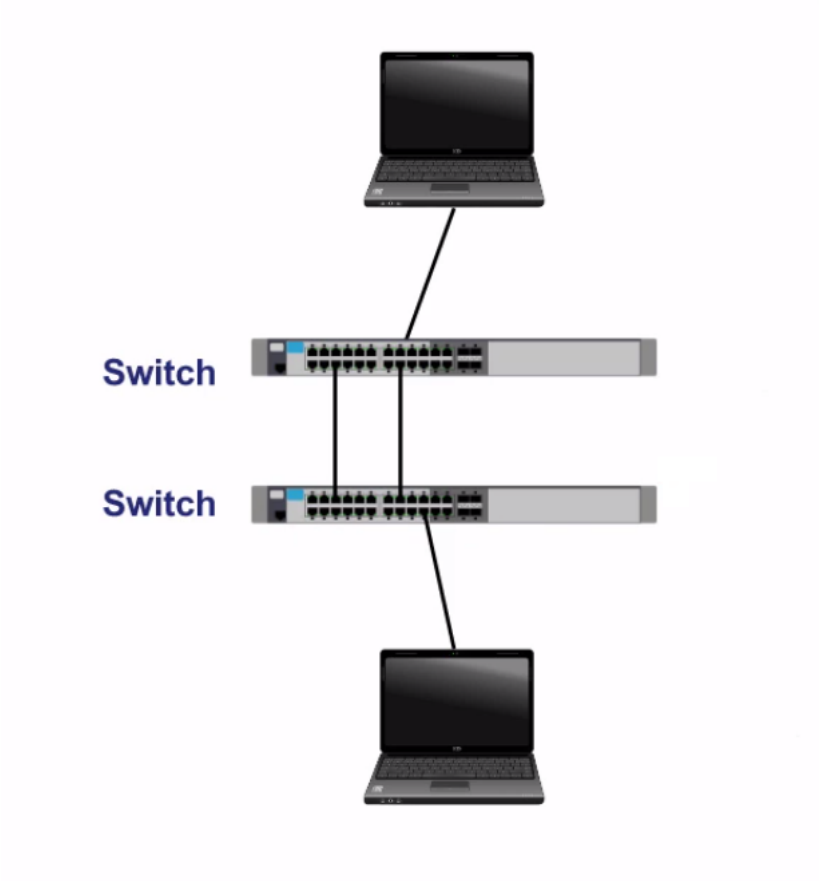
03

Importância do STP

Transcrição

Na etapa anterior conseguimos configurar o terceiro *switch*, que proporcionou uma redundância em nossa rede. Ou seja, se um dos outros links apresentar algum problema, isto não afetará os usuários, pois eles continuarão conseguindo acessar os recursos da rede como faziam até então.

Verificamos, também, que uma das portas dos *switches* ficou na cor laranja, indicando indisponibilidade para recebimento e envio de dados. Todas as outras estão em verde. Usarei uma apresentação para entendermos melhor o que houve:



O computador se conecta a um *switch* que, por sua vez, se conecta a outro, tendo um "link de *backup*" e outro principal. Este segundo *switch* liga-se a outro computador. Quando os dois computadores querem se comunicar, um deles verificará o registro armazenado em relação ao endereço MAC (ou "*MAC adress*") do outro. Caso não haja nenhum registro, o computador lançará o protocolo ARP, ou protocolo *broadcast*.

Assim, quando o segundo computador receber este protocolo, vai devolver o endereço MAC para quem estiver solicitando, o qual será entregue ao primeiro computador, tendo aí um registro guardado para uma possível futura comunicação. Quando o protocolo *broadcast* é enviado ao *switch*, ele será aberto, verificando-se que a informação de destino consta o endereço MAC ("FF-FF-FF-FF-FF-FF"), visto anteriormente nas análises do ***Package Tracer***.

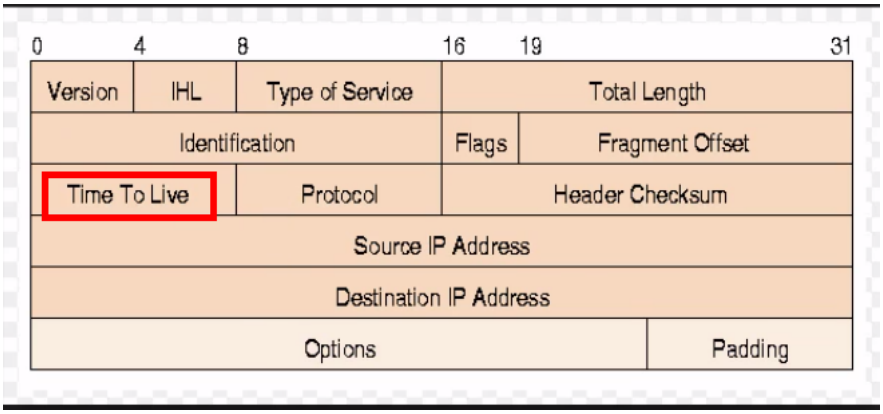
Após verificação deste endereço de destino por parte do primeiro *switch*, ele verá que é preciso repassá-lo para todas as portas conectadas a ele, com exceção daquela por onde veio a informação. Desta forma, o pacote é transferido através do link principal, e também pelo link de *backup*.

Tendo recebido as informações, o segundo *switch* as abrirá, conferindo que o destino é "FF-FF-FF-FF-FF-FF", tendo que repassá-las a todas as portas, excluindo-se a porta pela qual as informações vieram também. Uma destas interfaces, porém, está ligada ao link de *backup*, ou seja, a informação será devolvida ao *switch* principal.

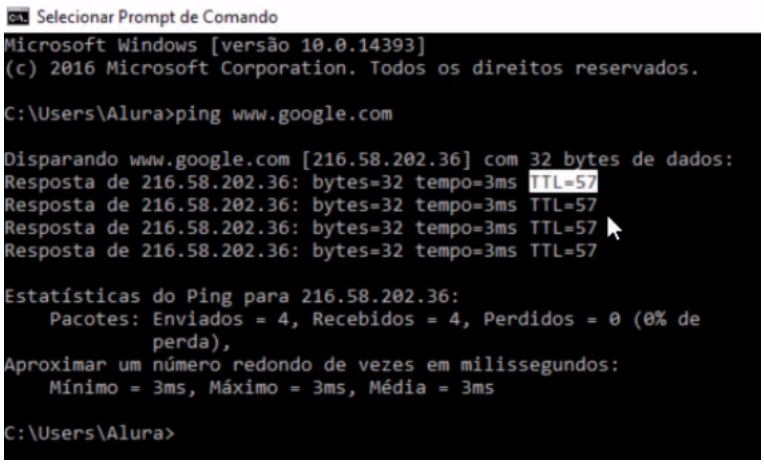
Este link que foi retornado ao *switch* primário será visto como sendo um pacote *broadcast*, então, mais uma vez, entende-se que precisa ser repassado a todas as interfaces, sendo devolvido ao link principal, o qual repassará ao segundo *switch*, que por sua vez verá que se trata de um link *backup*, retornando-o.

Estamos criando um *loop* na rede! Por quê eles acontecem? Vamos tentar verificar a estrutura deste pacote IP, pesquisando por imagens de "*IP packet*" na internet.

Dentro do pacote IP, tem-se o campo "*Time To Live*" (ou **TTL**), que tínhamos visto no curso anterior de Redes (https://cursos.alura.com.br/course/redes-introducao).



Para relembrarmos, abriremos o Prompt de Comando buscando por "cmd" no computador e digitando `ping www.google.com`. Estamos verificando a conectividade do servidor do Google, e obtivemos a informação "`TTL = 57`". Significa que este pacote conseguirá passar por no máximo `57` máquinas intermediárias até alcançar o servidor do Google.



Se em `57` máquinas isto não for possível, o pacote será extinguido, justamente para prevenção destes *loops* infinitos. O TTL é, assim, uma forma de garantir que o pacote deixe de existir em algum momento.

No entanto, este pacote IP é interpretado apenas pelos roteadores, e não pelos *switches*, que não conseguem decrementar este valor ou processar este pacote IP. Por isto, ele fica criando este *loop* constantemente.

Portanto, temos este problema: é preciso ter redundância nos links para garantir a segurança caso algum link fique indisponível, e sabemos que teremos um *backup*, porém, se tivermos um link de *backup*, formamos um *loop* na rede. Como resolveremos este problema?

Derrubando uma árvore. Como assim? Isto é possibilitado por um protocolo chamado ***Spanning Tree Protocol*** (STP), que "derruba uma árvore" em um dos links, por exemplo, o de *backup*, o qual se torna momentaneamente indisponível. Desta forma, as informações só serão passadas pelo link principal. Caso haja algum problema nele, o STP removerá esta árvore, fazendo com que o link *backup* fique disponível novamente, atuando no lugar do primário.

Através deste protocolo, conseguimos garantir a redundância na rede, e também prevenir que estes *loops* aconteçam desnecessariamente. Mas como o *Spanning Tree Protocol* faz para definir quais portas derrubar e quais manter ativas? Veremos na sequência.