelf bereikt en zal de verbinding tussen switch B en C geblokt worden. VLAN-cost waarden hoeven niet gewijzigd te worden omdat de verbindingen allemaal van dezelfde bandbreedte zijn.

4.6.2 Stabiliteit

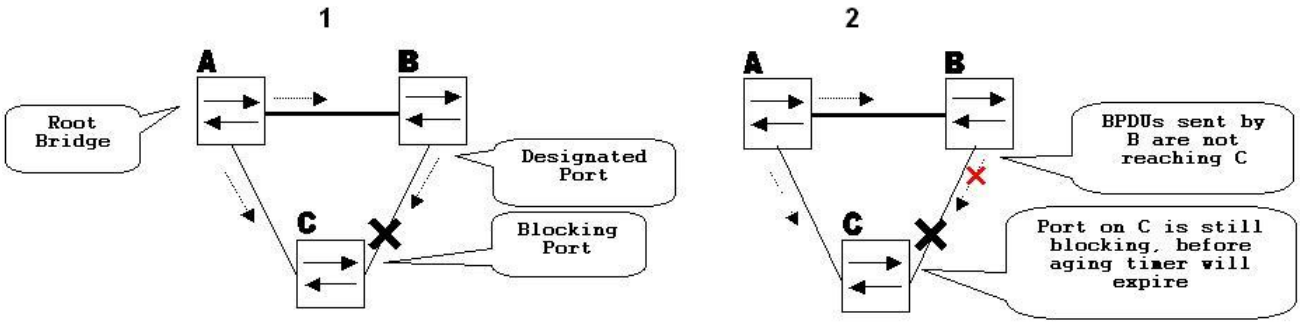
Enkelvoudige loops

Voor een correcte werking van (R)STP is het van belang dat alle link-poorten bi-directioneel zijn. STP gaat er vanuit dat op een link BPDU's zowel verzonden als ontvangen kunnen worden. Dit gebeurt met een frequentie van de hello-time. Als een link uni-directioneel is, kan er ondanks STP alsnog een loop ontstaan.

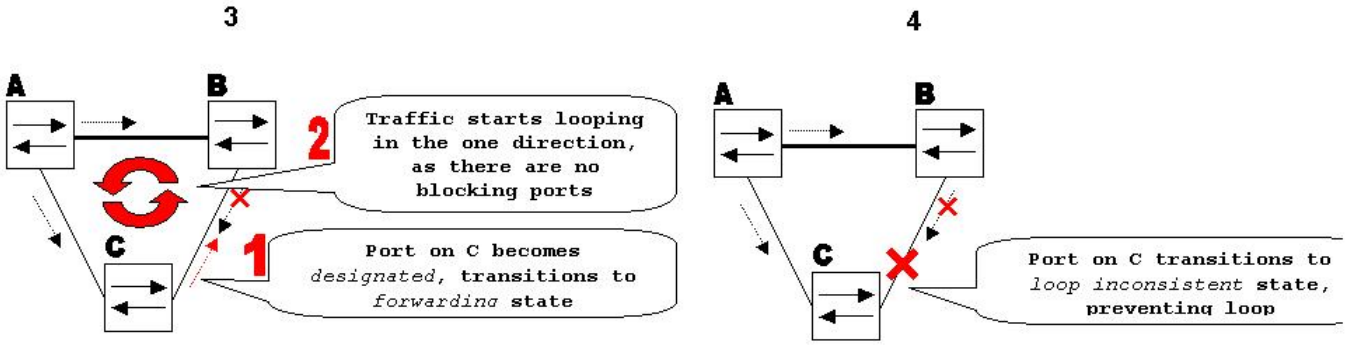
In figuur 4.12 en 4.13 zijn de consequenties van een uni-directionele link geschetst. Een switch C heeft op een segment een poort blocked omdat een andere switch B op hetzelfde segment BPDU's zend met een lagere path cost naar de root-switch. Switch B is dus de designated switch voor



Figuur 4.11: *GigaMAN-ring*



Figuur 4.12: *UDLD1*



Figuur 4.13: *UDLD2*

dat segment (zie figuur 4.12.1). Als de link tussen switch C en B door een failure uni-directioneel wordt waarbij switch C geen BPDU's meer ontvangt van switch B maar er wel data verzonden kan worden van switch C naar B, gaat switch C er vanuit dat switch B onbereikbaar is geworden op dat segment (zie figuur 4.12.2). Switch C maakt dan van de blocked poort een designated poort in forwarding mode. Omdat de designated poort op switch B ook nog steeds in forwarding is, is er een enkelvoudig loop gecrëerd via switch C naar B naar A (zie figuur 4.13.3). Vooral bij het gebruik van fiber is de kans op uni-directionele verbindingen groot door het falen van één van de twee vezels.

UDLD

Enkelvoudige loops kunnen met enhancements voorkomen worden, zoals met het Cisco's UniDirectional Link Detection protocol (UDLD). Dit echo-gebaseerde protocol dat op beide poorten van een link geconfigureerd moet worden, test of de poort aan de andere zijde ook echt een verstuurd bericht als echo terugstuurt. Hiermee wordt voorkomen dat een poort welke aangeeft een physical link (laag 1 OSI-model) te hebben en zo door STP als 'up' wordt gezien, een uni-directionele verbinding op laag 2 heeft. UDLD breng standaard een uni-directionele verbinding in de disabled mode.

Met de UDLD in aggressive mode kan 'traffic blackholing' voorkomen worden waarbij UDLD een verbinding welke fysiek 'up' is, maar waar zowel ontvangen als verzenden van data niet mogelijk is, in disabled mode wordt gebracht. Het ontbreken van een laag 2 verbinding terwijl de fysieke verbinding wel 'up' is, zijn verbindingen welke op laag 1 niet point-to-point zijn. Het fysiek down brengen van een poort heeft geen 'not connected' aan de andere zijde tot gevolg. Voorbeelden hiervan zijn het gebruik van mediaconverters of het gebruik van meerdere hubs op een point-to-point verbinding. Als tussen twee hubs een verbinding wegvalt, blijft voor de andere hubs de poorten fysiek up terwijl er geen verbinding op laag 2 meer is.

De UDLD message interval staat default op 15 seconden en is terug te brengen tot het minimum van 7 seconden. Na een 'age out' van 3 maal de interval tijd, zal de poort in disable mode worden gebracht. Als RSTP gebruikt wordt, zal door de port fast aging na het ontbreken van drie hello-BPDU's van de betreffende link een andere poort in forwarding worden gebracht. Dit is sneller dan de detectie door UDLD zodat de aggressive mode in combinatie met RSTP geen toegevoegde waarde heeft.

Voorbeeld van het activeren van UDLD in standard mode voor port 2/1 en 2/2:

```
ts1.amsterdam1> (enable) set udld enable
UDLD enabled globally


ts1.amsterdam1> (enable) set udld enable 2/1-2
UDLD enabled on port 2/2.
Warning: UniDirectional Link Detection should be enabled on all
 the ends of the connection in order to work properly.


ts1.amsterdam1> (enable) sh port
* = Configured MAC Address

Port  Name                 Status     Vlan       Level  Duplex Speed Type
----- ------------------ ---------- ---------- ------ ------ ----- ------------
 2/1  tr11.amsterdam1    connected  trunk      normal   full  1000 1000BaseSX
 2/2  ts2.amsterdam1 2/1 connected  trunk      normal   full  1000 1000BaseSX
 2/3  ts3.amsterdam1     connected  trunk      normal   full  1000 1000BaseSX


ts1.amsterdam1> (enable) sh udld port
UDLD              : enabled
Message Interval  : 15 seconds
Port      Admin Status  Aggressive Mode  Link State
--------  ------------  ---------------  ----------------
 2/1      enabled       disabled         undetermined
 2/2      enabled       disabled         bidirectional
 2/3      disabled      disabled         not applicable
```

Te zien is dat op aan de andere zijde van poort 2/1 geen UDLD geconfigureerd is en op poort 2/2 wel. Poort 2/1 zal niet down gaan als UDLD slecht aan één kant geconfigureerd is.

Loop Guard

Een andere enhancements om een enkelvoudige loop te voorkomen is Cisco's Loop Guard. Hiermee wordt in (R)STP voorkomen dat een blocked poort welke fysiek up is, en geen BPDU's meer ontvangt, een designated poort in forwarding mode wordt. In plaats daarvan komt de poort in een loop-inconsistent state. Op het moment dat de link weer gerepareerd is en de poort weer BPDU's ontvangt, gaat de poort weer terug naar de normale blocked mode. Een voorwaarde voor Loop Guard is dat de verbinding niet shared is. Het segment moet dus bestaan uit een point-to-point verbinding met twee poorten. Verder werkt Loop Guard alleen als op een blocked poort reeds BPDU's zijn ontvangen voordat de verbinding uni-directioneel wordt. Op het moment dat een uni-directionele verbinding van 'down' wordt 'up' gebracht, zal de betreffende poort geen BPDU's ontvangen en alsnog als designated in forwarding mode komen.

Port-negotiation

Er is getest met het onderbreken van fiber-verbindingen welke zonder mediaconversie de switches onderling verbindt. Deze directe verbindingen worden ook in de GigaMAN ringen gebruikt. Uit bevindingen is gebleken dat het onderbreken van één vezel van een fiberpaar beide poorten 'not connected' en dus down zijn als op beide poorten negotiation aan staat. Als dit uit wordt gezet, gaat alleen de poort met de onderbroken fiber van het ontvangende signaal (Rx) down en blijft de poort met de onderbroken fiber van het zendende signaal (Tx) up. In beide gevallen gaat er minimaal één poort down, er is er geen sprake van een uni-directionele verbinding welke loops zal veroorzaken. Port-negotiation zorgt dus naast het onderhandelen over link-snelheden en duplex-mode ook voor een snelle detectie van link-failure.

Het uitschakelen van de port-negotiation heeft consequenties voor de convergentietijd van RSTP. Als de root-poort van een switch X 'not connected' wordt en de andere kant (de designated-port op switch Y) up blijft, zal switch X door de poort-verandering meteen van een alternate- of backup-port een root-port maken. Anders is het wanneer de root-poort van switch X up blijft, maar de poort aan de andere kant (de designated switch-port op switch Y) 'not connected' wordt. Switch X zal pas na het missen van 3 hello's van switch Y overgaan naar een andere root-poort. Wanneer de BPDU's elke 2 seconden verstuurd worden, is dit een convergentietijd van 6 seconden.

Wanneer port-negotiation aan één zijde van de verbinding wordt aangezet, zal deze poort 'not connected' worden omdat negotiation aan de andere zijde niet aan staat.

UDLD versus Loop Guard

UDLD is flexibeler dan Loop Guard als er gebruik wordt gemaakt van EtherChannels. Als één van de verbindingen van de EtherChannel faalt en uni-directioneel wordt, zal alleen deze verbinding in de disabled mode komen. Het PAgP zorgt ervoor dat de EtherChannel actief blijft met de resterende verbindingen. Met Loop Guard, dat op STP-niveau kijkt en een EtherChannel als een logische poort ziet, zal de gehele EtherChannel in loop-inconsistent state brengen

Wanneer in de ring uitsluitend darkfiber wordt gebruikt, zijn UDLD en Loop Guard overbodig omdat uni-directionele verbindingen niet voor kunnen komen.

4.6.3 Koppeling klant/SURFnet

Edge-port versus link-port

Wanneer de klant op de switch aansluit met een router, zullen er door de switchpoort geen BPDU's ontvangen worden omdat de router op laag 2 als een eindstation gezien kan worden en geen STP gebruikt. De betreffende poort op de GigaMAN switch is als een edge-port geconfigureerd. Anders wordt het wanneer de klant met een switch-port aansluit welke BPDU's verstuurt omdat STP (default) geconfigureerd is. De poort op het GigaMAN zal dan door het ontvangen van BDDU's automatisch overgaan van edge-port naar link-port. De klant-switch wordt zo onderdeel van het GigaMAN (R)STP domein. Op het moment dat de klant-switch geen RSTP maar STP praat, zal RSTP op het segment van de klant-poort gebruik gaan maken van de STP-timers. Het gevolg is dat de klant-poort met het actief worden pas na 30 seconden in de forwarding mode komt.

Invloed klant op convergentietijd

Als er met een enkele spanning tree voor alle VLAN's gewerkt zal worden, kan de klant de gezamenlijke spanning tree van alle VLAN's voor alle klanten beïnvloeden. Zo kan een klant van de eigen switch een root-switch maken. Door geen MST maar Rapid PVST+ te gebruiken waarbij ieder VLAN een eigen spanning tree heeft, kan de klant alleen root-switch worden voor het eigen VLAN. Spanning trees van andere VLAN's worden hierbij niet beïnvloed en blijft loadbalancing gehandhaafd. Met het commando op de klant-poort:

```
set spantree guard root x/x
```

wordt voorkomen dat een klant root-switch wordt. Vooral bij het gebruik van MST is dit van belang.

BPDU Guard

Er kan voor gekozen worden om klant-switches geen onderdeel van het RSTP-domein te laten worden. Bijvoorbeeld om te voorkomen dat er door de klant topology-changes verstuurd worden of dat er onnodige root-elections plaatsvinden waardoor de switches het onvoorzien (te) druk kunnen krijgen wat tot instabiele situaties kan leiden. Het afbakenen van een (R)STP-domein kan met BPDU Guard gerealiseerd worden. Hiermee wordt de edge-port van de GigaMAN switch automatisch in errdisable mode gebracht als de klant BPDU's verstuurt. Om de handmatige tussenkomst van de beheerder te voorkomen wanneer een poort in errdisable mode is gekomen door een misconfiguratie van de klant, kan er een tijdsinterval geconfigureerd worden op de GigaMAN-switch, waarna de poort automatisch weer 'up' wordt gebracht.

Voorbeeld van een configuratie met BPDU Guard op een CatOS 4000 met een tijdsinterval van 5 minuten:

```
set errdisable-timeout enable bpdu-guard
set errdisable-timeout interval 300
set spantree portfast bpdu-guard 2/12 enable
```

Ook voor UDLD is het mogelijk om een errdisabled uni-directionele poort na dezelfde tijdsinterval weer automatisch actief te maken:

```
set errdisable-timeout enable udld
```

Als er door de klant een laag 3 switch (zoals een 3550) gebruikt wordt voor de aansluiting op de ring, kan er op twee manieren een IP-interface gedefinieerd worden:

- Switch Virtual Interfaces
- Routed Ports

De Switch Virtual Interface (SVI) representeert een VLAN met switch-poorten als een IP-interface naar de routeringsfunctie van het netwerk. De IP-interface is virtueel aan een VLAN gekoppeld. Deze oplossing wordt gebruikt om meerdere host in een VLAN van een gateway te voorzien.

Een Routed Port is een fysieke poort op de switch en gedraagt zich hetzelfde als een poort op een router. Deze oplossing wordt gebruikt om één host of een andere router aan de switch te koppelen.

Voorbeeld configuratie op een 3550 laag 3 switch waarbij poort 11 gekoppeld wordt aan de GigaMAN switch:

Switch Virtual Interface:

```
!
interface Vlan5
 ip address 145.125.16.70 255.255.255.240
!
interface GigabitEthernet0/11
 description ts2.amsterdam1 ( GE 2/12 )
 switchport access vlan 5
 switchport mode access
 no ip address
!
```

Routed Port:

```
!
interface GigabitEthernet0/11
 description ts2.amsterdam1 ( GE 2/12 )
 no switchport
 ip address 145.125.16.70 255.255.255.240
!
```

Met een Switch Virtual Interface wordt poort 11 in VLAN 5 geplaatst. Deze poort verzendt standaard BPDU's. Door spanningtree uit te schakelen (no spanning-tree vlan 5) kan dit gestopt worden. Dit kan echter loops in het netwerk van de klant veroorzaken. Een betere oplossing om het verzenden van BPDU's op één poort uit te schakelen is het commando:

```
interface GigabitEthernet0/11
 spanning-tree bpdufilter enable
```

Er wordt slechts één poort in het VLAN 5 geplaatst. Dit de de uplink naar SURFnet. Beter is het om met een Routed Port aan te sluiten waarbij geen VLAN nodig is en er geen BPDU's vanaf deze poort verzonden zullen worden.

4.6.4 Voorbeeld-configuratie

Voor de voorbeeld-configuratie van RSTP wordt verwezen naar de configuraties met bijbehorende architectuur van de RSTP tests welke uitgevoerd zijn op het testnetwerk van SURFnet.

4.6.5 Migratie tips

In de huidige situatie wordt er vanuit gegaan dat STP zonder extra instellingen is geconfigureerd.

- Communiceer naar de klanten toe dat er geen BPDU's toegestaan zullen gaan worden op de uplink-poort.

- Configureer de edge-ports. Dit configureren wordt gerealiseerd met portfast en dient voor zowel de klant-poorten als de trunk naar de router te gebeuren.

  Configureer de klant-poorten als edge-port:

  ```
  set spantree portfast x/x enable
  ```

  Configureer de trunk naar de router als edge-port:

  ```
  set spantree portfast x/x enable trunk
  ```

- Activeer RSTP.

  Op de (toe te wijzen) root-switch:

  ```
  set spantree priority 4096 1-1005
  ```

  Activeer RSTP op alle switches in de ring:

  ```
  set spantree mode rapid-pvst+
  ```

  Als alle consoles van de switches in de ring direct of via out-of-band te benaderen zijn, is het zinvol om dit laatste commando tegelijkertijd op alle switches uit te voeren. In de test-omgeving met drie switches was het netwerk 2,5 seconden instabiel tijdens het overgaan van pvst+ naar rapid-pvst+.

  Als niet alle consoles van de switches te benaderen zijn zal het omzetten in-band met telnet of SSH uitgevoerd moeten worden. Hierbij wordt begonnen met de (root-)switch op de POP-locatie en daarna iedere volgende switch links- en rechtsom afwisselend naar de switch met de blocked poort toe. Het nadeel is dat omzetting van iedere switch afzonderlijk een onbereikbaarheid en instabiliteit van 30 seconden van het netwerk tot gevolg heeft totdat de laatste switch van STP naar RSTP wordt overgezet.

- BPDU guard wordt voor de klanten aangezet. Om te voorkomen dat BPDU guard daadwerkelijk op de klant-poort wordt aangezet en deze meteen in errdisable mode komt, moet eerst gekeken worden of de klant geen BPDU's verstuurt zodat de interface een edge-port is en geen link port is. Dit kan worden bekeken met het commando 'sh spantree x/x' :

  ```
  ts2.amsterdam1> (enable) sh spantree x/x
  Edge Port:          No, (Configured) Enable
  Link Type:         P2P, (Configured) Auto
  Port Guard:    Default
  Port                      Vlan State         Role Cost     Prio Type
  ------------------------ ---- ------------ ---- -------- ---- -----------------
   2/12                     1    forwarding    DESG        4   32 P2P, PEER(STP)
  ```

  Te zien is dat de klant-poort geen edge port maar een link port is. In het volgende geval stuurt de klant geen BPDU's en is de poort een edge port:

  ```
  ts2.amsterdam1> (enable) sh spantree x/x
  Edge Port:         Yes, (Configured) Enable
  ```

```
Link Type:          P2P, (Configured) Auto
Port Guard:     Default
Port                        Vlan State          Role Cost     Prio Type
------------------------ ---- ------------ ---- -------- ---- -----------------
 2/12                       1    forwarding    DESG        4   32 P2P, Edge
```

Globale commando's op iedere switch voor het configureren van de interval:

```
set errdisable-timeout enable bpdu-guard
set errdisable-timeout interval 300
```

Het activeren van BPDU guard op de betreffende klant-poort:

```
set spantree portfast bpdu-guard x/x enable
```

4.6.6 Conclusie

Met het onderzoek naar de mogelijkheden om de convergentietijd van GigaMAN's te reduceren, is gekeken naar het aanpassen van de huidige STP-parameters, het gebruik van *fast-enhancements en naar RSTP. Het aanpassen van de huidige STP-parameters resulteert in een aanzienlijke verbetering van de convergentietijd (tot 15 seconden) welke met weinig moeite is te implementeren. Met RSTP als nieuwer spanning tree protocol, is de convergentietijd echter te reduceren tot maximaal 2 seconden. RSTP bevat alle *fast-enhancements zoals Portfast, Uplinkfast, en Backbonefast en is in tegenstelling tot deze *fast-enhancements eenvoudig te configureren. Na afweging is er gekozen voor RSTP.

Op de gebruikte GigaMAN-switches is het op twee manieren mogelijk om RSTP te implementeren: MST en rapid pvst+. Vanwege de eenvoud in configureren en het uitsluiten van interferentie tussen de klanten onderling op dezelfde ring, is er voor rapid pvst+ gekozen.

Opmerkelijk is dat testen uitwijzen dat IOS-gebaseerde switches veel sneller convergeren dan Cat-OS gebaseerde switches. Deze laatste worden gebruikt in de GigaMAN's. Het verschil is tientallen milli-seconden tegenover honderdtallen milli-seconden.

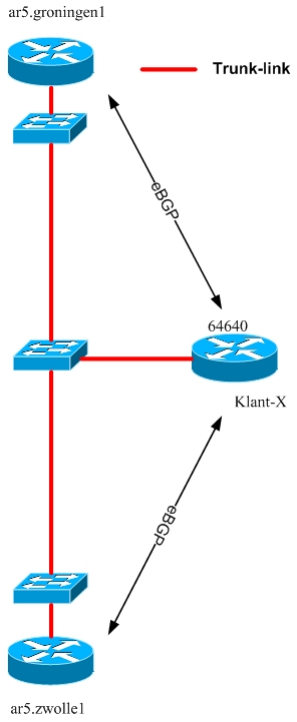
Wanneer er gebruik wordt gemaakt van directe glasvezelverbindingen, zijn de extra maatregelen om enkelvoudige loops tegen te gaan overbodig. Het gebruik van port-negotiation wordt aangeraden om te voorkomen dat convergentie in sommige gevallen 6 seconden duurt.

Om ongewenste invloeden van klanten op de spanning tree van de GigaMAN ring te voorkomen, wordt aangeraden om klant-switches niet in het spanning tree domein te betrekken en hier maatregelen tegen te treffen.

Hoofdstuk 5

Multi-POP

De instellingen welke niet direct op een POP kunnen worden aangesloten, omdat er geen POP in de buurt is, zijn aangesloten op de POPs elders. De verbinding naar deze POPs zijn point-to-point verbindingen. De klant router 'praat' BGP met de POP routers, zodoende wordt de redundantie gerealiseerd. Mocht de link naar het primaire pad onbeschikbaar worden, dan kan de klant router via het de secundaire POP het SURFnet5 netwerk bereiken. Dit wordt geregeld via het BGP protocol. Het netwerk ziet voor deze multi-POP aansluiting alsvolgt uit, zie figuur5.1

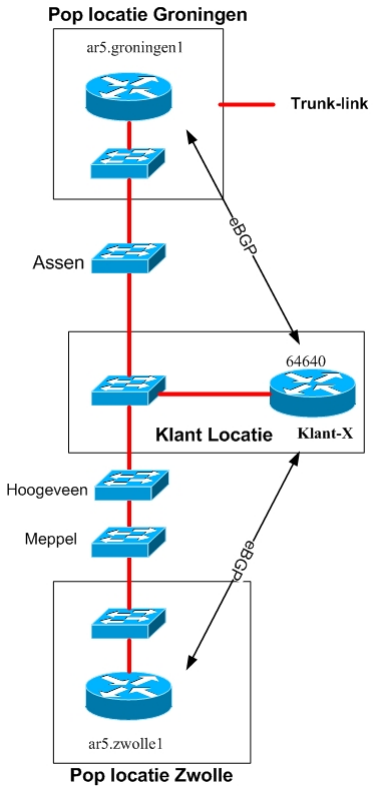


Figuur 5.1: *Multi-POP aansluiting*

BGP stuurt standaard (althans in de Cisco implementatie) om de 60 seconden een keep-alive pakket, dit wordt de hello-time genoemd. De standaard hold-timer, dit is de tijd waarna BGP zijn peer als onbereikbaar beschouwd, is 3 maal de hello-time. Op dit moment worden deze multi-POP

aansluitingen gerealiseerd met default timers. Dit heeft als resultaat dat waneer er ergens een link uitvalt (afhankelijk van waar in het point to point netwerk), dit tot 3 minuten (3 maal 60 seconden) kan duren voordat de peer als onbereikbaar wordt beschouwd. Deze tijd is erg lang en de wens van SURFnet is om deze tijd te reduceren.

Op Cisco routers staat standaard de feature 'BGP fast-external-failover' aan. Dit betekent dat wanneer een interface van de router waaraan de peer/neighbor is verbonden, down gaat door bijvoorbeeld een defecte interface of glasfiber, de hold-timer niet wordt gebruikt en de peer direct als onbereikbaar wordt beschouwd. Dit geldt echter alleen voor de interfaces van de router, dus van de router naar de eerste switch. Deze verbindingen zijn typisch verbindingen binnen een POP ruimte. De kans dat hier een fiber kapot gaat is niet groot. De kans dat de fibers tussen de switches onderling kapot gaan is groter. De fibers tussen de switches liggen door heel het land, het gebeurd dan ook met enige regelmaat (veel meer dan de fibers binnen de pop locatie) dat er een fiber-cut is. Deze fiber-cuts worden niet gedetecteerd door het fast-external-failover mechanisme omdat de interface van de routers niet down zullen gaan. In figuur 5.2 is dit nog eens weergegeven. Zoals is te zien kan het zijn dat de link naar de POP via meerdere laag 2 hops verloopt.



Figuur 5.2: *Multi-POP aansluitingen*

5.1 Mogelijke oplossingen

De 180 seconden die het nu duurt is erg lang vergeleken met de convergentietijden van andere delen van SURFnet5. Er zijn verschillende oplossingen denkbaar om de convergentietijd te reduceren. De meest voor de hand liggende mogelijkheid is de BGP timers aan te passen. Deze aanpassing is

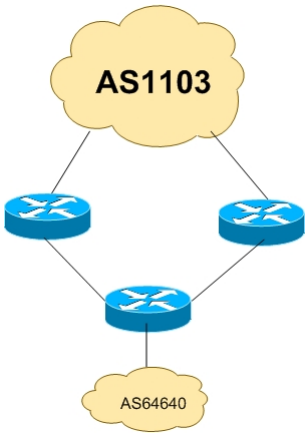
relatief eenvoudig te implementeren doordat er weinig hoeft te worden aangepast. Een alternatief is het gebruik van een IGP in plaats van BGP. Verschillende IGP's zoals IS-IS, OSPF en EIGRP hebben de eigenschap sneller te convergeren dan BGP. Het nadeel hiervan is echter dat het nogal wat nadelige gevolgen kan hebben op de stabiliteit van het netwerk als de klant-router niet juist geconfigureerd is. Het verdient daarom niet de voorkeur om het klant-netwerk in het SURFnet5 domein te betrekken.

Een andere mogelijkheid kan zijn het gebruik van Virtual Router Redundancy Protocol (VRRP) of Hot Standby Router Protocol (HSRP). In zo'n situatie zijn er 2 routers (de POP routers) die als default-gateway fungeren. De klant gebruikt als default-gateway het virtuele HSRP/VRRP adres.

5.2 Theorie BGP en HSRP / VRRP

5.2.1 Theorie BGP

Het klant-netwerk is in de huidige situatie via twee verschillende uplinks verbonden met het SURFnet netwerk. Het Autonomous System (AS) heeft 2 uplinks naar de zelfde ISP, dit is weergegeven in figuur5.3



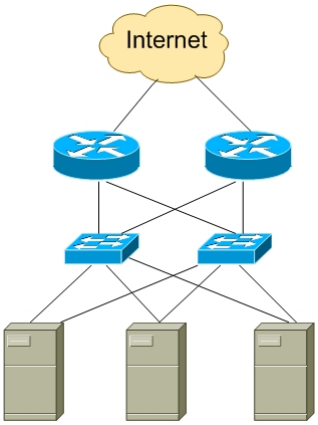
Figuur 5.3: *AS64640 heeft twee uplinks naar AS1103*

De convergentietijd in het geval van een failure is eenvoudig te berekenen met behulp van de hold-time en de keep-alive interval. In de huidige situatie kan het tot drie minuten duren voordat het alternatieve pad wordt gekozen. De BGP routers houden een hold-timer bij welke iedere keer gereset wordt (op nul wordt gezet) wanneer er een keep-alive van zijn peer wordt ontvangen. Als de hold-timer zijn maximale waarde bereikt (180sec is bij Cisco de default waarde) zal de peer als onbereikbaar worden beschouwd. Alle routes die via de betreffende peer zijn geleerd, zullen worden terug getrokken en uit de FIB[1] worden verwijderd.

[1]Forwarding Information Base (FIB), dit is de routing table

5.2.2 Theorie HSRP / VRRP

Technieken zoals het Virtual Router Redundancy Protocol (VRRP) of Hot Standby Router Protocol (HSRP), worden veel gebruikt om de netwerk uptime zo hoog mogelijk te krijgen. Situaties waarin deze technieken worden toegepast zijn bijvoorbeeld hosting netwerken. Dit zijn netwerken waarin typisch veel servers staan. Deze hebben allemaal een default gateway geconfigureerd, dit adres is dan een virtueel adres dat wordt gedeeld door twee of meer routers. In figuur 5.4 is een dergelijk netwerk weergegeven



Figuur 5.4: *VRRP/HSRP netwerk*

In figuur 5.4 zijn drie servers verbonden met twee switches. Deze twee switches zijn weer verbonden met de routers, welke ieder een uplink naar het Internet hebben. De servers gebruiken het ip adres 192.0.2.1 als default-gateway. Dit is echter een virtueel adres. De routers zelf hebben het adres 192.0.2.2 en 192.0.2.3, deze zitten samen in een HSRP/VRRP groep. In de normale situaties reageert de router met het adres 192.0.2.2 op ARP queries voor 192.0.2.1. De router met het adres 192.0.2.2 is dus de default-gateway. De twee routers onderling sturen keep-alive (multicast) packets naar elkaar. Zodoende kan worden gecontroleerd of de primaire router nog bereikbaar is. Mocht dit niet het geval zijn, dan neemt de backup router het virtuele IP en MAC adres over. Wie de master is kan met behulp van prioriteiten worden geconfigureerd. De default timers voor HSRP zijn drie seconden voor de hello-time en 10 seconden voor de de hold-time. Deze kunnen veranderd worden naar beide minimaal 1 seconde. De hello-time bij VRRP is standaard 1 seconde, deze kan geconfigureerd worden tot in milli-seconden. Met VRRP zal dus een snellere convergentietijd kunnen worden gerealiseerd dan HSRP. Echter, de hello-time dient niet te kort worden ingesteld. Dit kan vervelende gevolgen zoals het steeds wisselen van de master tot gevolg hebben. Het Gateway Load Balancing Protocol is een alternatief voor deze twee protocollen. Hiermee kan het verkeer worden ge-loadbalanced, het verkeer wordt dan dus verdeeld over 2 aktiefe routers. Het Gateway Load Balancing Protocol is net als HSRP een Cisco proprietary protocol, VRRP is een IEEE standaard.

5.3 Gekozen technologie

De eenvoudigste manier om de convergentietijd van de multi-POP aansluiting te verbeteren is door te blijven bij de huidige BGP aanpak. De tijd die het nu duurt voor dat het netwerk is geconvergeerd

na een fiber-cut is 2 tot 3 minuten. Deze tijd is door het aanpassen van de hold-time en de hello-time eenvoudig te reduceren. In theorie zouden dezelfde tijden als bij HSRP kunnen worden bereikt.

Een oplossing waarbij HSPR of VRRP wordt gebruikt is complexer en heeft bovendien tot gevolg dat er nogal wat gewijzigd moet worden aan de huidige situatie. Aan de klant zijde zullen de 2 VLANs naar de POPs, samen gevoegd moeten worden tot 1 VLAN. Dit is nodig omdat de twee routers elkaar moeten kunnen bereiken voor de hello-pakketten. De klant router wordt geconfigureerd met als default-gateway het virtuele ip adres van de HSRP/VRRP groep. Door één van de twee routers een hogere prioriteit te geven kan voor het uitgaande verkeer (van de klant naar SURFnet/Internet) een primair pad worden gekozen. In het geval van een link failure naar de aktieve router, zal de backup router het virtuele mac en IPadres overnemen. Voor het retour verkeer is het wat complexer. Op de twee POP routers zal een statische route moeten worden opgenomen voor de prefixen van de klant naar het ip adres van de klant router. De POP routers zullen deze statische route dan moeten injecteren in IS-IS. Hier doet zich dan een probleem voor. Om wederom een primair en secondair pad te creëren, zal de POP router van het primaire pad de prefix adverteren met een betere metric dan de andere POP router. Echter, wat als er een fiber tussen de POP router (van het primaire pad) en de klant router uitvalt? Er is geen mechanisme die dit kan detecteren en tegen IS-IS kan vertellen dat deze statische route niet meer geldig is. Dit komt omdat de klant-router in deze opstelling geen IS-IS praat. Het gevolg zal zijn dat al het retour verkeer naar de router van de primaire POP locatie gestuurd zal blijven worden. Een oplossing is om de klant router te betrekken bij het IS-IS domein. Echter hieraan kleven nogal wat nadelen en het past niet in de policy van SURFnet om dit te doen. Vanuit security oogpunt is dit ook erg gevaarlijk, tenzij er goed afgeschermd kan worden welke prefixen er van een bepaalde router via IS-IS ontvangen mogen worden.

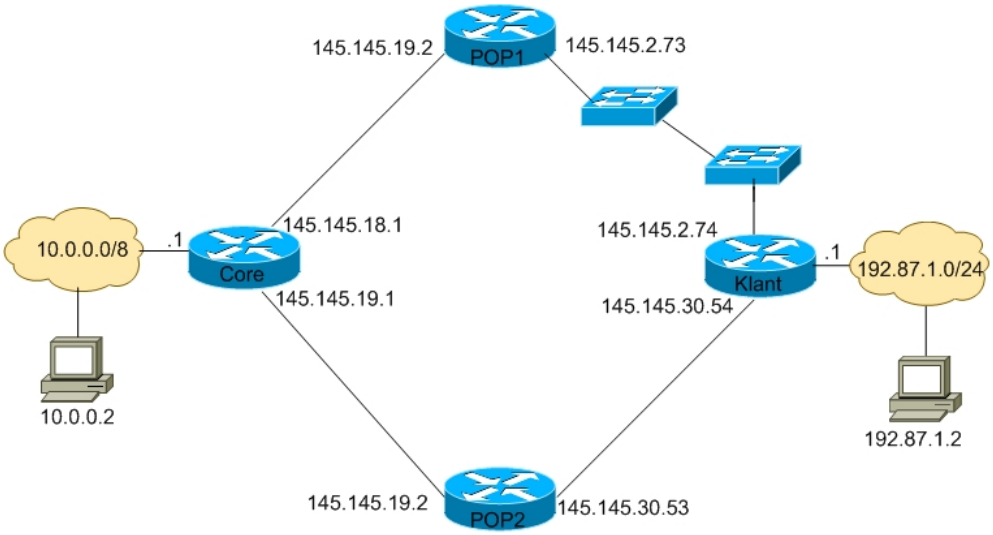
Er is uiteindelijk gekozen om verder te werken met de huidige techniek. Door het aanpassen van de BGP timers, kan met een minimale impact op het huidige platform toch een verbetering van de convergentietijd worden gerealiseerd.

5.4 BGP testen in het HvU communicatie lab

In het lab op de HvU zijn een aantal metingen gedaan om de convergentietijden die met BGP te realiseren zijn te testen. In het lab is het netwerk als in figuur5.5 gebouwd.

De bijbehorende relevante configuratie is hieronder weergegeven. Hieronder de configuratie van de klant router, de verbinding naar POP2 is een serieële verbinding, de overige connecties zijn FastEthernet.

```
!
hostname klant
!
interface FastEthernet0/0
ip address 192.87.1.1 255.255.255.0
duplex auto
speed auto
!
interface FastEthernet0/1
description connection to POP1
ip address 145.145.2.74 255.255.255.252
```



Figuur 5.5: *BGP test-opstelling*

```
duplex auto
speed auto
!
interface Serial0/1
description connection to POP2
ip address 145.145.30.54 255.255.255.252
!
router bgp 64640
bgp log-neighbor-changes
network 192.87.1.0
neighbor 145.145.2.73 remote-as 1103
neighbor 145.145.2.73 prefix-list surfnet-in in
neighbor 145.145.2.73 route-map surfnet-prim-uit out
neighbor 145.145.30.53 remote-as 1103
neighbor 145.145.30.53 prefix-list surfnet-in in
neighbor 145.145.30.53 route-map surfnet-sec-uit out
!
ip prefix-list surfnet-in seq 5 permit 0.0.0.0/0
access-list 99 permit 192.87.1.0 0.0.0.255
!
route-map surfnet-prim-uit permit 10
match ip address 99
set metric 10
!
route-map surfnet-sec-uit permit 10
match ip address 99
set metric 20
!
end
```

De configuratie van de POP1 router staat hieronder, de verbinding naar de klant router is een ethernet verbinding, de verbinding naar de core is serieël.

```
hostname pop1
!
interface FastEthernet0/0
description connection to KLANT
ip address 145.145.2.73 255.255.255.252
duplex auto
speed auto
!
router eigrp 1103
network 145.145.2.72 0.0.0.3
network 145.145.18.0 0.0.0.3
no auto-summary
!
router bgp 1103
no synchronization
bgp log-neighbor-changes
neighbor 145.145.2.74 remote-as 64640
neighbor 145.145.2.74 default-originate route-map Klant-X-out
neighbor 145.145.2.74 prefix-list Klant-X in
neighbor 145.145.2.74 distribute-list 99 out
neighbor 145.145.2.74 route-map Klant-X-in in
neighbor 145.145.18.1 remote-as 1103
!
ip prefix-list Klant-X seq 5 permit 192.87.1.0/24
ip prefix-list Klant-X seq 10 deny 0.0.0.0/0 le 32
!
access-list 99 permit 0.0.0.0
access-list 99 deny any
!
route-map Klant-X-in permit 10
set local-preference 200
!
route-map Klant-X-out permit 10
match ip address 99
set metric 10
!
end
```

De configuratie voor de POP2 routers is bijna hetzelfde als POP1, de verbinding naar de core en klant routers zijn nu beide serieële verbindingen.

```
hostname pop2
!
interface Serial0/0
ip address 145.145.19.2 255.255.255.252
```

```
no fair-queue
clockrate 128000
!
interface Serial0/1
ip address 145.145.30.53 255.255.255.252
clockrate 128000
!
router eigrp 1103
network 145.145.19.0 0.0.0.3
network 145.145.30.52 0.0.0.3
no auto-summary
!
router bgp 1103
no synchronization
bgp log-neighbor-changes
neighbor 145.145.19.1 remote-as 1103
neighbor 145.145.30.54 remote-as 64640
neighbor 145.145.30.54 default-originate route-map Klant-X-out
neighbor 145.145.30.54 prefix-list Klant-X in
neighbor 145.145.30.54 distribute-list 99 out
neighbor 145.145.30.54 route-map Klant-X-in in
!
ip prefix-list Klant-X seq 5 permit 192.87.1.0/24
ip prefix-list Klant-X seq 10 deny 0.0.0.0/0 le 32
!
access-list 99 permit 0.0.0.0
access-list 99 deny any
!
route-map Klant-X-in permit 10
set local-preference 195
!
route-map Klant-X-out permit 10
match ip address 99
set metric 20
!
```

De configuratie van de core router is alsvolgt, de verbinding naar POP2 en POP1 zijn serieële verbindingen. De core router heeft een default-route staan naar het werkstation 10.0.0.2.

```
!
hostname core
!
interface FastEthernet0/0
ip address 10.0.0.1 255.0.0.0
duplex auto
speed auto
!
interface Serial0/0
```

```
ip address 145.145.19.1 255.255.255.252
!
interface Serial0/1
ip address 145.145.18.1 255.255.255.252
!
router eigrp 1103
network 10.0.0.0
network 145.145.18.0 0.0.0.3
network 145.145.19.0 0.0.0.3
no auto-summary
!
router bgp 1103
no synchronization
bgp log-neighbor-changes
network 10.0.0.0
neighbor 145.145.18.2 remote-as 1103
neighbor 145.145.18.2 default-originate route-map default-map
neighbor 145.145.19.2 remote-as 1103
neighbor 145.145.19.2 default-originate route-map default-map
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.0.0.2
!
access-list 1 permit 0.0.0.0
route-map default-map permit 10
match ip address 1
!
end
```

Als IGP wordt in deze test-opstelling EIGRP gebruikt. Dit is nodig om er voor te zorgen dat de core router weet hoe deze de klant router moet bereiken. In een stabiele situatie waarbij alle paden actief zijn, heeft de core router twee paden naar het klanten netwerk 192.87.1.0/24. Dit is te zien in de BGP tabel van de core router:

```
core#sh ip bgp
BGP table version is 37, local router ID is 145.145.19.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

 Network       Next Hop Metric LocPrf Weight Path
*> 10.0.0.0      0.0.0.0 0 32768 i
*>i192.87.1.0   145.145.2.74 10 200 0 64640 i
* i             145.145.30.54 20 195 0 64640 i
```

De core router kan kiezen tussen twee verschillende next-hop adressen voor het netwerk 192.87.1.0/24. Het verschil is dat de route met next-hop 145.145.2.74 een localpref heeft van 200 en de route met next-hop 145.145.30.54 een localpref van 195. De route met de hoogste localpref wordt gebruikt.

Dit is te zien aan het > teken. Als het goed is, wordt de route met next-hop 145.145.2.74 dus in de route tabel gezet. De route tabel van de core router ziet er alsvolgt uit:

```
#sh ip route
<knip>
Gateway of last resort is 10.0.0.2 to network 0.0.0.0

C 145.145.18.0 is directly connected, Serial0/1
C 145.145.19.0 is directly connected, Serial0/0
D 145.145.30.52 [90/21024000] via 145.145.19.2, 01:10:09, Serial0/0
D 145.145.2.72 [90/2172416] via 145.145.18.2, 01:01:59, Serial0/1
C 10.0.0.0/8 is directly connected, FastEthernet0/0
B 192.87.1.0/24 [200/10] via 145.145.2.74, 00:01:51
S* 0.0.0.0/0 [1/0] via 10.0.0.2
```

Voor de volledigheid is hieronder de BGP en route table van de klant weergegeven.

```
klant#sh ip bgp
BGP table version is 7, local router ID is 192.87.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

 Network    Next Hop Metric LocPrf Weight Path
*> 0.0.0.0   145.145.2.73 10 0 1103 i
*            145.145.30.53 20 0 1103 i
*> 192.87.1.0   0.0.0.0 0 32768 i
```

De klant router heeft van beide peers een default route ontvangen, de route met de laagste metric wordt geïnstalleerd in de route tabel.

```
klant#sh ip route
<knip>
Gateway of last resort is 145.145.2.73 to network 0.0.0.0

C 145.145.30.52 is directly connected, Serial0/1
C 145.145.2.72 is directly connected, FastEthernet0/1
C 192.87.1.0/24 is directly connected, FastEthernet0/0
B* 0.0.0.0/0 [20/10] via 145.145.2.73, 00:01:04
```

Voor al het verkeer van en naar de het klanten netwerk 192.87.1.0/24, wordt dus het bovenste pad gebruikt. In de test is met ping gemeten hoelang het duurde voordat de verbinding terug was na een gesimuleerde glasvezel breuk. De ping is iedere keer gestart vanaf 10.0.0.2 en is gedaan naar 192.87.1.2. Omdat het bovenste pad in een stabiele situatie als primair pad wordt gebruikt is ook hier iedere keer de link onderbroken. De link werd onderbroken door de kabel tussen de twee switches te verwijderen. Dit is zo gedaan omdat anders het "BGP fast-external-failover"mechanisme zou gaan werken. Omdat dit in de praktijk bij het grootste gedeelte van de glasvezel breuken ook niet zo zal zijn, is er tussen de switches een breuk gesimuleerd.

5.4.1 Test resultaten

In de eerste test is er niets veranderd aan de timers. Er wordt dus gebruik gemaakt van de default instellingen, 60 seconden voor de keep-alive pakketten en 180 seconden voor de hold-timer. Het resultaat was dat het tussen de twee en drie minuten duurde voordat de de beide routers (core en klant) het onderste pad kozen. Maximaal kan het met deze configuratie 3 minuten duren voor dat het alternatieve pad gekozen wordt. Dit is afhankelijk van hoelang na een keep-alive pakket de kabel wordt verwijdert uit het bovenste pad. Stel dat dit 30 seconden na een keep-alive gebeurd, dan zal het alternatieve pad na twee en een halve minuut (150 seconden) gekozen worden. Want 30sec + 150 sec = 180 seconden =hold-time.

In een tweede meting werden zowel de hold-timer als de keep-alive timer aangepast naar een andere waarde. Dit maal werden de waarde zoals deze in RFC1771 worden aanbevolen gebruikt. In RFC1771 staat hierover het volgende:

```
The suggested value for the Hold Time is 90 seconds.
The suggested value for the KeepAlive timer is 30 seconds.
```

Deze timers zijn alsvolgt geconfigureerd op de klant router:

```
router bgp 64640
timers bgp 30 90
<etc>
```

De convergentietijd ligt nu tussen de 60 en 90 seconden. Dit is wederom afhankelijk van hoeveel seconden na een eerder keep-alive pakket de link wordt onderbroken. Er zijn nog een aantal testen gedaan met een andere timers. Echter omdat de convergentietijd eenvoudig te bereken is zullen deze niet verder worden besproken. Convergentietijd: maximaal $hold-time en minimaal: ($hold-time - keep-alive-time).

5.5 BGP testen op het SURFnet test netwerk

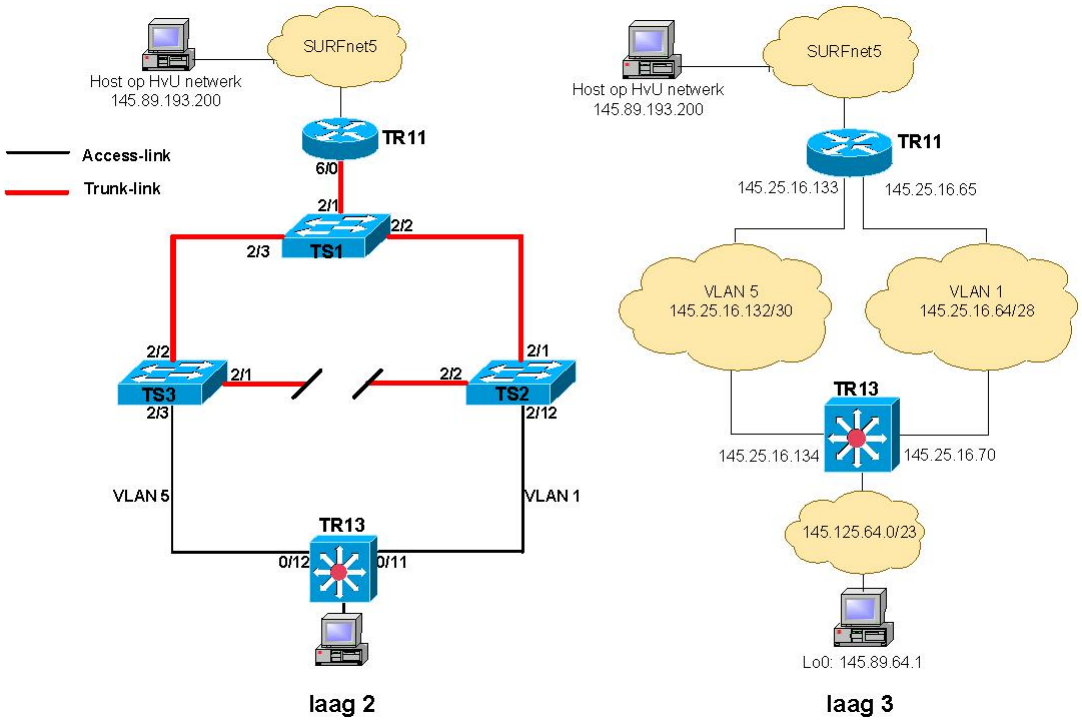
Om te controleren of de resultaten die behaald zijn op de apparatuur in het communicatie lab op de HvU hetzelfde zullen zijn als die op de apparatuur die door SURFnet wordt gebruikt, zijn soortgelijke testen gedaan op het testnetwerk van SURFnet.

5.5.1 Architectuur test opstelling

De testopstelling zoals deze bij de RSTP testen is gebruikt is zo min mogelijk aangepast. Hoe van de RSTP testopstelling naar de BGP testopstelling kan worden gemigreerd is beschreven in bijlage I.

De testopstelling voor deze test is weergegeven in figuur 5.6.

Zoals is te zien, zijn er vanaf TR11 twee mogelijke paden naar TR13. TR13 is de klant-router en adverteert het netwerk 145.125.64.0/23. VLAN1 is het primaire pad, VLAN5 is het secondaire pad.



Figuur 5.6: *bgptestopstelling*

5.5.2 Configuratie test opstelling

Er zijn twee een BGP sessies opgezet tussen TR13 en TR11. De relevante configuratie voor TR13 ziet er alvolgt uit:

```
hostname tr13.amsterdam1
!
interface Loopback0
description Lo0.tr13.amsterdam1
ip address 145.125.64.1 255.255.254.0
no ip mroute-cache
!
interface GigabitEthernet0/11
description ts2.amsterdam1 ( GE 2/12 )
no switchport
ip address 145.125.16.70 255.255.255.240
!
interface GigabitEthernet0/12
description ts2.amsterdam1 ( GE 2/11 )
no switchport
ip address 145.125.16.134 255.255.255.252
speed nonegotiate
!
router bgp 64640
```

```
bgp log-neighbor-changes
network 145.125.64.0 mask 255.255.254.0
neighbor 145.125.16.65 remote-as 1125
neighbor 145.125.16.65 description primair
neighbor 145.125.16.65 prefix-list surfnet-in in
neighbor 145.125.16.65 route-map surfnet-prim-uit out
neighbor 145.125.16.133 remote-as 1125
neighbor 145.125.16.133 description secondair
neighbor 145.125.16.133 prefix-list surfnet-in in
neighbor 145.125.16.133 route-map surfnet-sec-uit out
!
ip prefix-list surfnet-in seq 5 permit 0.0.0.0/0
!
route-map surfnet-prim-uit permit 10
match ip address 99
set metric 10
!
route-map surfnet-sec-uit permit 10
match ip address 99
set metric 20
!
```

TR13 adverteert naar zijn beide peers de prefix 145.125.64.0/23. Daarnaast ontvangt deze van beide peers een default route (0.0.0.0/0). De relevante configuratie voor TR11 ziet er alsvolgt uit:

```
hostname Tr11.Amsterdam1
!
interface GigabitEthernet6/0
description primair
ip address 145.125.16.65 255.255.255.240
no ip directed-broadcast
ip route-cache flow sampled input
load-interval 30
no negotiation auto
ipv6 enable
!
interface GigabitEthernet6/0.5
description secundair
encapsulation dot1Q 5
ip address 145.125.16.133 255.255.255.252
ip access-group 2002 in
no ip directed-broadcast
no cdp enable
!
router bgp 1125
neighbor 145.125.16.70 remote-as 64640
neighbor 145.125.16.70 activate
neighbor 145.125.16.70 default-originate route-map tr13-prim-out
```

```
neighbor 145.125.16.70 prefix-list tr13 in
neighbor 145.125.16.70 distribute-list 99 out
neighbor 145.125.16.70 route-map tr13-prim-in in
neighbor 145.125.16.134 remote-as 64640
neighbor 145.125.16.134 activate
neighbor 145.125.16.134 default-originate route-map tr13-sec-out
neighbor 145.125.16.134 prefix-list tr13 in
neighbor 145.125.16.134 distribute-list 99 out
neighbor 145.125.16.134 route-map tr13-sec-in in
!
ip prefix-list tr13 seq 5 permit 145.125.64.0/23
ip prefix-list tr13 seq 10 deny 0.0.0.0/0 le 32
!
access-list 99 permit 0.0.0.0
access-list 99 deny any
!
route-map tr13-prim-in permit 10
set local-preference 200
!
route-map tr13-prim-out permit 10
match ip address 99
set metric 10
!
route-map tr13-sec-in permit 10
set local-preference 195
!
route-map tr13-sec-out permit 10
match ip address 99
set metric 20
```

In een normale situatie zullen er 2 paden beschikbaar zijn, één hiervan wordt in de route tabel geïnstaleerd. Dit zal het pad via VLAN1 zijn.

```
Tr11.Amsterdam1#sh ip bgp 145.125.64.0
BGP routing table entry for 145.125.64.0/23, version 4057
Paths: (2 available, best #2)
Multipath: iBGP
 Advertised to update-groups:
 3 4 5 6
 64640
 145.125.16.134 from 145.125.16.134 (145.125.64.1)
 Origin IGP, metric 20, localpref 195, valid, external
 64640
 145.125.16.70 from 145.125.16.70 (145.125.64.1)
 Origin IGP, metric 10, localpref 200, valid, external, best
```

TR13 ontvangt van beide peers een default route (0.0.0.0/0). Degene met de laagste metric wordt gekozen, dit is hieronder te zien:

```
tr13.amsterdam1#sh ip bgp
BGP table version is 8, local router ID is 145.125.64.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

 Network Next Hop Metric LocPrf Weight Path
* 0.0.0.0 145.125.16.133 20 0 1125 i
*>         145.125.16.65 10 0 1125 i
*> 145.125.64.0/23 0.0.0.0 0 32768 i
```

5.5.3 Meet resultaten

In deze eerste meting wordt gebruik gemaakt van default timers. Het zou theoretisch tussen de twee en drie minuten duren voor dat het netwerk geconvergeerd is. Een link failure wordt gesimuleerd door op switch TS2 port 2/1 down te brengen.

```
ts2.amsterdam1> (enable) set port disable 2/1
```

Vanaf een host op de HvU wordt met een ping naar 145.125.64.1 gemeten hoelang de verbinding onderbroken is.

```
BOFH:~# fping -r 1 -s -l -e 145.125.64.1
```

Het resultaat van deze meting was dat het 167 seconden duurde voordat de het verkeer via het secondaire pad werd geleid. Deze tijd ligt zoals verwacht tussen de 120 en 180 seconden.
In een tweede meting zijn de timers verandert. Op TR11 (de POP router) zijn de timers aangepast naar twee seconden voor de hello-interval en zes seconden voor de hold-timer.

```
Tr11.Amsterdam1(config-router)#neighbor 145.125.16.70 timers 2 6
Tr11.Amsterdam1(config-router)#exit
Tr11.Amsterdam1#clear ip bgp 145.125.16.70
```

Nadat de timers zijn aangepast dient de BGP sessie te worden ge-reset, zodat de parameters opnieuw worden afgestemd op elkaar. De configuratie wijziging kan worden geverifieerd met het volgende commando:

```
Tr11.Amsterdam1#sh ip bgp neighbors 145.125.16.70
<knip>
 Last read 00:00:01, hold time is 6, keepalive interval is 2 seconds
 Configured hold time is 6, keepalive interval is 2 seconds
<knip>
```

De output van dit commando geeft weer wat de timers zijn welke voor deze sessie worden gebruikt. Theoretisch zal de convergentietijd van het netwerk met deze timers moeten liggen tussen de vier en zes seconden. Echter, uit de meting bleek dat het in de praktijk langer duurt. De gemeten convergentietijd is tien seconden. De BGP-debug informatie op TR13 helpt bij het verklaren:

```
Jun 23 10:19:16.504 UTC: %BGP-3-NOTIFICATION: sent to neighbor 145.125.16.65 4/0
```

```
(hold time expired) 0 bytes

Jun 23 10:19:22.504 UTC: BGP(0): 145.125.16.133 update run completed, afi 0, ran for 0ms,
neighbor version 22, start version 23, throt Connections established 8; dropped 8

Jun 23 10:19:22.504 UTC: BGP(0): Revise route installing 0.0.0.0/0 -> 145.125.16.133
to main IP table
```

Zoals is te zien aan de debug informatie duurt het, nadat TR13 heeft ontdekt dat zijn neighbor onbereikbaar is geworden (hold time expired), nog 6 seconden voordat het alternatieve pad wordt geïnstalleerd in de route tabel. Dit heeft vermoedelijk te maken met BGP implementatie op de Cisco 3550emi of de hardware.
Er is nog één meting gedaan met als hello-time drie seconden en als hold-time negen seconden. De totale convergentietijd van deze meting was dertien seconden. Theoretisch zou deze moeten liggen tussen de zes en negen seconden. Wederom bleek uit de debug informatie van TR13 dat het zes seconden duurde voordat de alternatieve next-hop voor 0.0.0.0/0 word geïnstalleerd in de route tabel.

```
Jun 23 10:26:02.737 UTC: %BGP-3-NOTIFICATION: sent to neighbor 145.125.16.65 4/0
(hold time expired) 0 bytes
Jun 23 10:26:08.737 UTC: BGP(0): Revise route installing 0.0.0.0/0 -> 145.125.16.133
to main IP table
```

5.5.4 Conclusie van de meting

Uit de metingen is een interessant gegeven naar voren gekomen. Blijkbaar heeft de 3550emi (layer3 switch) nadat deze heeft ontdekt dat zijn peer onbereikbaar is geworden, altijd zes seconden extra nodig om het alternatieve pad te installeren in de route tabel. In de vorige testen (HvU communicatie lab) werden Cisco2600 routers gebruikt. Bij deze metingen installeerde de routers na het verlopen van de hold-time, onmiddellijk een alternatieve route in de route tabel. Er is dus een verschil tussen de 3550emi en de Cisco 2600 router, wat betreft hoe met BGP wordt omgegaan. Omdat veel klanten die een 'multi-POP' aansluiting hebben, aansluiten met een 3550emi is het belangrijk om met dit gegeven rekening te houden.

5.6 Aanbevelingen voor implementatie

In deze paragraaf zullen een aantal aanbevelingen worden gedaan welke betrekking hebben op de wijziging van de configuratie van de routers van de multi-POP aansluitingen.

5.6.1 Timer keuze

Door de timers aan te passen kan een aanzienlijke verbetering van de convergentietijd worden bereikt. Default staan deze nu op 60 seconden voor de keep-alive-time en 180 seconden voor de hold-time. Deze tijd is erg lang, zelfs twee keer zo langs als wordt aanbevolen in RFC1771[2]. Deze

[2] A Border Gateway Protocol 4 (BGP-4)

RFC zegt ook dat de minimale waarde van de hold-time 3 seconden dient te zijn. Ook voor de KEEPALIVE messages schrijft RFC1771 waarden voor. Aanbevolen wordt dat de KEEPALIVE messages worden verzonden met een interval van 1/3 van de hold-time, echter niet vaker dan 1 keer per seconde. De minimale convergentietijd welke dus bereikt kan worden is tussen de twee en drie seconden. Welke convergentietijden bereikt kunnen worden door het aan passen van de timers, is weergegeven in tabel5.1.

Keep-Alive Interval (seconden)	**Hold-Time (seconden)**	**convergentietijd (seconden)**
1	3	2 < convergentie-tijd < 3
2	6	4 < convergentie-tijd < 6
3	9	6 < convergentie-tijd < 9
4	12	8 < convergentie-tijd < 12
5	15	10 < convergentie-tijd < 15
6	18	12 < convergentie-tijd < 18
7	21	14 < convergentie-tijd < 21
8	24	16 < convergentie-tijd < 24
9	27	18 < convergentie-tijd < 27
10	30	20 < convergentie-tijd < 30
15	45	30 < convergentie-tijd < 45
20	60	40 < convergentie-tijd < 60
25	75	50 < convergentie-tijd < 75
* 30	* 90	60 < convergentie-tijd < 90
** 60	** 180	120 < convergentie-tijd < 180
** Aanbevolen RFC waarden*	*** Default Cisco waarden*	

Tabel 5.1: *Convergetietijden BGP*

De minimale convergentietijd ligt dus tussen de 2 en 3 seconden. Dit kan worden gerealiseerd door de hold-timer op 3 seconden te zetten en de hello-interval op 1 seconde. De timers moeten echter niet 'zomaar' op de kortste waarden worden gezet. Dit 'kan' nadelige gevolgen als resultaat hebben. Wanneer de router bijvoorbeeld de keep-alives niet binnen de hold-time kan verwerken (om wat voor rede dan ook) zal het gevolg zijn dat de prefix steeds aangekondigd wordt en weer terug getrokken wordt. Dit fenomeen wordt route-flapping genoemd.

5.6.2 Flapping / damping

Route flapping kan worden tegen gegaan door het gebruik van damping. Dit vermindert de 'load' welke wordt veroorzaakt door route flapping. Een route flap is een route verandering met als gevolg dat de router zijn route tabel opnieuw moet gaan doorrekenen. Door route damping te gebruiken wordt de invloed van flapping op de router performance verminderd. Dit wordt gedaan door een iedere prefix die 'flapt'[3] een penalty te geven. Wanneer een bepaalde maximale penalty waarde is bereikt zal de prefix voor een bepaalde tijd genegeerd worden.

Route flapping is dus tegen te gaan met behulp route damping. Echter dit heeft ook nadelen, want hiermee wordt bewust de convergentie van het netwerk vertraagd. Beter is om in het geval van ongewenste route flapping de BGP timers anders te kiezen, deze staan dan blijkbaar te kort. Bovendien zal de impact van route flapping op de routers waarschijnlijk geen merkbare invloed

[3]Het steeds verschijnen en verdwijnen van een prefix wordt flapping genoemd

hebben op de performance van de routers. Dit omdat het zal gaan om enkele prefixen en niet om een gehele BGP feed[4].

5.6.3 Voorbeeld configuratie

De BGP timers kunnen globaal worden geconfigureerd of specifiek voor een bepaalde neighbor. Om de timers globaal aan te passen, dient het volgende commando gegeven te worden

```
router bgp 64640
! hold-time 9sec, keep-alive 3sec
timers bgp 3 9
<etc>
```

Om de timers voor een specifieke neighbor aan te passen, dient het volgende commando gegeven te worden:

```
router bgp 64640
! globale timers: hold-time 90sec, keep-alive 30sec
timers bgp 30 90
! specifieke timers: hold-time 9sec, keep-alive 3sec
neighbor 145.145.19.2 remote-as 1103
neighbor 145.145.19.2 timers bgp 3 9
```

5.6.4 Tips voor migratie

De impact van het wijzigen van de configuratie om de timers aan te passen is minimaal. Eventueel kan dit zelfs door de klant zelf gedaan worden. Bij het opzetten van een BGP sessie wordt er namelijk onderhandeld over de diverse parameters. Twee van die paramaters zijn de hold-time en de keep-alive interval. Mochten deze twee parameters tussen de twee peers verschillen, dan wordt de kortste tijd gekozen. De klant kan zijn router dus configureren met een keep-alive van 2 seconden en een hold-time van 6 seconden. De SURFnet router heeft met de default waarde veel hogere tijden. Voor deze betreffende BGP sessie zullen dus de waarden die door de klant zijn geconfigureerd worden gebruikt. In de praktijk blijkt het zo te zijn dat op er op dit moment slecht één klant is die de BGP timers heeft aangepast. Het is verstandig dat SURFnet zichzelf beschermd tegen te veel klanten die zelfstandig te lage timers instellen. Dit is mogelijk door een minimale hold-time te configureren voor een bepaalde sessie, dit gaat alsvolgt:

```
Tr11.Amsterdam1(config)#router bgp 1125
Tr11.Amsterdam1(config-router)#neighbor 145.125.16.70 timers 3 9 ?
  <0-65535>  Minimum hold time from neighbor
  <cr>
```

Op deze manier wordt de load, welke veroorzaakt kan worden door de lage timers, in de hand gehouden. Na een test periode met een aantal klanten kan er voor gekozen worden die timers later globaal te configureren. Eventueel kan dit later ook gedaan worden voor de iBGP sessies.

[4]De volledige BGP table bestaat op dit moment ongeveer uit 142000 prefixen

5.7 Conclusie

Er zijn verschillende manieren om de convergentietijd van de multi-POP aansluiting te reduceren. Dit is mogelijk met nieuwe technieken zoals VRRP of HSRP, maar ook door de huidige techniek (BGP) aan te passen. Dit laatste verdient door de minimale wijzigingen de voorkeur. De convergentietijd kan zo voor de multi-POP aansluitingen relatief eenvoudig terug gebracht worden van 2 tot 3 minuten naar 10 seconden. Dit is dus een verbetering van 12 tot 18 keer vergeleken met de huidige situatie. De convergentietijd is eenvoudig te bereken, deze ligt normaal gesproken tussen (hold-time - hello-time) en de hold-time. Echter uit de testen met een Cisco 3550emi bleek dat er voor deze switch nog 6 seconden bijkomen. Eventueel kan er voor gekozen worden de klanten zelf de timers te laten aanpassen. Wanneer twee peers verschillende waarden voor de timers hebben geconfigureerd, zal er onderhandeld worden over deze timers, uiteindelijk wordt de laagste waarde gekozen. Een advies waarde is 3 seconden voor de hello-interval en 9 seconden voor de hold-timer.

Hoofdstuk 6

Tot Slot

In de afgelopen vier weken is hard gewerkt aan het project "Analystisch Netwerk Project"(ANP). Dit project is voor ons zeer leerzaam geweest. SURFnet heeft ons de kans geboden een inzicht te krijgen hoe moderne netwerken zijn opgebouwd en waar hierbij rekening gehouden dient te worden. We zijn tijdens dit project in aanraking te komen met nieuwe technieken zoals RSTP. Nadat door ons eerst het Spanning Tree Protocol is doorgrond, hebben is een goed beeld van RSTP verkregen. In dit document is uitéén gezet hoe een verbetering van de convergentietijd bereikt kan worden. We zijn tevreden over het bereikte resultaat. Over de onderlinge samenwerking kunnen we kort zijn, deze was goed en erg prettig. SURFnet heeft ons veel vrijheid gegeven in hoe we dit project wilden aanpakken. Veel hebben we eerst getest in het communicatie lab op de HvU. Daarnaast is ons toegang gegeven tot het testnetwerk van SURFnet bij SARA. Hier hebben we alles nogmaals getest, om er zeker van te zijn dat het ook op de apparatuur van SURFnet werkte zoals we verwachten. Al met al zijn wij zeer tevreden over het verloop en zeker ook over het resulaat van dit project.

Utrecht, 29-06-2004
Andree Toonk & Leendert van Doesburg

Bijlage I

Handleiding RSTP versus BGP metingen op de huidige test-omgeving

Om interferenties tussen de BGP en RSTP metingen in de huidige test-opstelling te voorkomen, zijn er een aantal minimale configuratie-wijzigingen noodzakelijk om te wisselen tussen deze twee metingen.

RSTP-meting

Voor het meten van convergentietijden bij RSTP, wordt er vanaf een willekeurige host op het SURFnet met (f)ping 'gepingt' naar de destination-host welke een interface is op TR13 met het ip-adres 145.125.16.70

Op TR13 wordt hiervoor handmatig een statische quad zero (default) route geconfigureerd. Er wordt voor de RSTP-meting dus geen gebruik gemaakt van de (mogelijk) aanwezige default BGP-route, om te voorkomen dat door langdurige verbrekingen van verbindingen tussen switches de BGP peers down gaan en de betreffende default route wordt teruggetrokken.

Het commando dat voor de quad zero route op de TR13 gegeven moet worden is:

```
ip route 0.0.0.0 0.0.0.0 145.125.16.65
```

Door poorten op de switches 1,2 en 3 down te brengen, kunnen korte onderbrekingen in de ping waargenomen worden.

Voorbeeld van een fping:

```
leendert@BOFH:~$ fping -s -l -r 1 -p 250 145.125.16.70
145.125.16.70 : [0], 84 bytes, 2.17 ms (2.17 avg, 0% loss)
145.125.16.70 : [1], 84 bytes, 8.80 ms (5.48 avg, 0% loss)
145.125.16.70 : [2], 84 bytes, 1.56 ms (4.17 avg, 0% loss)
145.125.16.70 : [3], 84 bytes, 1.55 ms (3.52 avg, 0% loss)
145.125.16.70 : [10], 84 bytes, 3.55 ms (3.52 avg, 54% loss)
145.125.16.70 : [11], 84 bytes, 1.54 ms (3.19 avg, 50% loss)
145.125.16.70 : [12], 84 bytes, 1.53 ms (2.95 avg, 46% loss)

145.125.16.70 : xmt/rcv/%loss = 13/7/46%, min/avg/max = 1.53/2.95/8.80

       1 targets
```

```
       1 alive
       0 unreachable
       0 unknown addresses

       0 timeouts (waiting for response)
      13 ICMP Echos sent
       7 ICMP Echo Replies received
       0 other ICMP received

 1.53 ms (min round trip time)
 2.95 ms (avg round trip time)
 8.80 ms (max round trip time)
        3.157 sec (elapsed real time)
```

13-7=6 pakketten welke iedere 250 ms verstuurd worden, zijn verloren gegaan na het down-brengen van een interface op één van de switches. Dit resulteert in een onderbreking van 6 * 0,250=1,5 seconden.

BGP-meting

Voor deze meting wordt een host van een netwerk gepingt dat door TR13 wordt geadverteerd. Hiervoor wordt gebruikt: 145.125.64.1

Met het commando:

```
no ip route 0.0.0.0 0.0.0.0 145.125.16.65
```

wordt op de TR13 de quad zero route verwijderd. Eventueel kan daana de route-table bekeken worden of deze een default-route via BGP geleerd heeft. Om vervolgens een indirecte link-faulure te realiseren op de primaire link (145.125.16.64/28 op VLAN 1), moet eerst de redundantie door RSTP uitgeschakeld worden. Dit wordt gedaan door de verbinding tussen ts2 en ts3 down te brengen:

```
ts2.amsterdam1> (enable)set port disable 2/2
```

Het creëren van een indirecte link-failure op de primaire verbinding:

```
ts2.amsterdam1> (enable) set port disable 2/1
```

Voorbeeld van een fping:

```
leendert@BOFH:~$ fping -s -l -r 1  145.125.64.1
145.125.64.1 : [0], 84 bytes, 1.56 ms (1.56 avg, 0% loss)
145.125.64.1 : [1], 84 bytes, 1.74 ms (1.65 avg, 0% loss)
145.125.64.1 : [15], 84 bytes, 1.62 ms (1.64 avg, 81% loss)
145.125.64.1 : [16], 84 bytes, 1.55 ms (1.61 avg, 76% loss)
145.125.64.1 : [17], 84 bytes, 1.53 ms (1.60 avg, 72% loss)
145.125.64.1 : [18], 84 bytes, 1.54 ms (1.59 avg, 68% loss)
```

```
145.125.64.1 : [19], 84 bytes, 1.50 ms (1.57 avg, 65% loss)
145.125.64.1 : [20], 84 bytes, 1.74 ms (1.59 avg, 61% loss)
145.125.64.1 : [21], 84 bytes, 1.52 ms (1.58 avg, 59% loss)

145.125.64.1 : xmt/rcv/%loss = 22/9/59%, min/avg/max = 1.50/1.58/1.74

       1 targets
       1 alive
       0 unreachable
       0 unknown addresses

       0 timeouts (waiting for response)
      22 ICMP Echos sent
       9 ICMP Echo Replies received
       0 other ICMP received

 1.50 ms (min round trip time)
 1.58 ms (avg round trip time)
 1.74 ms (max round trip time)
       21.322 sec (elapsed real time)
```

Hieruit blijkt het netwerk 13 seconden niet bereikbaar was.

Hoofdstuk 7

Bronvermelding

Spanning Tree
http://www.cisco.com/pcgi-bin/Support/browse/psp_view.pl?p=Technologies:Spanning_Tree&viewall=true

Understanding and Configuring Spanning-Tree Protocol (STP) on Catalyst Switches
http://www.cisco.com/warp/public/473/5.html

VLAN Load Balancing Between Trunks Using the Spanning-Tree Protocol Port Priority
http://www.cisco.com/warp/public/473/15.html

Spanning Tree Protocol Problems and Related Design Considerations
http://www.cisco.com/warp/public/473/16.html

Understanding Spanning-Tree Protocol Topology Changes
http://www.cisco.com/warp/public/473/17.html

Understanding and Configuring Backbone Fast on Catalyst Switches
http://www.cisco.com/warp/public/473/18.html

Understanding and Configuring the Cisco UplinkFast Feature
http://www.cisco.com/warp/public/473/51.html

Spanning Tree Portfast BPDU Guard Enhancement
http://www.cisco.com/warp/public/473/65.html

Understanding and Configuring the Unidirectional Link Detection Protocol Feature
http://www.cisco.com/warp/public/473/77.html

Spanning-Tree Protocol Enhancements using Loop Guard and BPDU Skew Detection Features
http://www.cisco.com/warp/public/473/84.html

Understanding and Tuning Spanning Tree Protocol Timers
http://www.cisco.com/warp/public/473/122.html

Unicast Flooding in Switched Campus Networks
http://www.cisco.com/warp/public/473/143.html

Understanding Rapid Spanning Tree Protocol (802.1w)
http://www.cisco.com/warp/public/473/146.html

Understanding Multiple Spanning Tree Protocol (802.1s)
http://www.cisco.com/warp/public/473/147.html

Technically Speaking: New IEEE 802.1w/1s Protocols
http://www.cisco.com/en/US/about/ac123/ac114/ac173/ac222/about_cisco_packet
_department09186a0080142dfa.html

Configuring Spanning Tree PortFast, UplinkFast, BackboneFast, and Loop Guard
http://www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/sw_7_2/confg_gd/stp_enha.htm

BGP, the Border Gateway Protocol / Advanced Internet Routing
http://www.bgp4.as/tools

C-BGP - An efficient BGP simulator
http://cbgp.info.ucl.ac.be/

BGPexpert.com On-line BGP Resources and Information
http://www.bgpexpert.co

Building Reliable Networks with the Border Gateway protocol
O'REILLY Iljitsch van Beijnum ISBN: 0-596-00254-8

Lijst van figuren

Lijst van tabellen