

Departamento de Engenharia Informática e de Sistemas

**Protocolos Spanning Tree**

Rafael Tavares Ribeiro - 2019131989

Guião laboratorial no âmbito da Licenciatura de Engenharia Informática, para a unidade curricular de Disponibilidade e Desempenho, lecionada pelo Docente Luís Eduardo Faria dos Santos

dezembro de 2021

# Índice

[Abreviaturas **3**](#_Toc68729827)

[Lista de figuras **4**](#_Toc68729828)

[1. Introdução **5**](#_Toc68729829)

[2. Visão geral do MTConnect **6**](#_Toc68729831)

[2.1. Conceitos gerais **6**](#_Toc68729832)

[2.2. Conceitos do adaptador **7**](#_Toc68729834)

[2.3. Conceitos do Agente **8**](#_Toc68729835)

[3. Como funciona a comunicação **9**](#_Toc68729836)

[4. Problemas empresariais e tipo de dados **11**](#_Toc68729837)

[4.1. Alertas **11**](#_Toc68729838)

[4.2. Disponibilidade e uso do equipamento **11**](#_Toc68729839)

[4.3. Tipo de dados **11**](#_Toc68729840)

[5. Conclusão **12**](#_Toc68729841)

[6. Bibliografia **13**](#_Toc68729842)

# Abreviaturas

**IoT:** Internet of Things

**XML:** Extensible Markup Language

**Big Data:** conceito de grande volume de dados

**HTTP:** Hypertext Transfer Protocol

**TCP/IP:** Transmission Control Protocol/Internet Protocol

**CNC:** Computer Numerical Control

**OEE:** Overall Equipment Effectiveness

# Lista de figuras

[Figura 1 - Estrutura MTConnect 6](#_Toc69644582)

[Figura 2 - Equipamento com adaptador 7](#_Toc69644583)

[Figura 3 - Equipamento com MTConnect de origem 7](#_Toc69644584)

[Figura 4 - Equipamento com unidade de conexão 8](#_Toc69644585)

[Figura 5 - Exemplo de comunicação Server-Agent 9](#_Toc69644586)

# Introdução

O presente relatório tem o propósito de servir como um guião laboratorial (provisório) do workshop realizado à unidade curricular de Disponibilidade e Desempenho. O tema escolhido foi um estudo comparativo entre os diversos protocolos existentes de Spanning Tree e o objetivo deste guião, ainda que provisório, é que sirva de base para que qualquer um consiga replicar as experiências realizadas ao longo do estudo. Para que estas experiências sejam possíveis de serem concebidas a partir do acompanhamento deste guião, será necessário possuir as seguintes ferramentas: simulador GNS3 com uma imagem switch como por exemplo a i86bi\_linux\_l2-adventerprise-ms.high\_iron\_20170202.bin (ou idêntica) e ainda com imagens de terminais como VPCS ou ipterm. Aliada ao GNS3 será também necessário o uso da ferramenta Wireshark para as capturas de tráfego. Com isto, é possível replicar todo o trabalho presente neste guião.

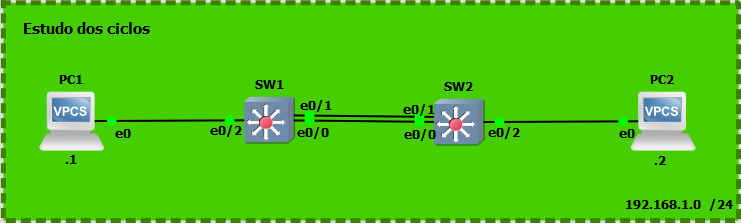
Os protocolos Spanning Tree foram criados quando existiu a necessidade de corrigir os loops criados numa rede servida por switchs. Estes loops são criados devido à redundância que uma rede necessita de ter para o melhor funcionamento da mesma e impedem que pacotes cheguem a determinados destinos, prejudicando assim a disponibilidade de uma rede. Existem diversos protocolos: STP, RSTP (respetivas alternativas cisco PVST+ e Rapid PVST+) e ainda o MSTP. Estes protocolos definem uma tabela de forwarding nos switches e bridges de modo que, uma rede constituída por estes equipamentos, fique livre dos ciclos.

Neste documento, vão ser relatadas todas as experiências e conclusões tiradas das mesmas. Começando por compreender como os ciclos são realmente criados, passando pelo estudo mais exaustivo de cada um dos protocolos referidos anteriormente e terminando com uma conclusão breve do que se conseguiu retirar dos aspetos estudados. De notar que este guião não tem como objetivo principal, tirar todas as conclusões comparativas e/ou especificas de cada protocolo nem explicar pormenorizadamente os vários protocolos.

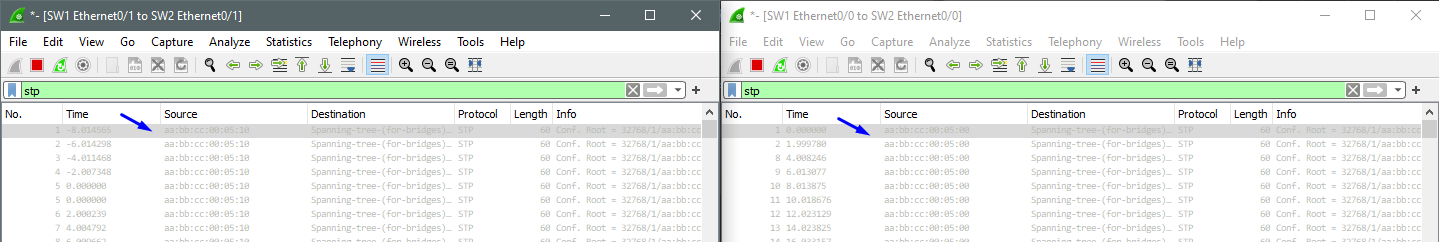
# Estudo dos ciclos

Não é preciso muito para termos ciclos numa topologia. Se existirem dois switchs ligados entre si a partir de duas ligações, temos um ambiente propício à criação dos loops, se não existir nenhum protocolo a correr que evite isso.

Para uma melhor perceção de como são realmente criadas estas menos valias numa rede, foi criada uma topologia simples que segue a ideia referida anteriormente: dois switchs dois duas ligações entre si, tal como podemos ver na imagem seguinte.



Sabendo que o protocolo que vem administrado por defeito nos switchs é o STP, inicialmente o mesmo não foi imediatamente desligado. A partir da consola do switch 1 e com a ajuda do comando show spanning-tree, conseguimos ver que o SW1 é a Root Bridge da rede. Com o Wireshark, capturou-se tráfego STP nas duas ligações e, como está representado na imagem seguinte, aparecem pacotes STP. Também é possível pingar do PC1 para o PC2 sem problemas.



De seguida, com o comando no spanning-tree vlan 1-1014 em ambos os switchs, desativou-se o protocolo para todas as Vlans possíveis

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamente

Posteriormente, realizou-se um ping entre os dois PCs da rede e conseguimos perceber que a comunicação nunca é estabelecida. A imagem seguinte demonstra isso.

Uma imagem com texto

Descrição gerada automaticamente

Capturou-se novamente tráfego nas duas ligações (desta vez sem nenhum filtro) e conseguimos perceber que existiu uma troca contínua de pacotes ARP, nos dois caminhos, enviadas pelo mesmo equipamento. Estas mensagens ocorrem com intervalos de tempo inferiores a 1 segundo. Isto acontece porque, uma vez que o STP foi desligado, o SW1 recebe o pedido ARP e envia-o para o SW2 por uma das interfaces. O SW2 recebe esse pedido e volta a reencaminhá-lo pela 2ª interface, ou seja, pela outra ligação. Estas trocas de mensagem estão sempre e ocorrer e o SW2 nunca consegue entregar os pacotes ao PC2, visto que recebe por uma interface (de uma das ligações) e envia por outra que está também ligada ao SW1. Aqui se formam então os ciclos que congestionam uma rede e impossibilitam, por exemplo, a comunicação entre equipamentos. Na imagem seguinte, a partir da captura referida anteriormente, conseguimos ter uma amostra do que acontece quando se desliga o protocolo STP e se começam a criar ciclos de envio e receção dos pacotes ARP.

Uma imagem com texto, captura de ecrã, interior

Descrição gerada automaticamente

# Protocolo STP

* 1. **Conteúdo teórico STP**

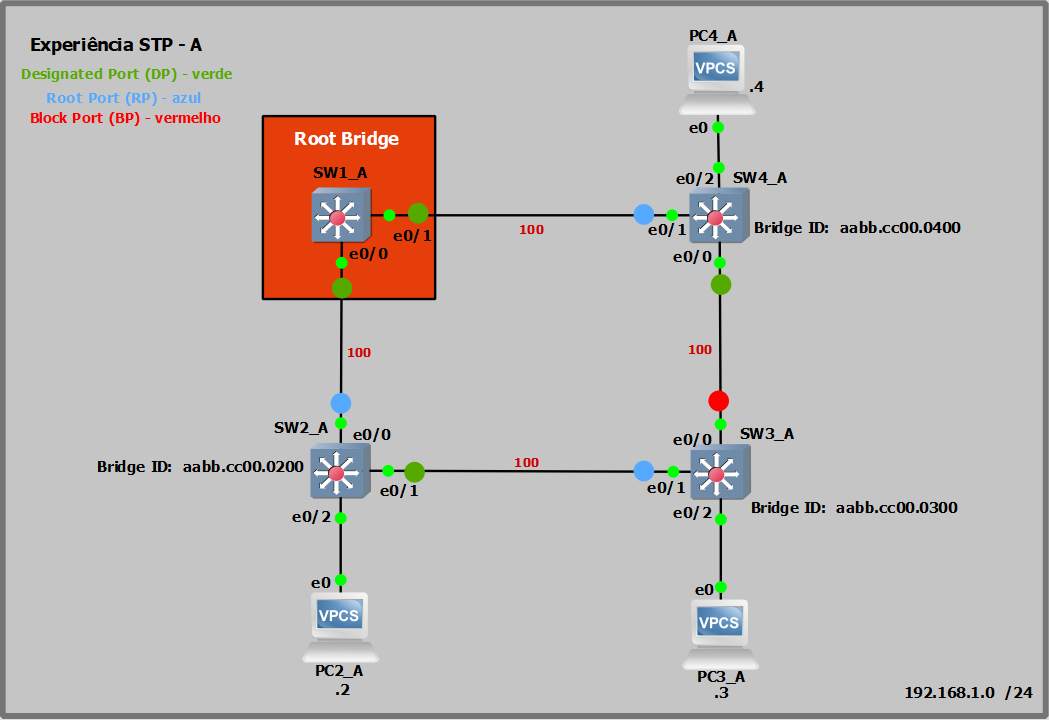
Para a redundância de caminhos existir e a criação de loops ser evitada em simultâneo, o STP define uma espécie de árvore que abrange todos os switches de uma rede e este aspeto é comum entre todos os protocolos. Este protocolo força a que certos caminhos fiquem no estado de *standby* (blocking) e deixa outros no estado de *forwarding*. Se uma das ligações ficar indisponível, o protocolo adapta a rede de maneira que, a porta que estava bloqueada entre em ação, assegurando assim a continuidade dos serviços prestados pela rede.

Existem 3 funções às quais as portas podem estar atribuídas: *designated port, root port e alternate port*. Associado a isto, as portas também podem ficar configuradas em 4 estados*: blocking, listening, learning e forwarding.*

Todos os switchs da rede elegem uma Root Bridge que vai ser o switch mais importante da rede. É a partir dele que todas as decisões na rede vão ser tomadas, tais como a quem e onde colocar os estados e funções referidos anteriormente. Diferentes Vlans devem ter diferentes Root Bridge. A eleição deste switch pode ser feita a partir do gestor de rede ou na escolha automática do switch com Bridge ID menor, constituída pela prioridade do switch e pelo MAC Address. Nestes protocolos, quanto menor a prioridade de um equipamento, melhor. Tudo isto é possível devido à troca de informações que os switchs fazem entre si a partir das Bridge Protocol Data Units (BPDUs). Cada switch compara as BPDUs que recebe com aquelas que envia. Quando um switch recebe uma BPDU com um Root ID inferior ao seu, atualiza o seu parâmetro do Root ID até todos os equipamentos chegarem a um acordo de qual é o switch com um identificador menor e aí consideram esse switch como a Root Bridge.

* 1. **Experiência STP - A**

Com a ajuda de uma topologia criada no GNS3, conseguimos perceber melhor como este protocolo funciona. Entendo bem o STP, obtemos as bases mais importantes para que os restantes protocolos sejam mais fáceis de serem aprendidos. Na imagem seguinte conseguimos ver a topologia realizada. De notar que não foi alterada a prioridade em nenhum switch, visto que o objetivo era perceber como é que o STP vinha configurado nos switchs e como realmente funciona. A prioridade pode ser alterada, de maneira a escolhermos qual o switch que funciona como Root Bridge, mas não é relevante para realmente entender como este protocolo funciona. As questões mais relevantes postas em estudo foram por exemplo: como o protocolo escolhe a Root Bridge, como é que se atribuem as características às diferentes portas, como é feita a comunicação entre os switchs.



Com o comando show spanning-tree, conseguimos ver que o SW1 é a Root Bridge, visto que, apesar de ter a mesma prioridade que os outros switchs, contém um MAC Address menor (aabb.cc00.0100 < .200 < .300 < .400), como conseguimos ver de seguida. Assim, percebemos que se a escolha do switch principal for feita automaticamente, é escolhido o que tiver menor prioridade e depois o que tiver menor MAC.

Uma imagem com texto

Descrição gerada automaticamente

As portas foram estabelecidas segundo as designações que a imagem inicial apresenta pelos círculos coloridos (azul para Root, verde para Designated e vermelho para Alternate, bloqueada). Com o comando show spanning-tree, conseguimos ver as funções e estados das interfaces. Sabendo que as mesmas são escolhidas perante a posição que a Root Bridge ocupa e a partir dos custos das ligações necessárias para se chegar até ao SW1 (quanto menos custo, melhor), podemos resumir da seguinte maneira o porquê destas escolhas por parte do protocolo:

* As do SW1 são ambas Designated porque é a Root Bridge e assim, as interfaces dela ficam com esta designação.
* No SW2 temos a e0/0 como Root Port porque esta é a porta do switch que acarreta menos custo (100) para chegar até ao SW1. A e0/1 ficou como designated pela mesma lógica.
* No SW4 a lógica é a mesma do SW2. A e0/1 é a interface que menos custo total leva para chegar à root bridge e a e0/0 é a melhor porta da ligação SW4-SW3.
* Por fim, no SW3, temos empate para eleger a Root Port, visto que ambas as interfaces levam o mesmo custo (200) para chegar ao SW1.
  + Nesta situação, escolhemos a interface que estabelece uma ligação ao vizinho que tem menor Bridge ID. Visto que tanto o SW2 como o 4 têm a prioridade igual, então o aspeto a ter em conta é o switch com menor MAC da Bridge ID, que neste caso é o SW2.
  + Ficamos com a e0/1 como Root Port e a e0/0 como porta bloqueada visto que é a única porta que não pode levar outra designação que não esta. Esta porta ficará como de backup para uma falha.

Entendendo o porquê da Root Bridge e das portas serem escolhidas de tal maneira, conseguimos levar isto para os outros protocolos, visto que se baseiam na mesma ideia.

* + 1. **Capturas e injeção de falhas**

Com a ferramenta Wireshark, para melhor se perceber a troca de pacotes BPDUs entre os switchs e entender também a estrutura destas mensagens, realizou-se uma captura entre SW1-SW4 e SW4-SW3. A primeira é exemplificada na imagem seguinte.

Uma imagem com texto

Descrição gerada automaticamente

Estas mensagens são enviadas de 2 em 2 segundos (Hello Time: 2) e nelas conseguimos ver:

* **Root ID:** contém a bridge priority e o MAC Address da Root Bridge.
* **Root Path Cost:** apresenta o custo da ligação entre a Root Bridge e o switch que enviou o BPDU
* **Bridge ID:** contém a bridge priority e o MAC Address do switch
* **Message age**: representa a distância em switchs até à Root Bridge. Cada switch que não seja root e que envie uma BPDU incrementa este valor em 1.
* **Max age:** que representa o tempo máximo possível de existir antes que uma porta de uma bridge guarde as informações da BPDU.
* **Hello time:** intervalo de tempo entre as mensagens enviadas

Neste caso, a mensagem teve origem na interface e0/1 do SW1 e foi enviada para o grupo multicast de Spanning-Tree. Contém o Root Id do switch 1 que é o Root Bridge da rede e também os identificadores do switch que enviou a mensagem, o SW1. Neste caso estes dois aspetos coincidem, visto que o pacote foi enviado pelo SW1 que é a Root Bridge.

De seguida, experimentou-se injetar uma falha na porta e0/1 do SW1 (fazendo o comando shut nessa interface) para entender como o protocolo se adapta a uma mudança na rede. Quando se faz isto, ao fazer uma captura de tráfego entre o SW1 e o S24, conseguimos ver que os pacotes STP deixam de ser enviados durante, aproximadamente, 20 segundos. Sendo assim, um switch demora 20 segundos a perceber que não foi enviada nenhuma BPDU do vizinho, ou seja, ao final de 10 BPDUs não recebidas, como podemos nas duas imagens seguintes: a primeira representa o último pacote enviado antes da falha e conseguimos ver que a flag “topology change” encontra-se no estado “no”. Na segunda imagem vemos que é o primeiro pacote enviado depois da falha e já conseguimos ver que a flag “topolgy change” encontra-se a “yes”, visto que houve uma alteração na topologia. Esta flag é colocada a yes pela Root Bridge quando esta recebe uma TCN (explicado abaixo).

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

Se fizermos capturas nas restantes 3 ligações, conseguimos ver que é enviada, em todas elas, uma mensagem que segue os mesmos aspetos de origem e destino, mas que no entanto, apenas contém a informação de que ocorreu uma mudança na topologia, para assim os restantes switches conseguirem reajustar as suas portas. Esta mensagem é uma Toplogy Change Notification, uma BPDU especial que não contém qualquer tipo de informação e é enviada em direção à Root Bridge pelo switch que deteta a mudança na rede, para assim o avisar do que aconteceu.

Uma imagem com mesa

Descrição gerada automaticamente

Depois destes 20 segundos, o switch que tem a porta de backup (a que está no estado de bloqueada), coloca a mesma em Listening durante 15 segundos e em seguida no estado de Learning também durante 15 segundos. Só após estes dois estados é que a interface passa para Forwarding e o serviço é restabelecido. Podemos concluir que o STP demora 50 segundos a reagir a uma falha e a colocar a rede novamente disponível.

Com isto, se o comando sh spanning-tree for feito em todos os switchs, podemos ver que alterações as portas sofreram:

* **SW4:** a Root Port passa a ser a e0/0, visto que a anterior (e0/1) está ligada à porta da root bridge que se desligou. A e0/1 passa a ser designated, visto que agora o caminho tem de passar por essa porta e dar a volta à topologia.
* **SW3:** a e0/0 do SW3 que se encontrava bloqueada, passou a estar no estado de designated root
  + É que aqui que o STP trabalha, mantendo a redundância e assegurando a resolução de falhas na rede.
* **SW1 e SW2:** o switch 1 fica sempre com as suas portas a DP e o 2, como está ligado à Root, também mantém o que tinha anteriormente.

A próxima experiência realizada foi um ping do PC4 para o PC2 e a injeção da mesma falha anterior. Inicialmente o PC4 consegue pingar sem problemas o PC2. Os echo request e reply vão da interface e0/1 do SW4, visto que é esta a Root Port (conseguimos comprovar isso pois obtivemos esse mesmo tráfego entre o SW4 e o SW1, como vemos na imagem abaixo) e por fim chegam à e0/0 do SW2.

Quando é efetuada uma falha na interface e0/1 do SW1 e fazendo uma captura de tráfego entre o SW4-SW1 e o SW3-SW4, conseguimos ver que a primeira captura (a da esquerda) deixa de detetar tráfego ICMP e passado algum tempo, é a captura sobre a ligação SW3-SW4 que interceta tráfego, visto que o switch 4 muda a sua Root Port para a e0/0. Como referido anteriormente, o PC4 demora 50 a voltar a apresentar comunicação para com o PC2.

Uma imagem com mesa

Descrição gerada automaticamente

Nas imagens seguintes também conseguimos ver, com a ajuda do comando sh spanning-tree, a passagem pelos 2 estados (listening e learning, depois dos 20 segundos sem BPDUs), que a porta que estava bloqueada no SW3 sofreu.

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamente

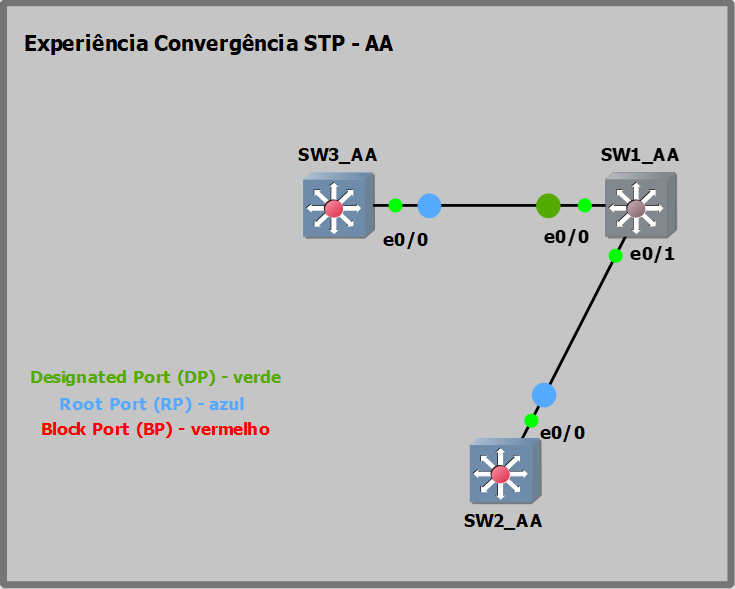
METER ISTO TIPO LADO A LADO, MAS VER COMO SE LEGENDA ISTO, NEM QUE SE TIRE PRINT DE NOVO

Com isto, podemos dizer que o protocolo spanning tree consegue adaptar-se perante falhas na rede, mudando as configurações das portas de cada router perante a falha ocorrida. As portas que se encontravam bloqueadas, entram em ação para assumirem o papel de DP e assim é possível a continuação dos serviços, no entanto há sempre uma demora relativamente grande associada a este processo. Os custos dos caminhos de cada switch até à root bridge acabam por mudar, visto que as portas do switches mudam para outras funções e os equipamentos têm de usar outras ligações para lá chegarem. No subcapítulo seguinte, iremos perceber melhor como funciona a convergência do STP.

* 1. **Experiência Convergência STP – AA**

Visto que a experiência anterior serviu mais como base para se entender os aspetos mais importantes (muitos deles gerais a todos os protocolos), realizou-se outra experiência, mais simples, para melhor se perceber como funciona a convergência no STP, ou seja, como são os mecanismos que os equipamentos adotam e como os mesmo comunicam com a Root Bridge quando uma alteração na rede existe.

Como é possível de se ver na imagem seguinte, partiu-se de uma topologia com apenas 3 switchs onde inicialmente, não estavam todos ligados entre si, não existindo a possibilidade de acontecerem ciclos e assim não há a necessidade de existirem portas bloqueadas. As funções iniciais das interfaces estão representadas também. Com o comando spanning-tree vlan 1 root primary, colocou-se o SW1 como Root Bridge.



De seguida, adicionou-se uma ligação entre o switch 2 e 3, capturou-se tráfego nas 3 ligações com o Wireshark e usou-se o comando sh spanning-tree nos 3 switchs, logo no momento da criação da nova ligação. (MTER ISTO?) Na captura realizada entre o SW2 e o SW1, conseguimos ver que o SW2 envia uma TCN para o SW1, visto que foi o switch 2 que detetou a mudança na rede e avisa então a Root Bridge.

Uma imagem com texto

Descrição gerada automaticamente

A Root Bridge recebe este TCN e responde logo de seguida com uma BPDU onde a flag da topology change acknowledgment vem a “yes”, para quem enviou o aviso de que a rede mudou, saber que a Root Bridge já tem conhecimento disso.

Uma imagem com texto

Descrição gerada automaticamente

A porta e0/1 do SW3 ficou como designated e a e0/1 do SW2 como alternate, ou seja, bloqueada. Isto aconteceu porque o SW3 mandou BPDUs para o SW2 como informações melhores na Bridge ID, neste caso com um identificador mais baixo (como podemos ver na imagem seguinte). Sendo assim, o SW2 percebe que “perde” em relação ao switch 3 e bloqueia a sua porta.

Uma imagem com mesa

Descrição gerada automaticamente

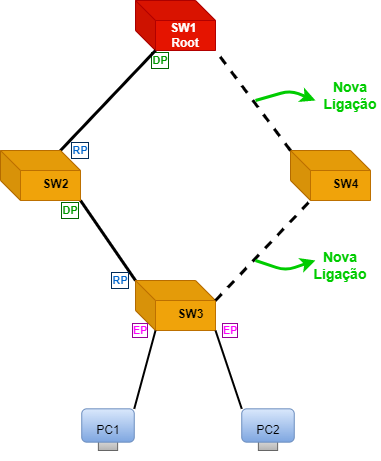
O STP resolve perfeitamente o problema e consegue calcular, eficientemente, as novas alterações na rede. No entanto, este processo demora sempre imensos segundos. Assim, como iremos ver no capítulo seguinte, foi criado o protocolo RSTP, que é muito mais rápido.

1. **Protocolo RSTP**
   1. **Conteúdo Teórico RSTP**

O protocolo Rapid Spanning-Tree é uma evolução do STP. Melhora o tempo de convergência ao reduzir para apenas 3 Hello BPDUs para a rede perceber que tem de se adaptar perante uma alteração. O que torna este protocolo mais rápido, é que o mesmo é capaz de confirmar rapidamente que uma porta pode passar para o estado de forwarding sem depender de qualquer temporizador.

As características do protocolo RSTP não diferem muito do anterior. Temos novos tipos de papéis para as portas: Alternate e Backup. As mesmas podem estar nos estados de Learning, Forwarding e Discarding, que substitui os estados de Disabled, Blocking e Listening visto que é usado para também aprender o MAC Address da interface e impede o encaminhamento. No entanto, temos diferenças no preenchimento do campo *Flag* e no envio das BPDUs. As portas com o papel de DP e RP continuam a desempenhar a mesma função que no STP. No entanto, a Alternate port que tínhamos no protocolo anterior, pode ser Backup port (backup à DP) ou Alternate port (backup à RP). Estas funções, tal como o STP, são determinadas perante o conteúdo trocado nas BPDUs.

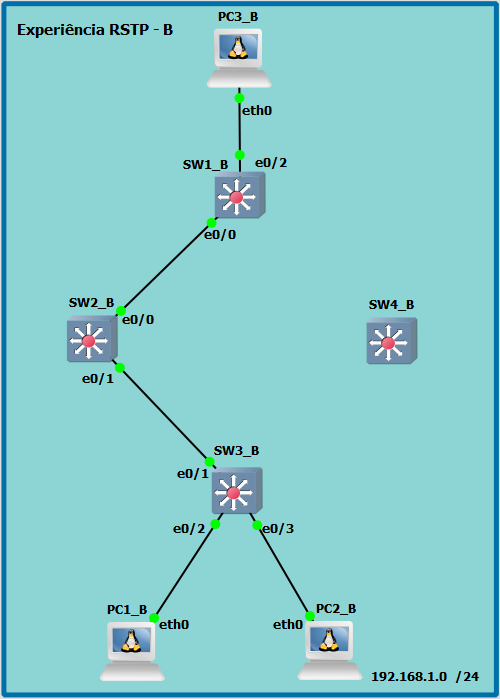
A convergência no RSTP é baseada no mecanismo de proposal/agreement. Um mecanismo de negociação entre os switchs que aliado a um processo de synchronization, os equipamentos entendem a melhor maneira de colocar as suas portas. Com a ajuda do esquema seguinte, onde temos a Root Bridge e as portas representadas, conseguimos perceber melhor estes processos.



O processo de proposal/agreement inicia-se quando, por exemplo, é estabelecida uma nova ligação e acontece entre todos os switchs, por exemplo o switch 1 e 4. No entanto, antes disso, ambos colocam as portas que fazem parte dessa nova ligação no estado de discarding. De seguida, o SW1 envia uma proposta ao SW4 em forma de uma mensagem BPDU com as suas informações. Este propose tem como intuito o SW1 mostrar que é a Root Bridge colocar a sua porta no modo DP e indica ao SW4 que deve colocar a sua no modo Root. Quando o SW4 recebe a BPDU, percebe que é melhor do que aquela que ele poderia oferecer. No entanto, antes de aceitar a proposta, passa primeiro pelo processo de sychronization e tem o cuidado de colocar todas as suas portas, que não sejam edge ports, bloqueadas assegurando que nenhum loop seja criado durante a negociação entre os switchs. Depois disto, o SW4 aceita a proposta do SW1 e envia-lhe a mensagem de agreement e a nova ligação fica estabelecida com as seguintes decisões: DP do lado do SW1 e RP do lado do SW4. Estes processos são idênticos para as restantes ligações e no caso da SW3-SW2, o processo de negociação leva a que os switchs entendam que a porta bloqueada tem de fica do lado do SW3, evitando assim a criação de loops. No subcapítulo seguinte, iremos analisar isto experiencialmente.

* 1. **Experiência RSTP - B**

Criou-se uma experiência idêntica (imagem seguinte) à do esquema anterior para perceber a rapidez do RSTP e como funciona a sua convergência. Foi necessário colocar todos os switchs no modo RSTP, como o comando spanning-tree mode rapid-pvst. Em todas as interfaces dos switchs que se ligavam a outros switchs, colocou-se no modo trunk com os comandos switchport trunk encapsulation dot1q e switchport mode trunk e colocou-se ligações point-to-point com o comando spanning-tree link-type point-to-point. Já nas portas que se ligam a terminais, colocou-se no modo access com o comando switchport mode access e em portfast com o spanning-tree portfast. Estas portas ficam do tipo edge. O SW1 é a Root Bridge. Como ainda não estabelecemos ligações do SW4 para o SW1 e SW3, ainda não temos a possibilidade existirem loops. Assim, a interface do SW1 encontra-se na função de designated, tal como a e0/1 do SW2. A e0/0 do 2º e 3º switch está em root.



Foi feita uma captura de tráfego entre o SW2-SW3. Segundo a imagem seguinte, vemos são enviados de cada vez, dois pacotes STP ao mesmo tempo. Um deles continua com o mesmo destino que o protocolo normal de Spanning Tree. O outro pacote é enviado também para um grupo multicast/Broadcast do pvst+ (protocolo explicado mais à frente). A mensagem teve origem na e0/1 do SW2.

Uma imagem com texto

Descrição gerada automaticamente

Estes dois pacotes apresentam o mesmo conteúdo. Inicialmente conseguimos ver que a versão RSTP está identificada. Tal como referido anteriormente, este protocolo apresenta mais flags. Como a topologia ainda não sofreu nenhuma alteração, as flags de topology change (comuns ao STP) encontram-se a “no” e as de Agreement e Proposal também se encontram a “no”. Já a flag que indica a função da porta por onde foi enviada a mensagem, encontra-se em designated, visto ser esta a função da e0/1 do switch que enviou o pacote. As flags de Learning e Forwarding encontram-se a “yes” e assim se mantiveram enquanto não houve uma alteração à rede. Vemos isto na próxima imagem.

Uma imagem com texto

Descrição gerada automaticamente

Continuando a analisar o pacote, vemos a identificação da Root Bridge e da Bridge que enviou a BPDU. O custo do caminho até ao switch principal é apresentado e vemos a message age, o max age e o Hello Time que é igual ao protocolo estudado anteriormente.

Seguidamente, foi adicionado um novo switch, o SW4, que se ligou ao SW1 e 3. No momento da adição do equipamento, foi introduzido o comando show spanning-tree em todos os switchs e capturou-se tráfego STP em todas as ligações.

* + 1. **Capturas e injeção de falhas**

Analisando o SW3 conseguimos ver que, inicialmente, como a interface e0/0 ainda não se encontrava com uma ligação estabelecida, a mesma passa para um estado de blocking, enquanto a Root port ainda se manteve na e0/1.

Uma imagem com texto

Descrição gerada automaticamente

Como explicado anteriormente, existiu a negociação do SW4 com o SW3 resultando em que a DP ficasse do lado deste último switch. Portanto a e0/0 do sw3 ficou com o papel de Root port. Como o switch 3 também teve de fazer o proposal/agreemente com o SW2, conseguimos ver que o SW3 fez a sincronização durante a negociação. Com isto, bloqueou todas as portas que não fossem edge ports (e0/2-3, imagem da esquerda). Visto que o SW2 apresentava uma BPDU melhor, ficou decidido que a DP ficaria do lado desse switch (imagem da direita) e a porta Alternate ia lado do SW3, para impedir os loops.

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

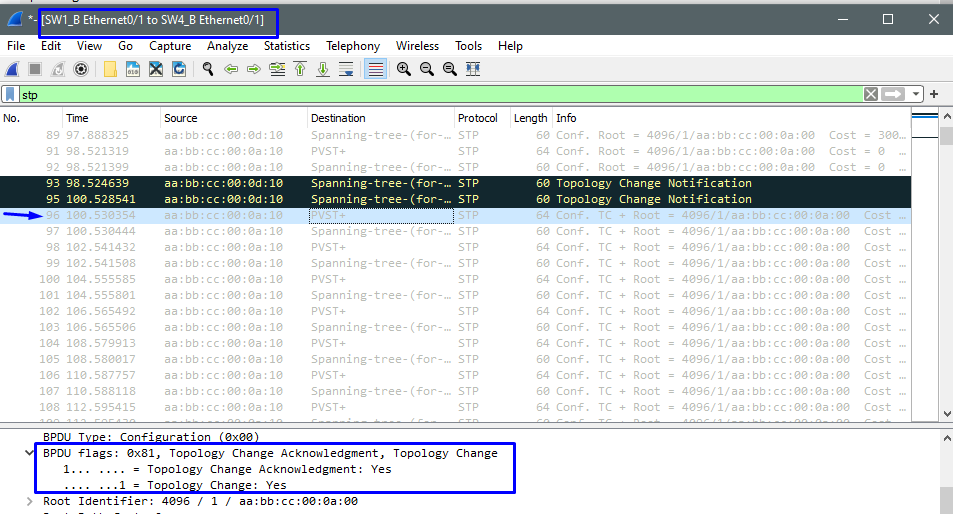
Descrição gerada automaticamente

Se analisarmos as ligações SW1-SW4 e SW4-SW3, aquando da introdução do novo switch, vemos a presença de pacotes STP do tipo Topology Change Notification nas duas capturas. Estes pacotes são enviados para a Root Bridge por quem deteta a alteração na rede. Neste caso, foi o SW3 primeiramente a detetar essa mudança e enviou a TCN para o SW4 que posteriormente enviou para a Root Bridge. A próxima imagem relata exatamente isto.

Uma imagem com texto

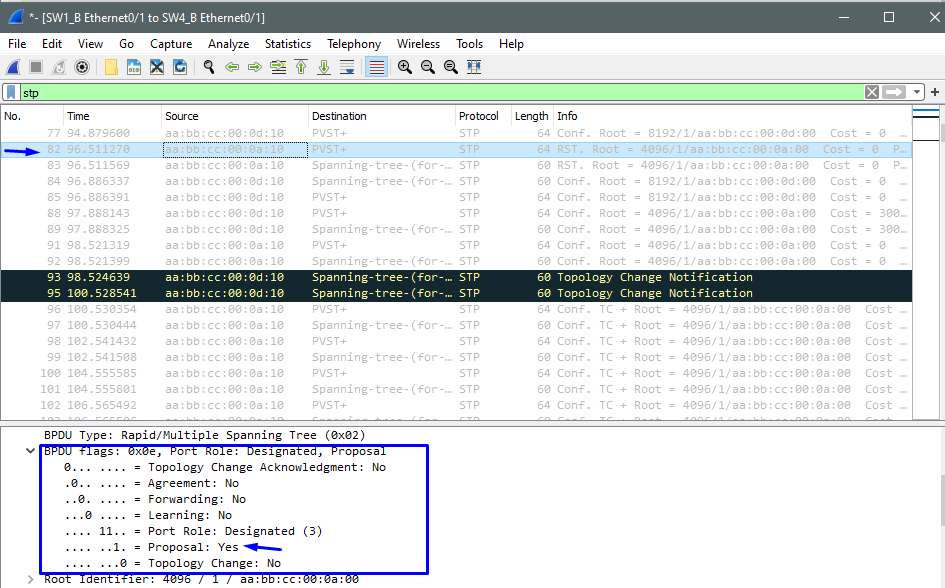
Descrição gerada automaticamente

Vendo o pacote enviado imediatamente a seguir ao TCN no link SW1-4, o mesmo já é enviado novamente pela Root Bridge. Esta BPDU serve para o SW1 avisar quem detetou a mudança que já sabe da alteração na rede. Assim, esta mensagem vem com a flag “topology change acknowledgment” ativa, tal como a flag “topology change”.



Entre o SW4 e o SW3 percebemos que o 4º switch, depois de receber pela Root Bridge a BPDU referida anteriormente, vai enviar também uma BPDU com as mesmas características para o SW3, visto que foi este a detetar a alteração para ele saber que a Root Bridge já foi identificada.

Podemos ver os pacotes proposal (falados anteriormente) no tráfego entre o SW1-SW4, antes da mensagem TCN. É o SW1 que envia uma BPDU ao SW4 com a flag “proposal” ativa. Isto demonstra o processo de proposal/agreement entre estes dois switchs. Neste caso, é o SW1 que está a propor ao SW4 que a DP fique do lado do 1º switch.



Foi realizado um ping do PC1 o 3 e, de seguida, foi injetada uma falha na interface e0/0 do SW3, visto ser esta a Root port do switch a que o PC1 se encontra ligado. Como é confirmamos pela imagem seguinte, o protocolo RSTP é tão rápido a responder a uma falha que nem se deteta nenhuma alteração no ping entre os PCs.

Uma imagem com texto

Descrição gerada automaticamente

O SW3 rapidamente muda a sua porta root para porta que estava como backup, com o papel de Alternate. Assim, a e0/1 passa para Root Port e as comunicações continuam.

Uma imagem com texto

Descrição gerada automaticamente

# Conclusão

O MTConnect é uma excelente tecnologia fornecedora de um largo leque de dados às fábricas e empresas para que as mesmas consigam gerir os seus negócios de uma forma eficaz, prática e consciente daquilo que possuem e produzem.

Praticamente todos os equipamentos (desde os mais antigos até aos mais modernos) são capazes de utilizar este protocolo, conforme explicado em cima. Sendo assim, as empresas conseguem começar a recolher informações valiosas sem a necessidade de um grande investimento em novos equipamentos, nem em formação para os mesmos. Apenas é necessário utilizar uma aplicação para que os dados resultantes sejam trabalhados e com o uso de tecnologias conhecidas mundialmente como a XML, o protocolo pode ainda ser evoluído pelas próprias fábricas, segundo as suas necessidades.

Não é necessário um grande investimento para a instalação deste protocolo numa fábrica, visto que o código de software está disponível para quem necessitar e mesmo as aplicações podem ser obtidas a partir dos fornecedores de equipamentos já compatíveis com o MTConnect e também já existem disponíveis gratuitamente.

A norma está ainda em desenvolvimento de forma que mais equipamentos e áreas consigam usufruir da mesma. Contudo, atualmente os proprietários empresariais já estão conscientes de que esta deve ser um requisito aquando da compra de novos equipamentos ou da criação de um negócio.