Aspectos de Segurança na Camada de Enlace

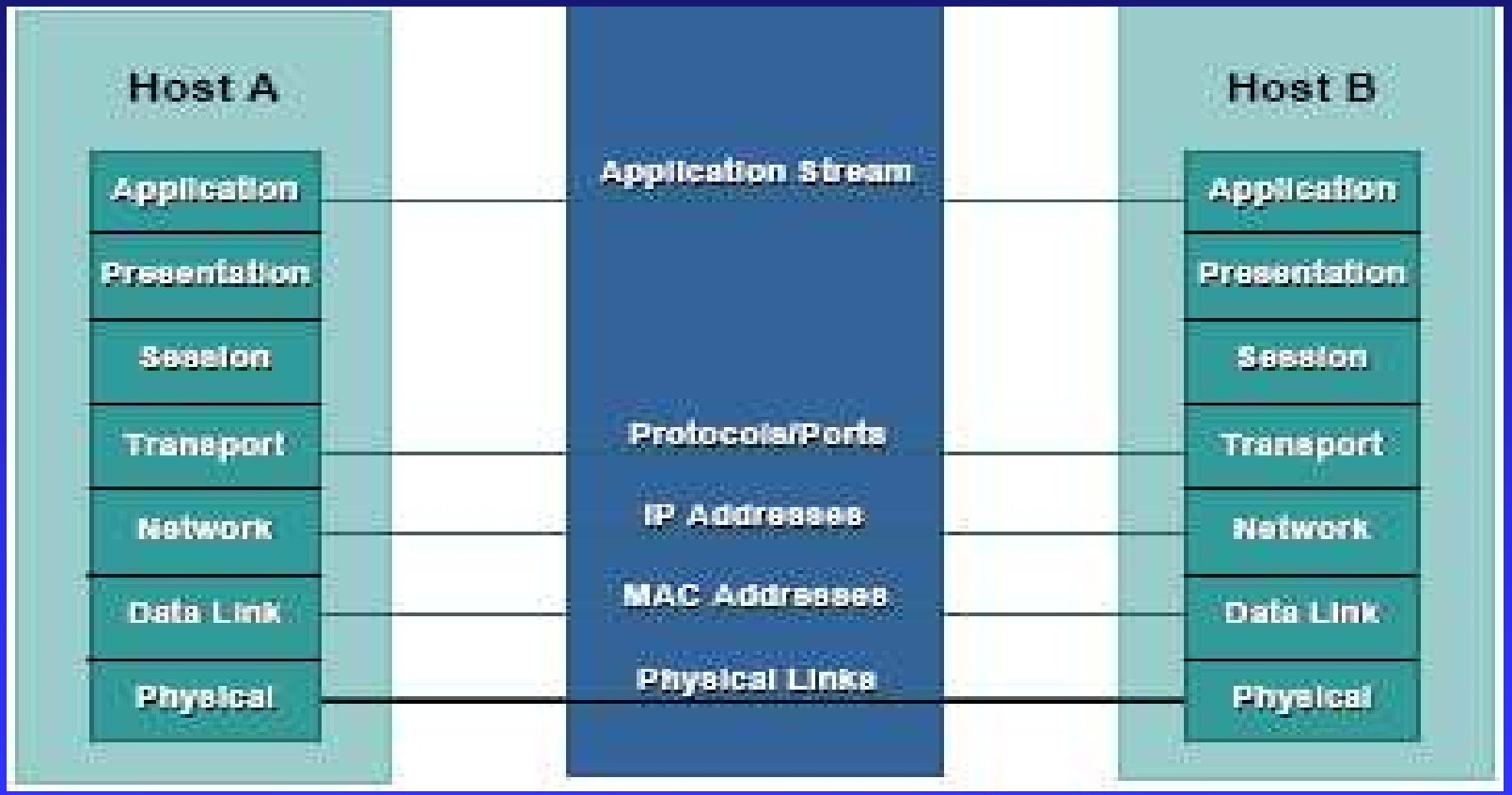
João J. C. Gondim
gondim@cic.unb.br

Dez/2013

Agenda

- Motivação
- Ataques e contramedidas
  - MAC Flooding
  - ARP Spoofing
  - VLAN Hopping
  - STP Tampering
- Comentários finais

A Camada de Enlace



Segurança da Camada 2

- L2 consolida a infraestrutura de redes locais
- Normalmente é tratada no escopo da segurança física: cabeamento e equipamentos
- Trazida ao foco da discussão pelas tecnologias wireless
- Não tem um modelo de segurança...

Mas, quem cuida da L2 ?

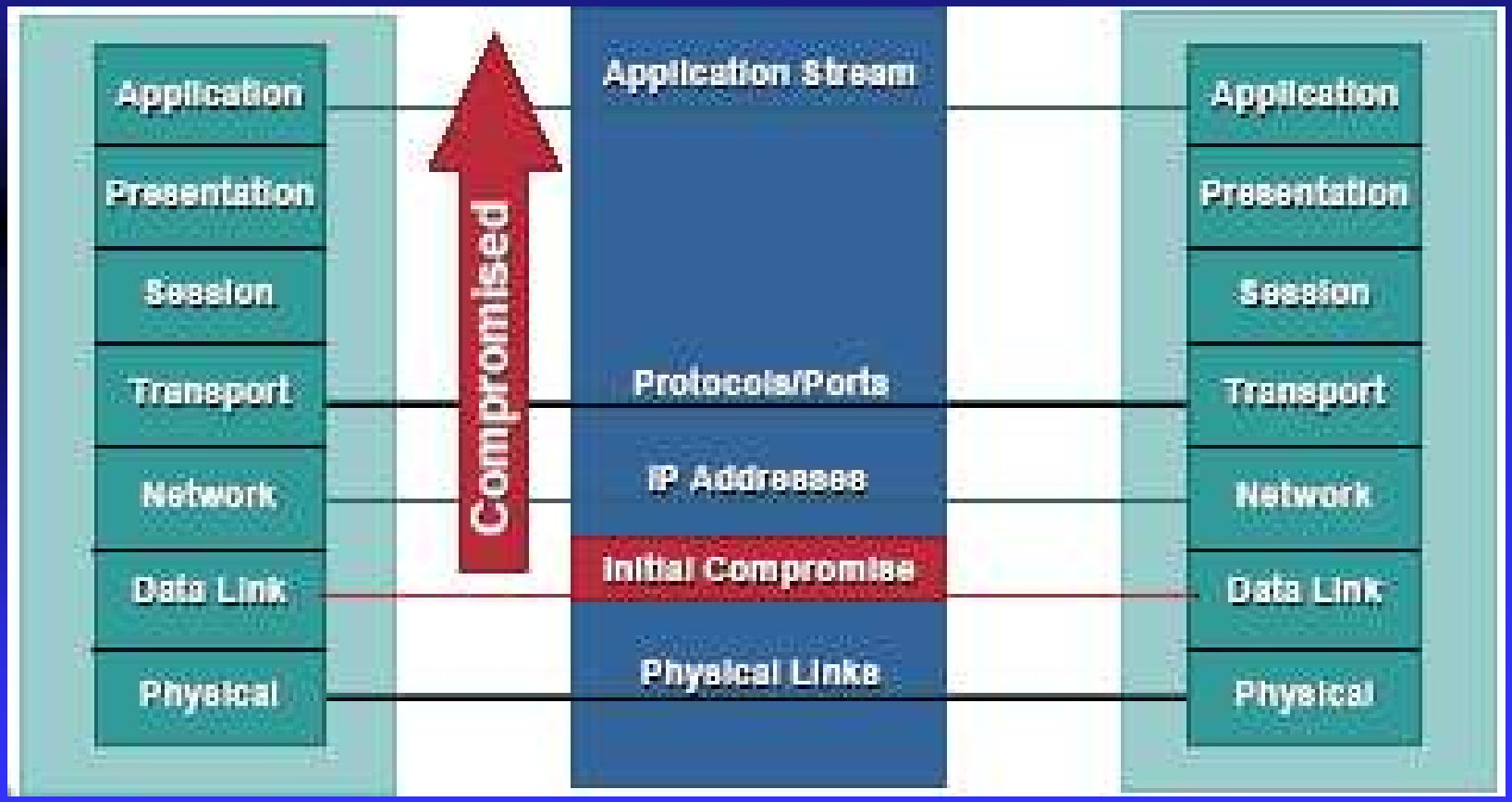
- O gerente da rede
  - Desconhece as questões de segurança
  - Usa/opera os equipamentos e tecnologias
  - Segue o que o gerente de segurança pede
- O gerente de segurança da rede
  - Trata de segurança da L3 para cima
  - Desconhece as questões de segurança e os equipamentos e tecnologias da L2
  - Acredita que o gerente da rede sabe o que está fazendo

Para entender a L2...

- Requer conhecimento sobre:
  - Os protocolos
  - As tecnologias
  - Os equipamentos
  - A operação
- O profissional típico de computação “sabe” desenvolver /manter sistemas
- O profissional típico de redes “sabe” configurar os equipamentos e se preocupa com disponibilidade e performance

Por que se preocupar com a L2 ?

- O Efeito Dominó



Disclaimer

- Contexto de redes ethernet comutadas
- Não seremos exaustivos
- Demonstrações realizadas em ambiente controlado para fins puramente didáticos e com consentimento institucional
- No Gedanken-only approach
- No karate kid approach
- Não tentem isso em casa

Ataques na L2

- Incidentes internos
- Altamente intecionais
- Se aproveitam de relações de confiança implícitas entre os diversos componentes da infraestrutura
- Ataques exploram tanto características de equipamentos como de protocolos

MAC Flooding

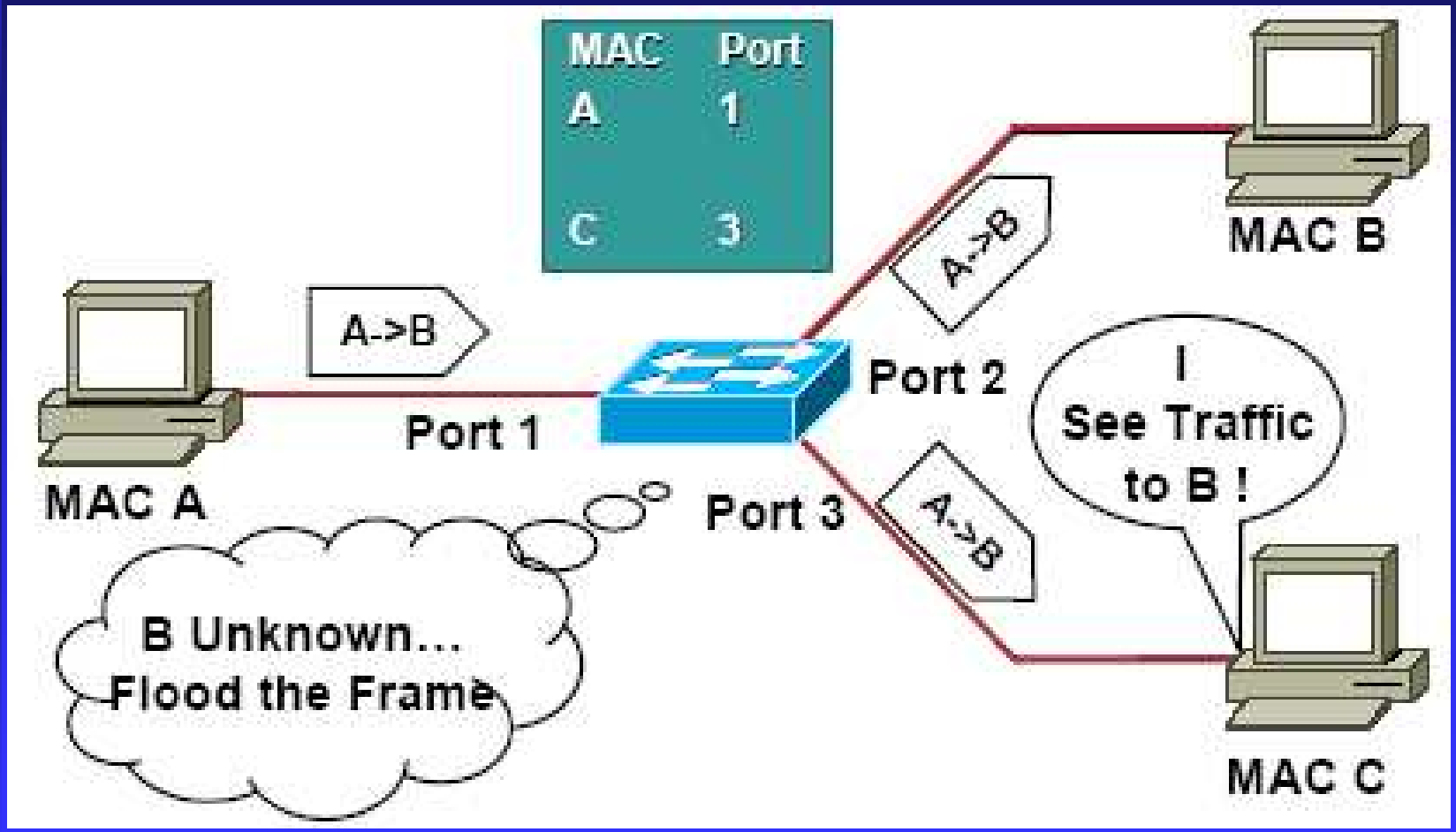
- Consiste em saturar a capacidade de decisão de encaminhamento do switch
- Por ser fail open, para evitar DoS, o encaminhamento é feito para todas as portas
- Efeito obtido:

"o switch funciona como um hub"

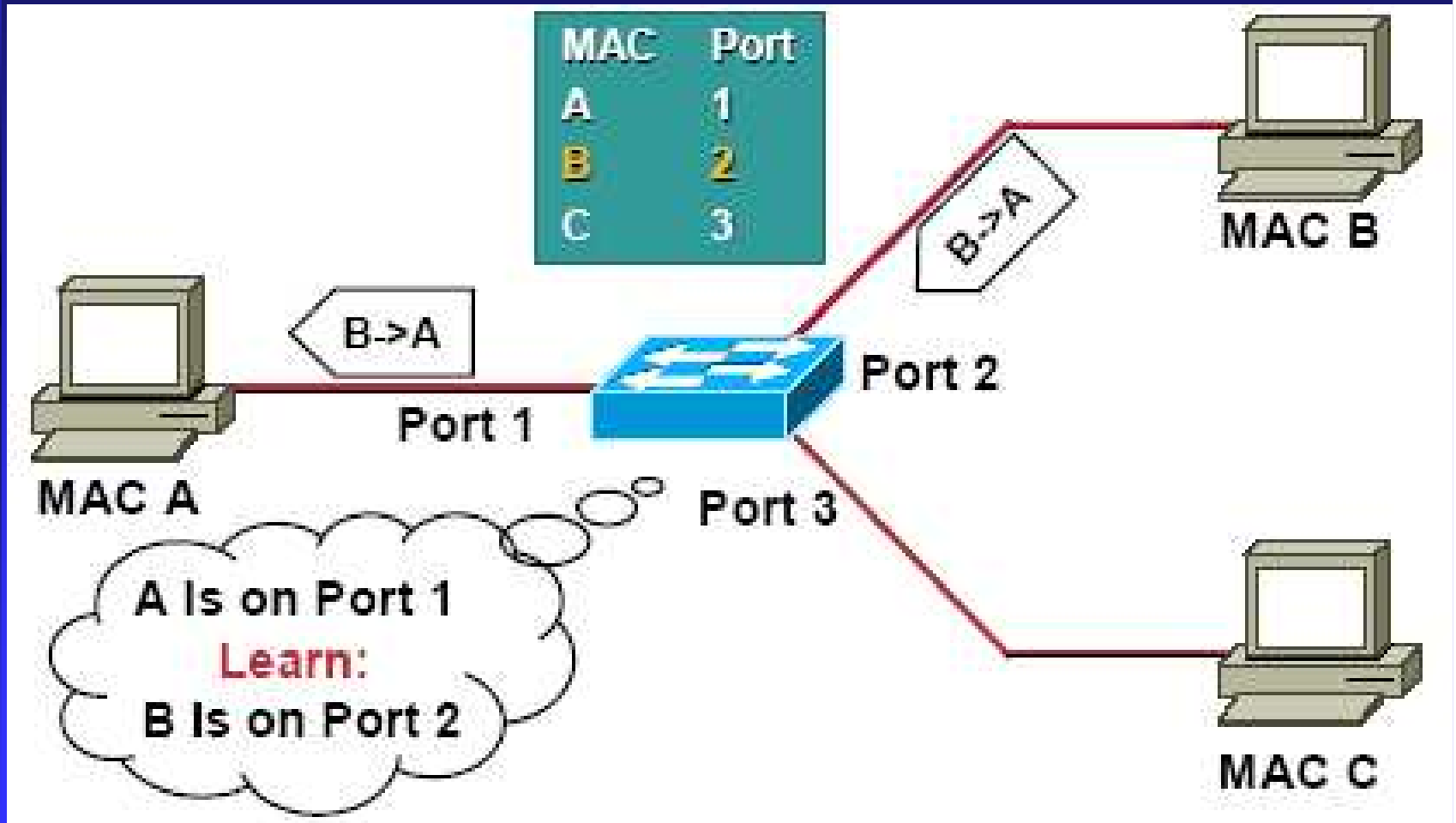
MAC Flooding

- O switch monta dinnamicamente uma tabela com associando portas com endereço MAC
- Para cada frame que entra, uma linha da MAC Address/CAM table é acrescentada, ou se já presente tem seu timer reinicializado (30 segundos – típico)
- Fica armazenado um hash de 17 bits gerado a partir de 63 bits (SRC MAC, VLAN, etc)

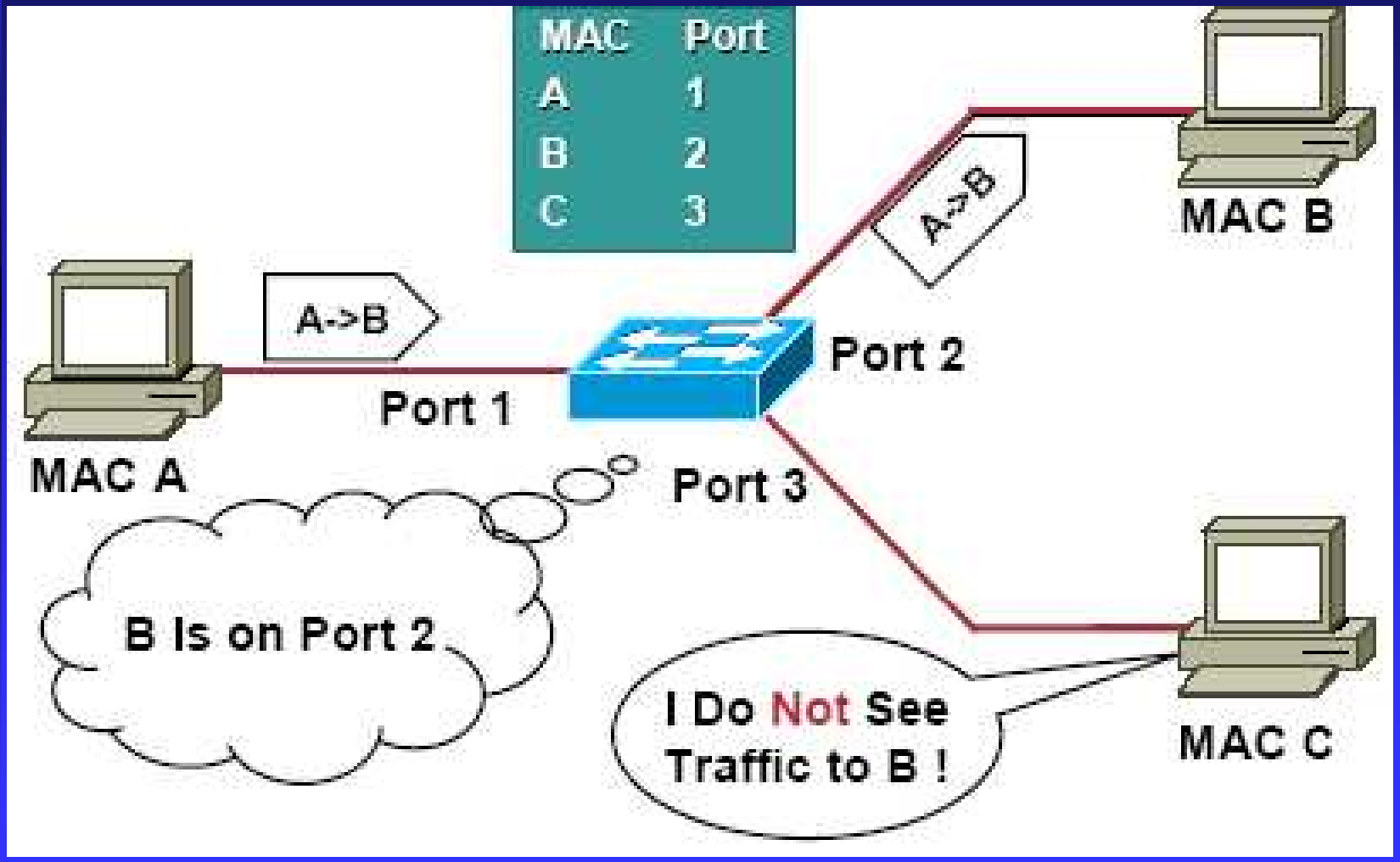
CAM normal


MAC Port
A 1
C 3
A->B
MAC B
A->B
Port 2
I
See Traffic
to B !
A->B
Port 1
MAC A
Port 3
B Unknown...
Flood the Frame
MAC C

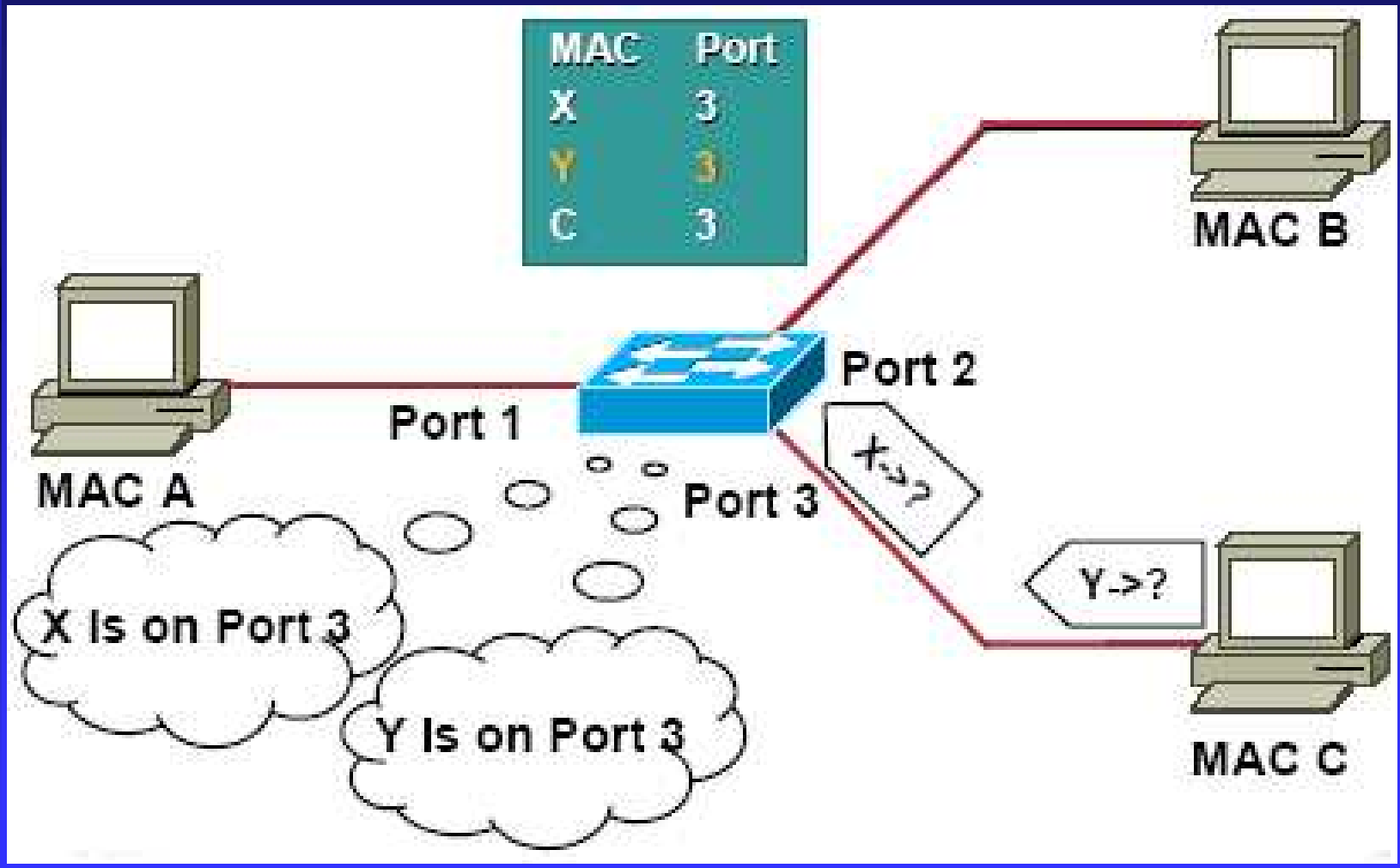
CAM normal


MAC Port
A 1
B 2
C 3
B->A
MAC B
B->A
Port 1
Port 2
MAC A
Port 3
A Is on Port 1
Learn:
B Is on Port 2
MAC C

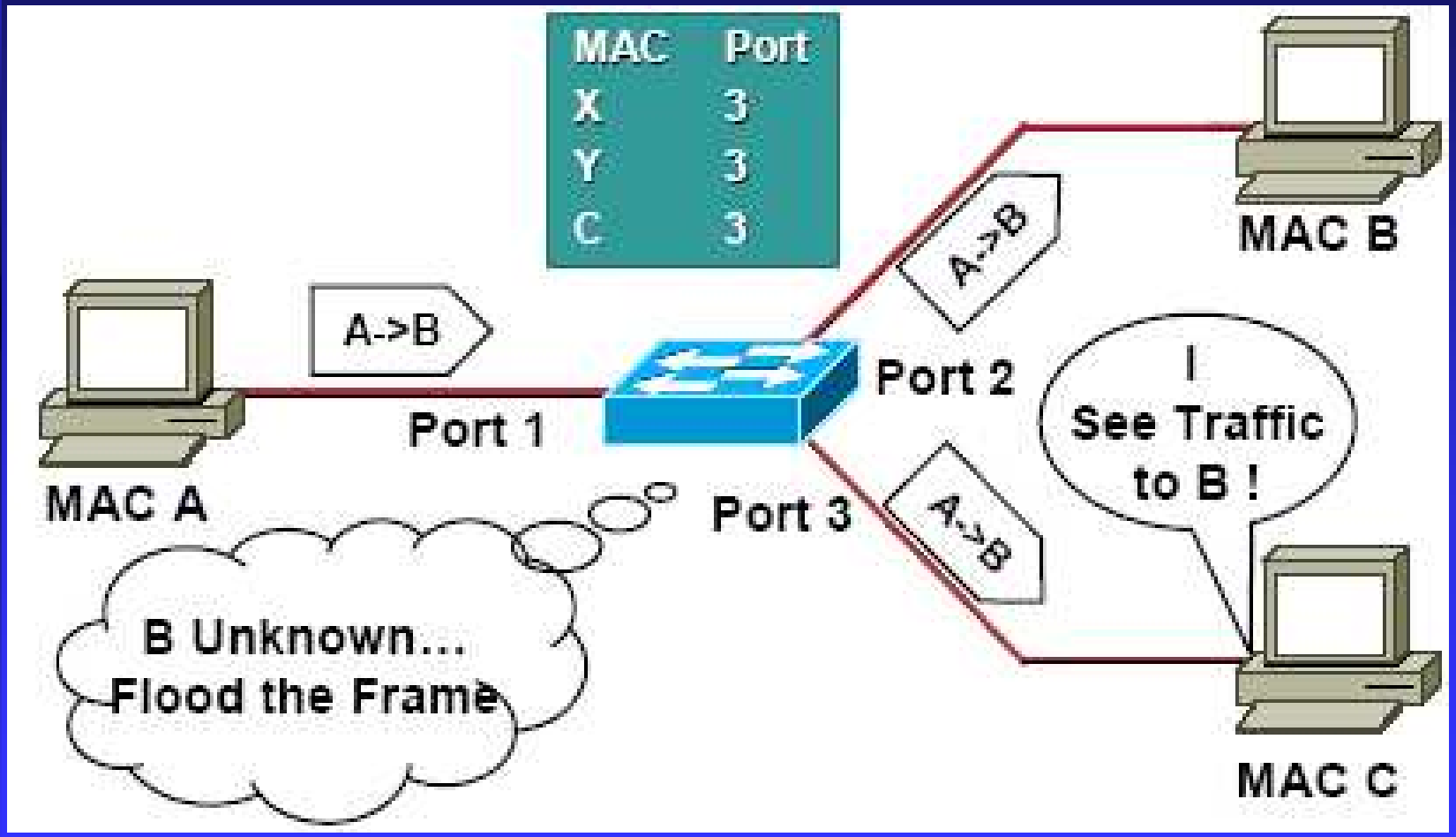
CAM normal 3/3



CAM overflow 1/2



CAM overflow 2/2



MAC Flooding

```
[root@attack-lnx dsniff-2.3]# ./macof
b5:ef:65:4b:d5:59 2c:01:12:7d:bd:36 0.0.0.0.4707 > 0.0.0.0.28005: S 106321318:106321318(0) win 512
68:2a:55:6c:1c:1c bb:33:bb:4d:c2:db 0.0.0.0.44367 > 0.0.0.0.60982: S 480589777:480589777(0) win 512
1e:95:26:5e:ab:4f d7:80:6f:2e:aa:89 0.0.0.0.42809 > 0.0.0.0.39934: S 1814866876:1814866876(0) win 512
51:b5:4a:7a:03:b3 70:a9:c3:24:db:2d 0.0.0.0.41274 > 0.0.0.0.31780: S 527694740:527694740(0) win 512
51:75:2e:22:c6:31 91:a1:c1:77:f6:18 0.0.0.0.36396 > 0.0.0.0.15064: S 1297621419:1297621419(0) win 512
7b:fc:69:5b:47:e2 e7:65:66:4c:2b:87 0.0.0.0.45053 > 0.0.0.0.4908: S 976491935:976491935(0) win 512
19:14:72:73:6f:ff 8d:ba:5c:40:be:d5 0.0.0.0.867 > 0.0.0.0.20101: S 287657898:287657898(0) win 512
63:e8:58:03:4e:f8 82:b6:ae:19:0f:e5 0.0.0.0.58843 > 0.0.0.0.40817: S 1693135783:1693135783(0) win 512
33:d7:e0:2a:77:70 48:96:df:20:61:b4 0.0.0.0.26678 > 0.0.0.0.42913: S 1128100617:1128100617(0) win 512
f2:7f:96:6f:d1:bd c6:15:b3:21:72:6a 0.0.0.0.53021 > 0.0.0.0.5876: S 570265931:570265931(0) win 512
22:6a:3c:4b:05:7f 1a:78:22:30:90:85 0.0.0.0.58185 > 0.0.0.0.51696: S 1813802199:1813802199(0) win 512
f6:60:da:3d:07:5b 3d:db:16:11:f9:55 0.0.0.0.63763 > 0.0.0.0.63390: S 1108461959:1108461959(0) win 512
bc:fd:c0:17:52:95 8d:c1:76:0d:8f:b5 0.0.0.0.55865 > 0.0.0.0.20361: S 309609994:309609994(0) win 512
bb:c9:48:4c:06:2e 37:12:e8:19:93:4e 0.0.0.0.1618 > 0.0.0.0.9653: S 1580205491:1580205491(0) win 512
e6:23:b5:47:46:e7 78:11:e3:72:05:44 0.0.0.0.18351 > 0.0.0.0.3189: S 217057268:217057268(0) win 512
c9:89:97:4b:62:2a c3:4a:a8:48:64:a4 0.0.0.0.23021 > 0.0.0.0.14891: S 1200820794:1200820794(0) win 512
56:30:ac:0b:d0:ef 1a:11:57:4f:22:68 0.0.0.0.61942 > 0.0.0.0.17591: S 1535090777:1535090777(0) win 512
```

Contra-medidas

- Port secure / MACbased filtering
- Limitar a quantidade de endereços que uma porta pode aprender
- Especificar os endereços que uma porta pode aprender
- Administrativamente pode ser um pesadelo
- Engessa a infra....

ARP Spoofing

- Conhecido desde 95-96 na comunidade
- ferramentas pelo menos desde 97
  - ARPTOOL
  - ARP0.c
  - ARPSEND

ARP Spoofing

- Ferramentas não tão antigas assim:
  - `hunt`
  - `denver`
  - `ettercap`
  - `angst`
  - `arp-sk`
  - `parasite...`

Protocolo ARP

- Address Resolution Protocol - RFC 826
  - Protocolo de Resolução de Endereços

- Mapeia Endereços IP ⇔ Endereços MAC
- E os guarda em um cache
  - Totalmente sem autenticação
  - Crédulo: aceita respostas a perguntas que ele não fez

Protocolo ARP

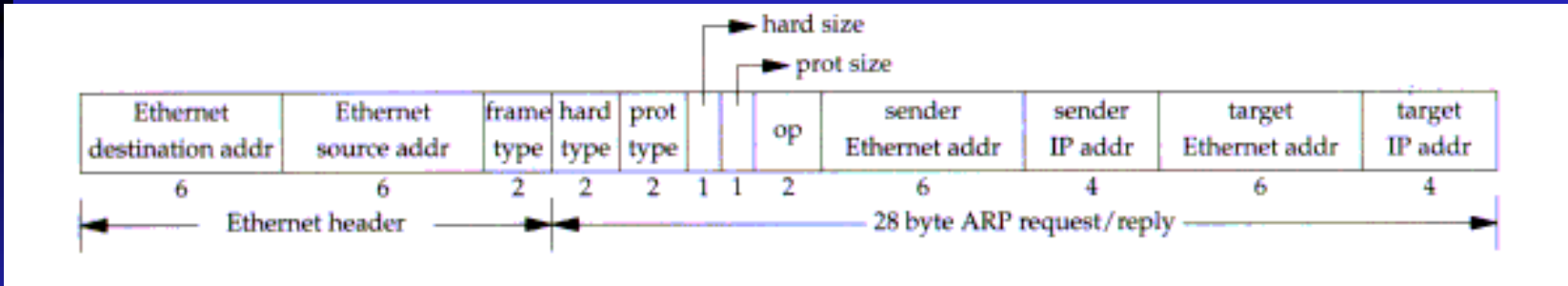
- Cache ARP
  - Mantém na memória do computador durante algum tempo (da ordem de minutos) as associações entre endereços IPs e MACs
  - Registra qualquer coisa que lhe seja mandada, inclusive o que não perguntou.

Protocolo ARP

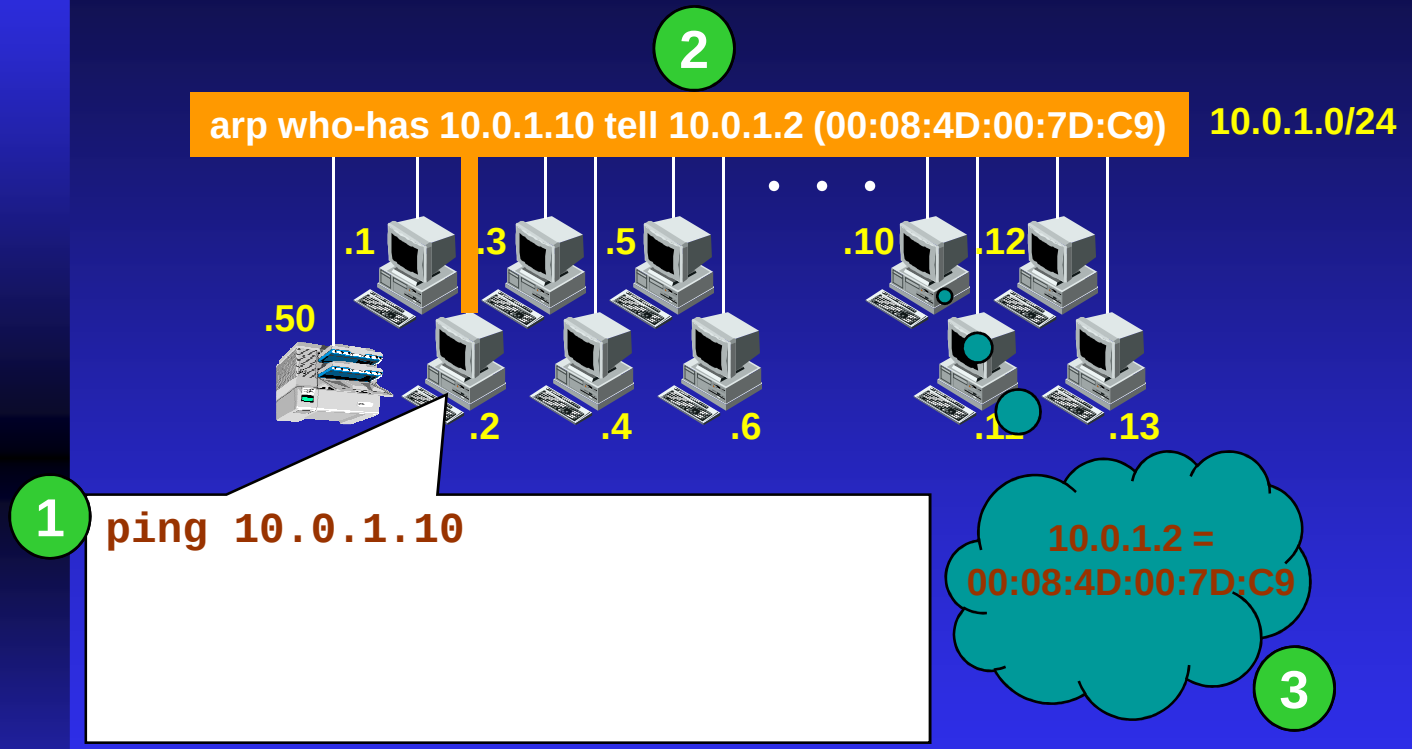
- Dois tipos de mensagem:
  - Request (“who-has”)
    - normalmente broadcast
  - Reply (“is-at”)
    - normalmente unicast

Protocolo ARP

- Formato do frame:

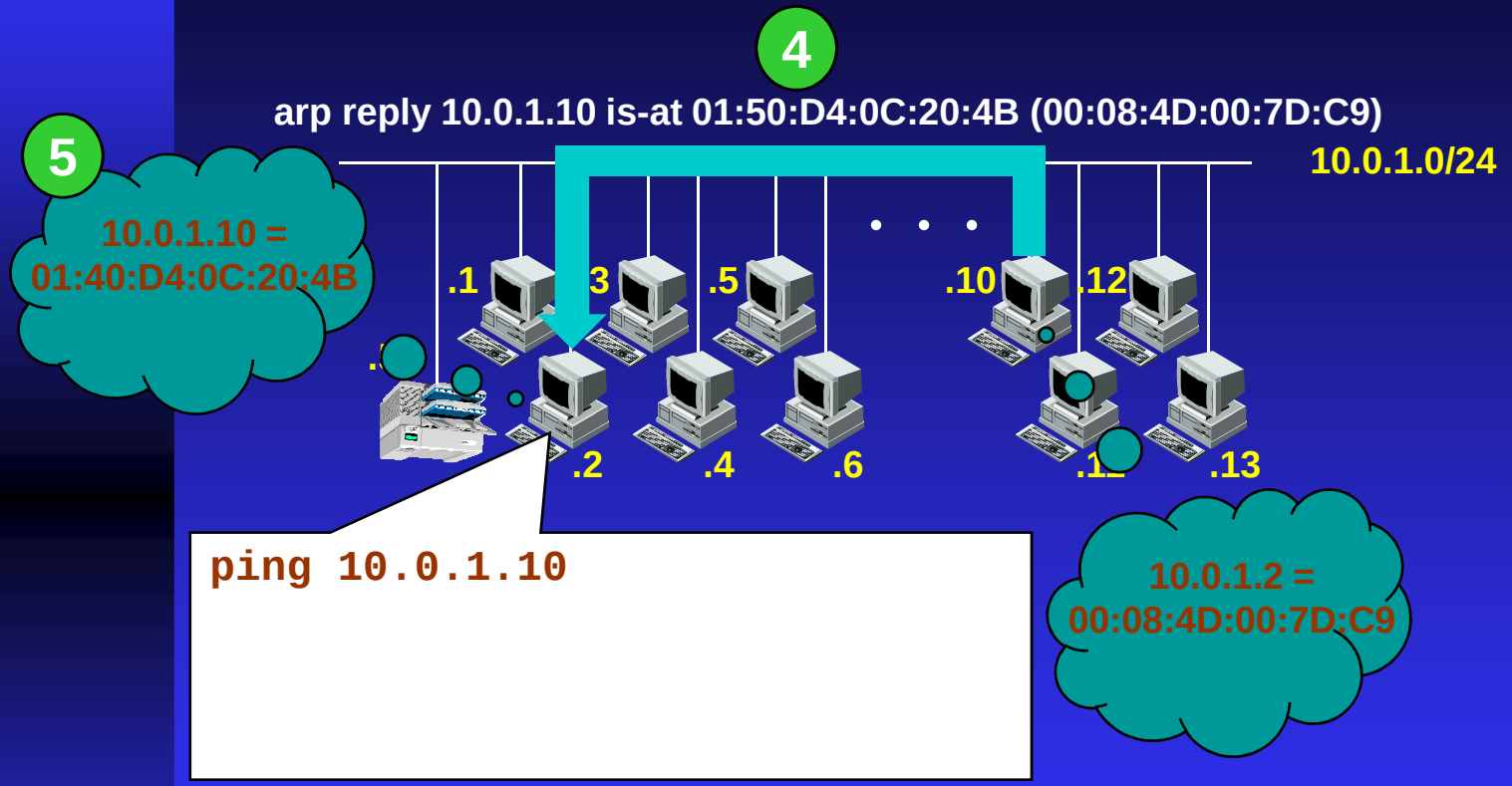


Protocolo ARP: exemplo



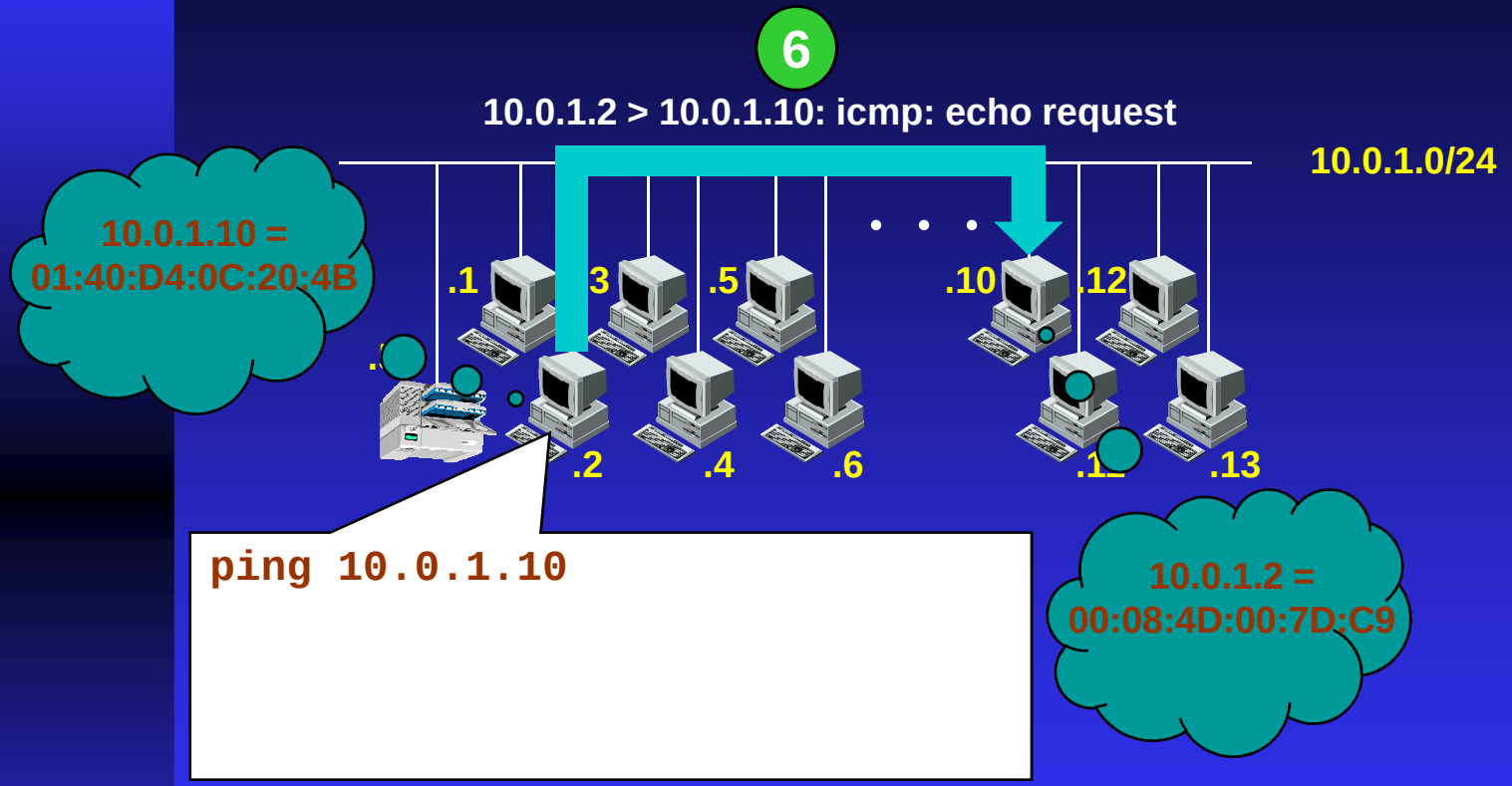
1. O usuário manda um “ping” para uma outra máquina
2. Como o computador desconhece o endereço MAC da máquina destino, ele manda um broadcast arp perguntando por ele.
3. A máquina-destino recebe a pergunta, e aproveita para registrar o endereço MAC da máquina que perguntou no seu cache ARP.

Protocolo ARP: exemplo



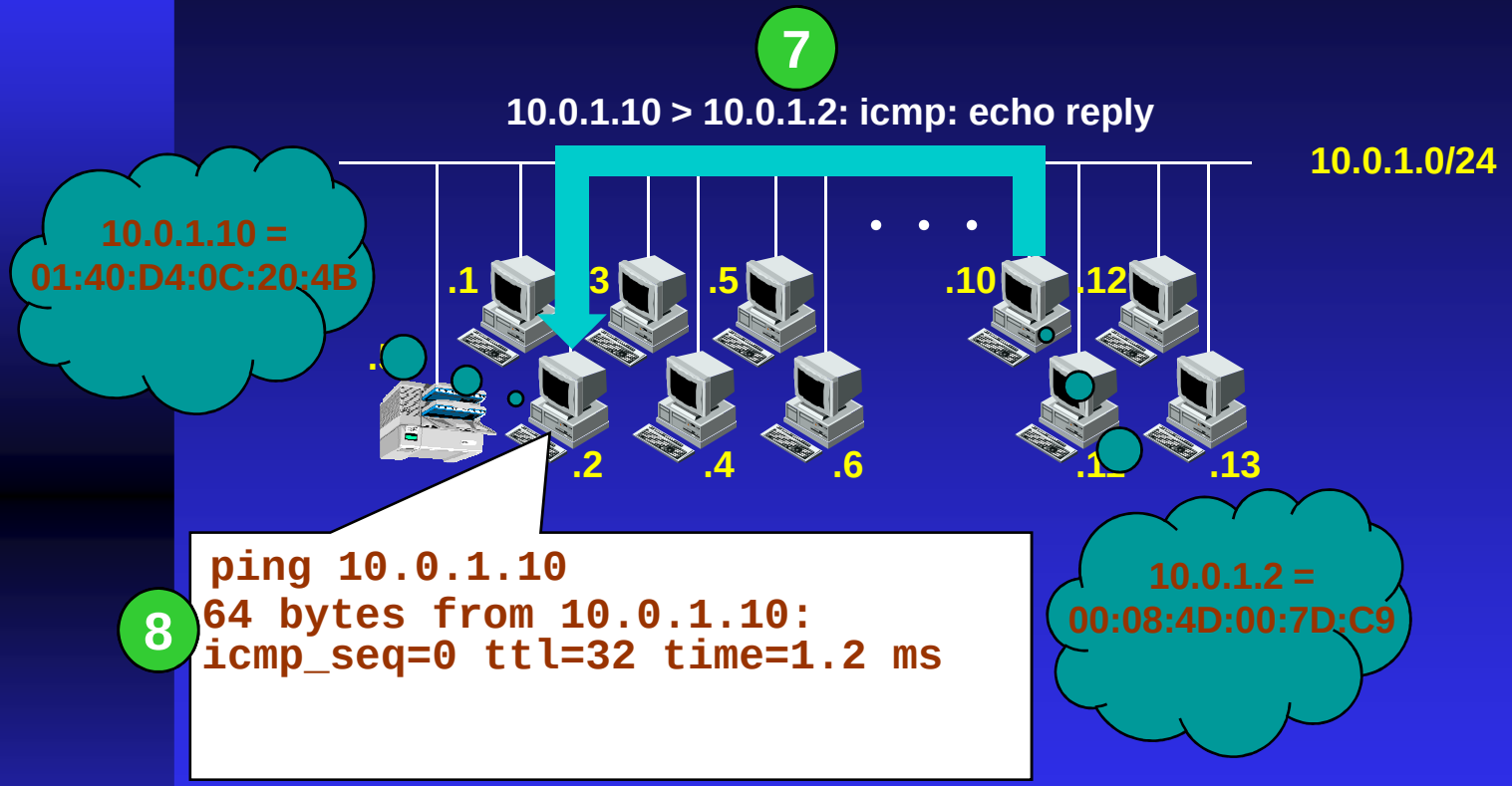
4. A máquina-destino responde para o computador de origem, mencionando seu endereço MAC.
5. O computador de origem recebe a resposta e registra o endereço no seu cache ARP.

Protocolo ARP: exemplo



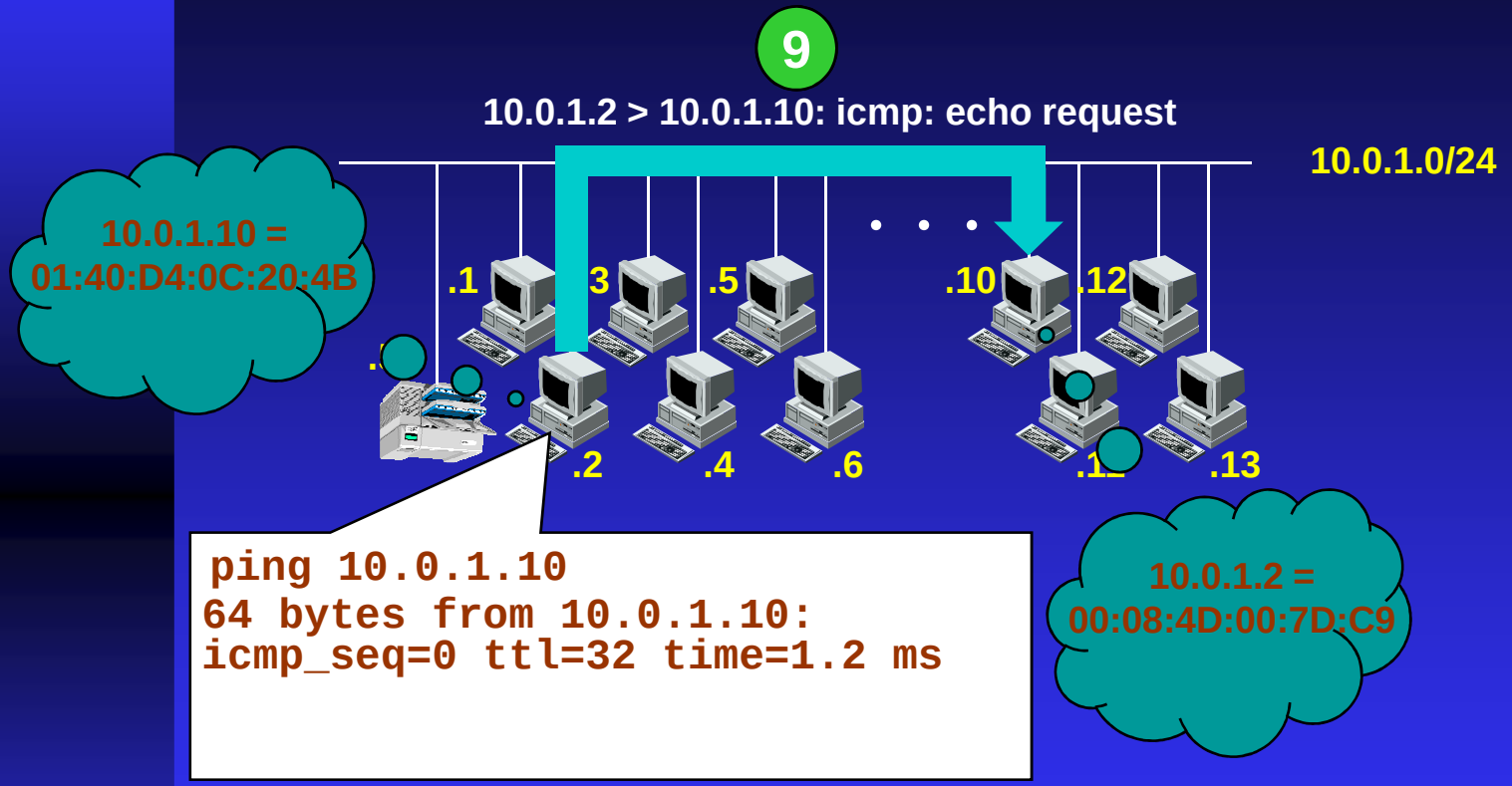
6. De posse do endereço MAC da máquina destino, o computador de origem finalmente já pode enviar o ping (pedido de eco ICMP) para ela.

Protocolo ARP: exemplo



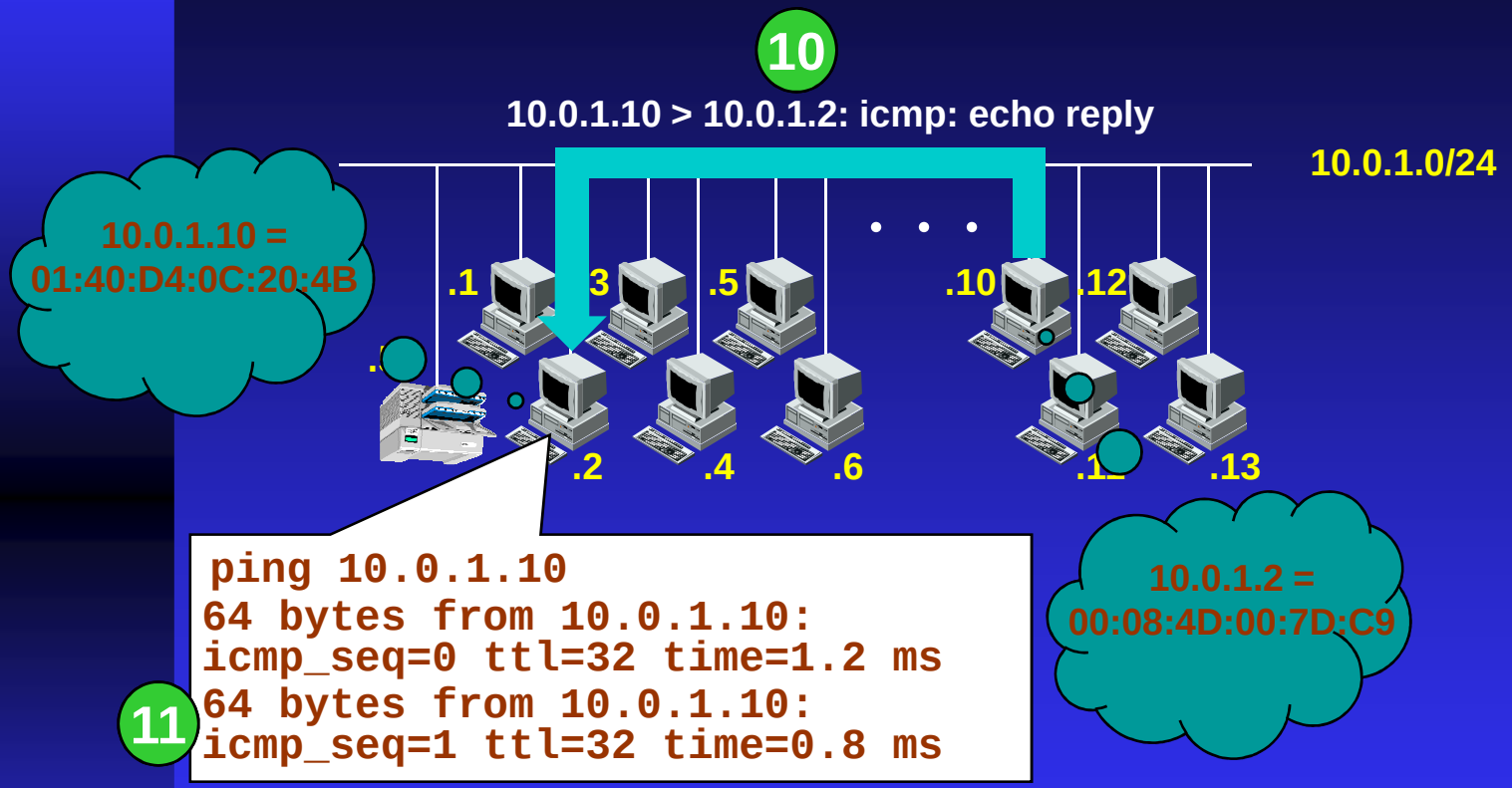
7. A máquina destino responde ao “ping”, enviando uma resposta a eco ICMP de volta ao computador de origem.
8. Ao receber a resposta, o sistema operacional do computador de origem a repassa para o aplicativo, que notifica o usuário.

Protocolo ARP: exemplo



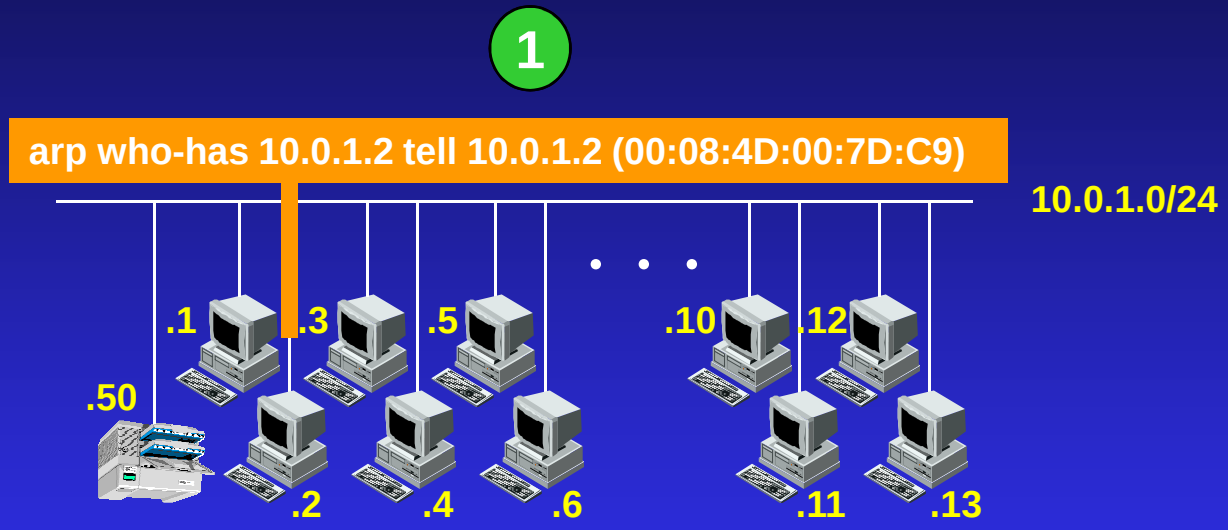
9. O próximo ping já segue direto, pois o endereço MAC da máquina destino já consta no cache ARP

Protocolo ARP: exemplo



10. A máquina destino responde ao segundo “ping”.
11. Novamente, a resposta é recebida e repassada para a aplicação, que notifica o usuário.

ARP Gratuito



1. Durante a inicialização da pilha TCP, a máquina manda um pedido ARP procurando pelo seu próprio IP, a fim de determinar se alguma outra máquina já esta usando aquele IP. Normalmente, nenhuma outra máquina deveria responder.

ARP Estático

- mantém entradas permanentes no cache ARP
- não são (*não deveriam ser*) sobre-escritas por mensagens eventualmente recebidas
- definição local em cada máquina

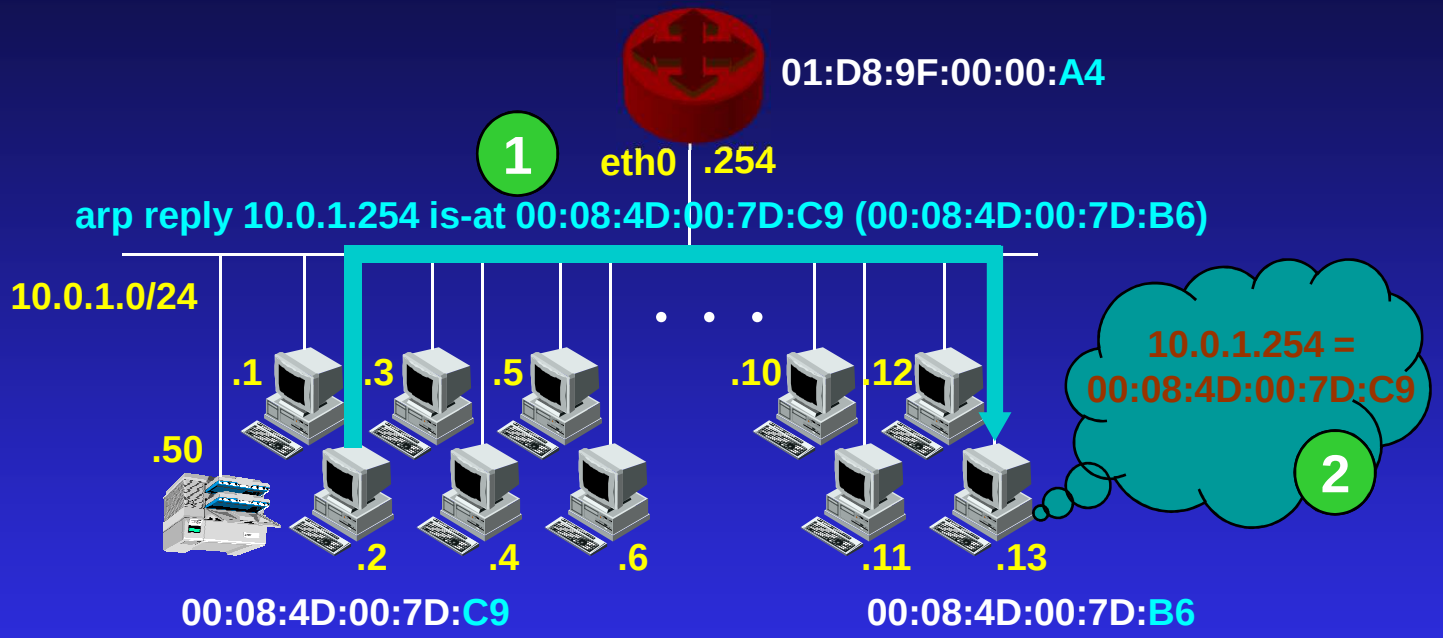
ARP Proxy

- Em algumas situações, uma máquina pode responder ARP requests por outra
- tipicamente, um roteador responderia a um request destinado a uma máquina fora da sub-rede
- o roteador passaria o seu endereço MAC

ARP Spoofing

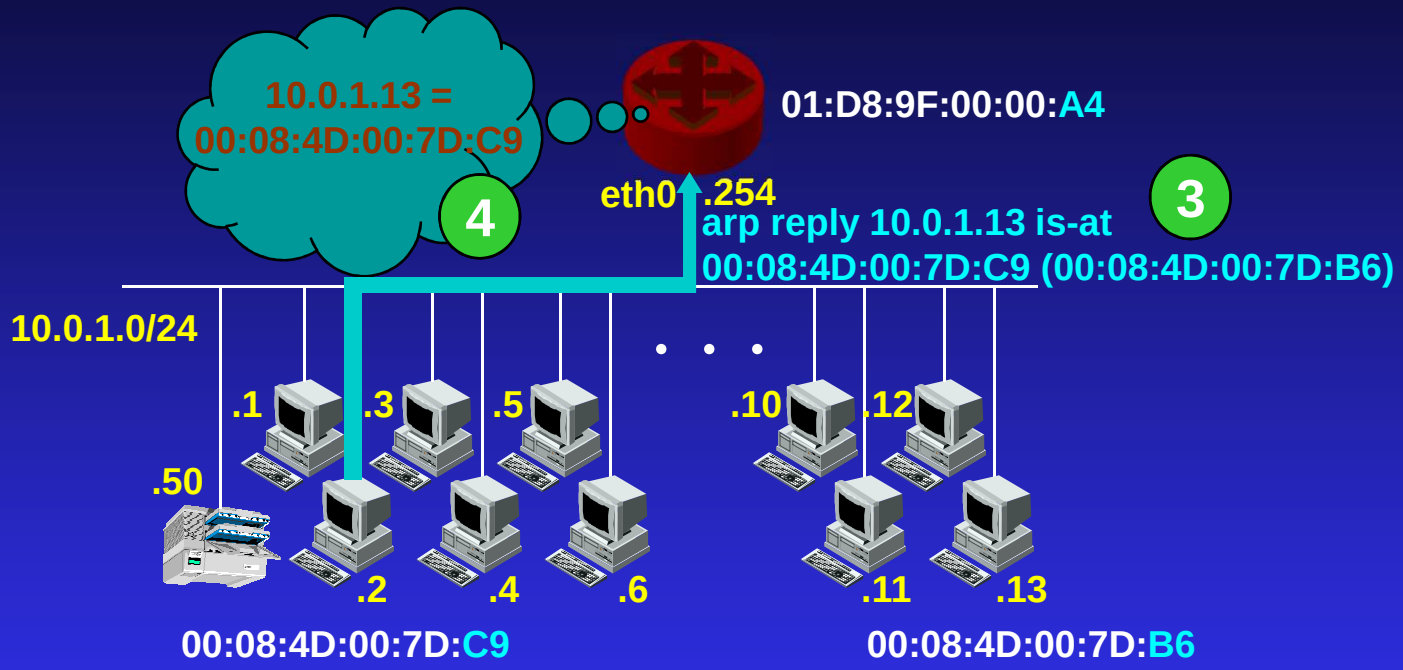
- Adulteração/fabricação de tráfego
- pode ser usado para:
  - negação de serviço
  - man-in-the-middle
    - escuta
  - sequestro de sessão

ARP Spoofing



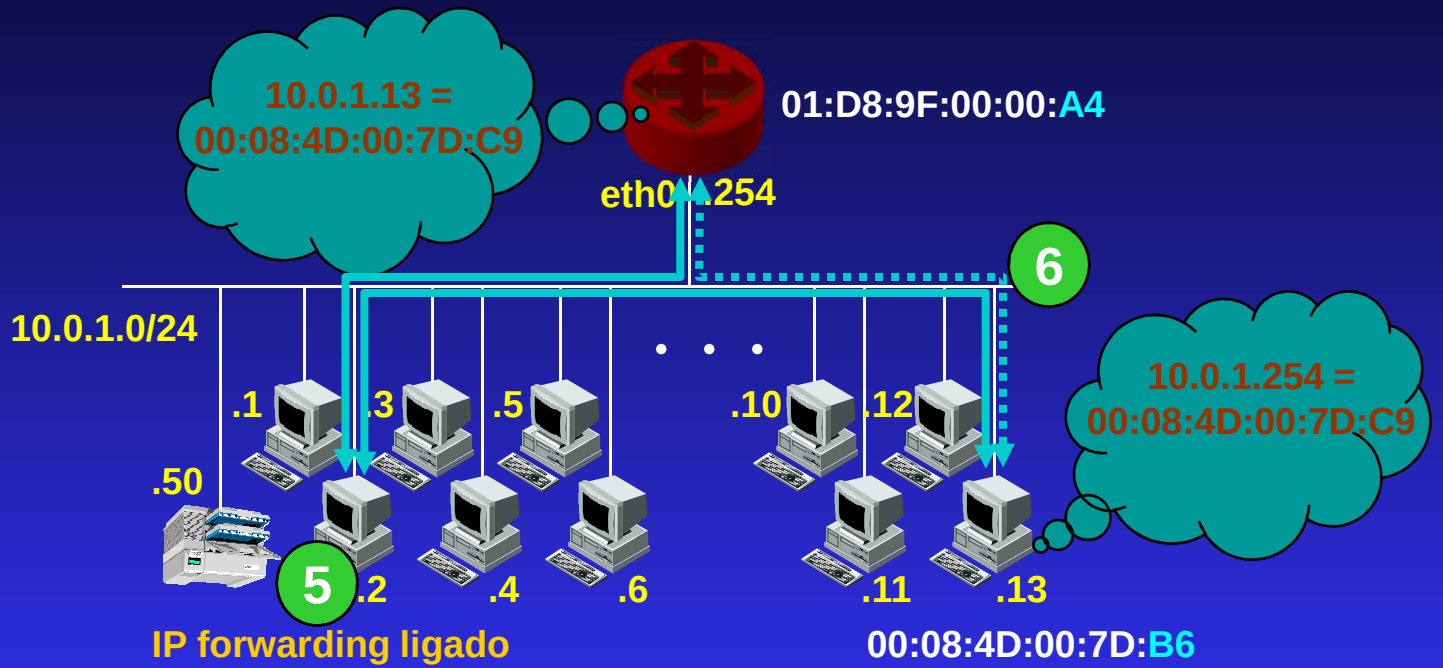
1. A máquina atacante (C9) manda um ARP reply para a vítima (B6), informando que seu IP é 10.0.1.254. Como o protocolo é sem estado, a vítima não tem como saber se de fato perguntou ou não. Já que o pacote veio como sendo para ela, não lhe resta opção senão aceitá-lo.
2. A vítima prontamente atualiza seu cache com o dado falso. Daí em diante, tudo que ela tentar mandar para 10.0.1.254 (o roteador) na realidade irá para 10.0.1.2 (o atacante).

ARP Spoofing



3. A atacante (C9) usa o mesmo truque com o roteador: manda um ARP reply para ele informando que ela o IP 10.0.1.13 (que, na realidade, é da vítima) é dela.
4. O roteador prontamente acredita no pacote, atualizando seu cache ARP. Daí em diante, tudo o que ele tentar enviar para 10.0.1.13 (a vítima), na realidade irá para 10.0.1.2 (o atacante).

ARP Spoofing



5. O atacante liga o IP forwarding de sua máquina e possivelmente toma toma algumas outras medidas, como suprimir ICMP redirects. Isso faz com que todo pacote IP que a vítima mande para ele seja repassado para o roteador e vice-versa.
6. Assim, a vítima e o roteador conversam normalmente, como se estivessem falando diretamente, totalmente sem saber que seu tráfego está passando pelo atacante. Nessa posição privilegiada, o atacante pode gravar o tráfego com um sniffer ou partir para ataques ativos.

ARP Spoofing: componentes

- Poisoning
  - injeção mapeamentos IP/MAC falsos no cache ARP da vítima
  - pode requerer repetição, devido ao timeout do cache
- Relay
  - repasse de tráfego

ARP spoofing: encaminhamento

- L2 bridging
  - intercepta no driver da placa
  - muda o endereço MAC e envia
  - lento -> no hunt a implementação é lenta
- L3 routing
  - desabilitar ipforwading e icmpredirect
  - TTL -> TTL - 1

Contra-medidas tradicionais

- ARP estático em todas as máquinas e todos os endereços da rede
  - inviável administrativamente em uma rede “de verdade”
- Port secure/MAC based filtering
  - limita/define os endereços a serem aprendidos por cada porta
  - efetivo com flooding
  - inócuo para spoofing

Porque é inócuo ?

- ARP spoofing não se baseia na falsificação do endereço
- há ferramentas de ataque que usam o seu próprio MAC real
- ARP spoofing se baseia na falsificação do mapeamento IP -> MAC !!!

Características de implementações: Win 9x/NT/2k

- Atualizam os cache ARP até para entradas estáticas quando recebem um ARP reply.
- Aceitam replies em broadcast
  - poisoning em mais de uma máquina
  - com um único pacote

Características de implementações: Linux

- ARP implementado “quase” stateful
  - não é crédulo
  - só aceita reply para requests que enviou
  - o ataque ainda é possível
    - basta forçar o Linux a “perguntar”...
    - é exatamente o que o `denver` faz
    - manda um ICMP echo req + trem de arp replies

Novas contra-medidas

- ARP Inspection
  - Define em cada porta os mapeamentos ARP esperados
  - Exemplo:

```
set security acl ip ACL_VLAN951 permit
        arpinspection host 132.216.251.129
 00-d0-b7-11-13-14
set security acl ip ACL_VLAN951 deny
  arpinspection host 132.216.251.129 any log
set security acl ip ACL_VLAN951 permit
  arpinspection host 132.216.251.250
  00-d0-00-ea-43-fc
```

Novas contra-medidas

- Dynamic ARP Inspection
  - Monta dinamicamente uma tabela ARP global baseada em tráfego DHCP
- Private VLANs
  - Subdivide a VLAN segundo funçoes
    - Comunidades
    - Portas promíscuas e isoladas

O futuro é promissor...

- IPv6 é imune ao ARP Spoofing
  - Só porque não tem ARP
- NDP – Neighbour Discovery Protocol
  - Projetado e operado sob as mesmas premissas do ARP
  - Só muda o formato dos frames
  - Coming attractions ...
    - NDP Spoofing
    - NDP Spoofing detection

VLANs

- VLANs particionam os domínios de broadcast em switches
- Permitem refletir a estrutura funcional/organizacional na rede
- Dão escalabilidade a redes ethernet comutadas

VLANs

- A priori o tráfego de uma VLAN não passa para outra
- Para passar, seria necessário roteamento
- Daí a motivação para switches de camada 3
- A informação de VLAN é acrescentada ao frame como um tag, que o switch usa para encaminhamento

VLANs

- Os frames para VLANs fora do switch que o recebeu, são passadas pelas portas de trunking
- As portas de trunk normalmente estão na VLAN 1, que é default para gerenciamento e controle
- O encapsulamento dos tags segue IEEE 802.1q

VLANs

- O encapsulamento dos tags segue IEEE 802.1q
- TPID 0x8100
- TCI: vlan id

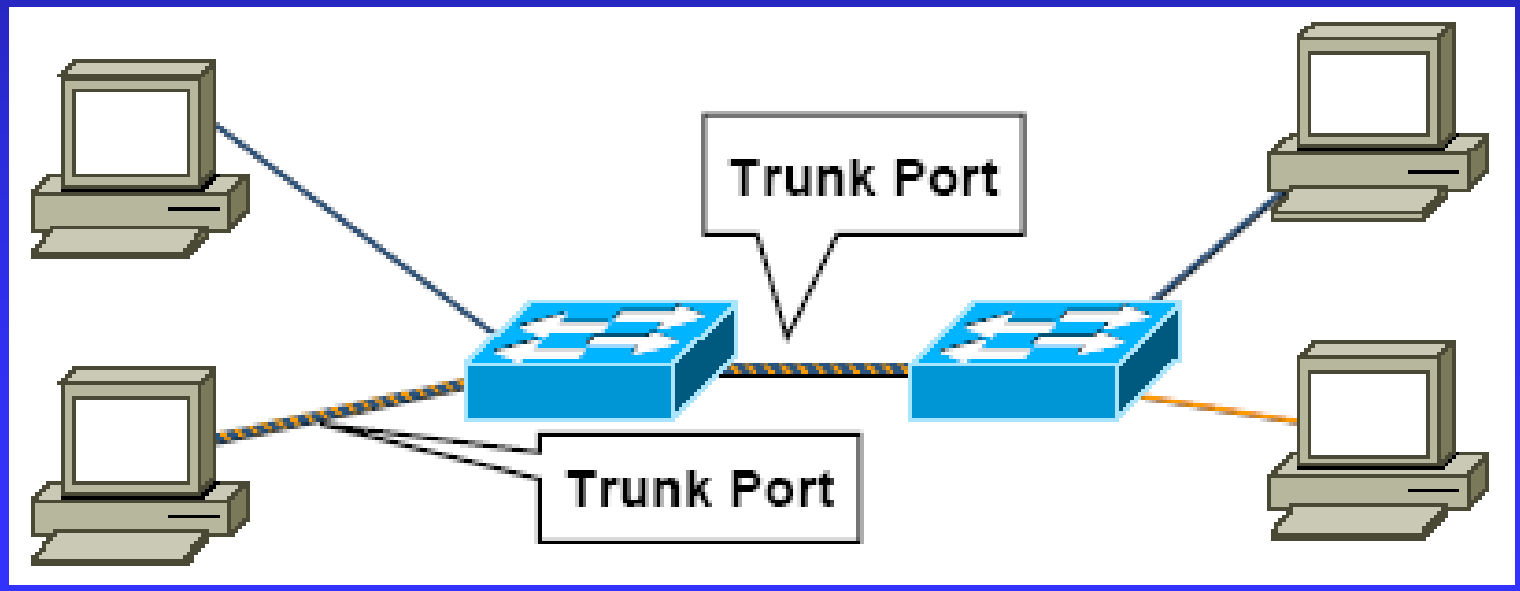
Protocol Structure - VLAN: Virtual Local Area Network and the IEEE 802.1Q

IEEE 802.1Q Tagged Frame for Ethernet:

7	1	6	6	2	2	2	42-1496	4
Preamble	SFD	DA	SA	TPID	TCI	Type Length	Data	CRC

VLAN Hopping - básico

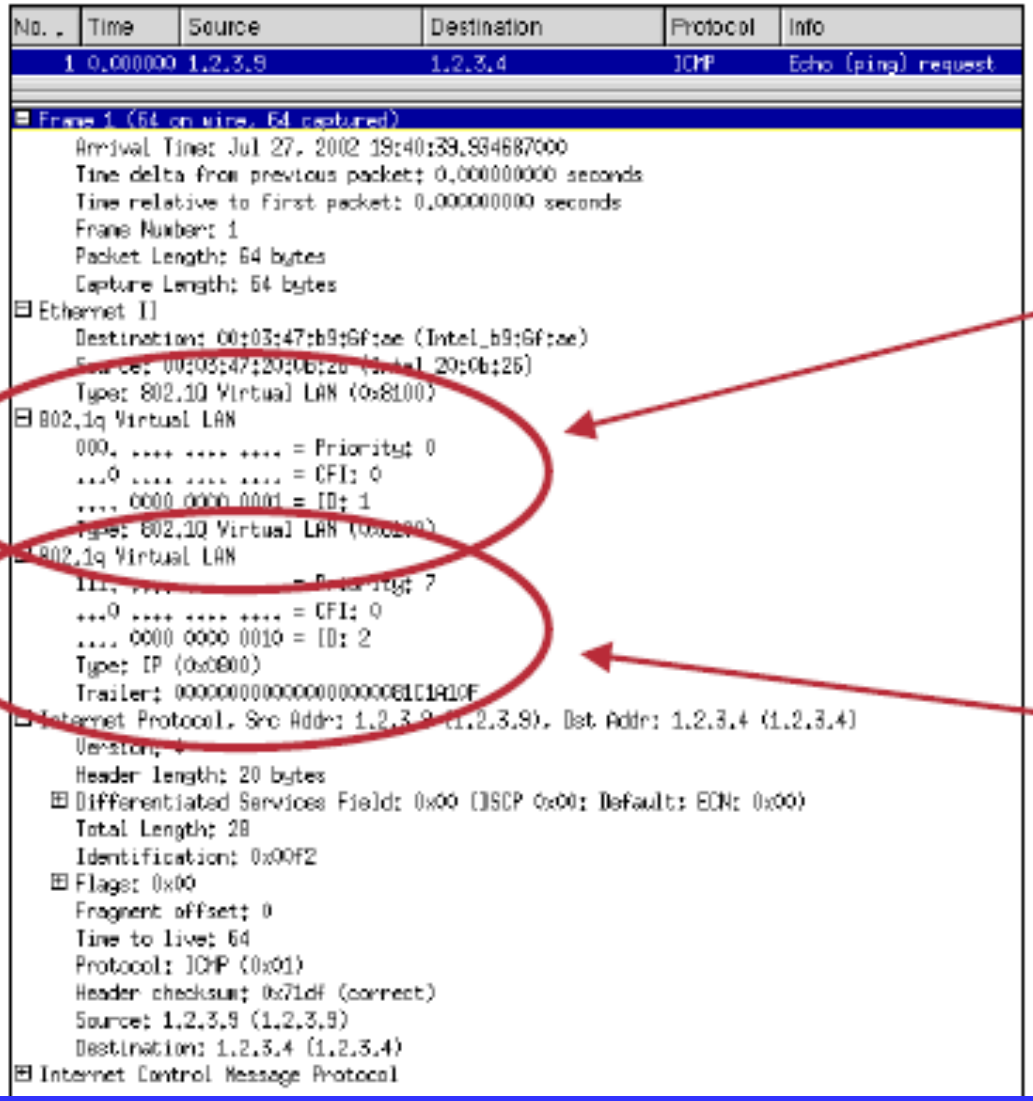
- Uma estação em uma porta de trunk se faz passar por um switch enviando sinalização DTP ou 802.1q



VLAN Hopping – double tagging

- Envie dois tags, o switch só desencapsula uma vez
- Unidirecional, mas efetivo mesmo se a porta de trunk está OFF

VLAN Hopping – double tagging



Outer Tag, Attacker VLAN

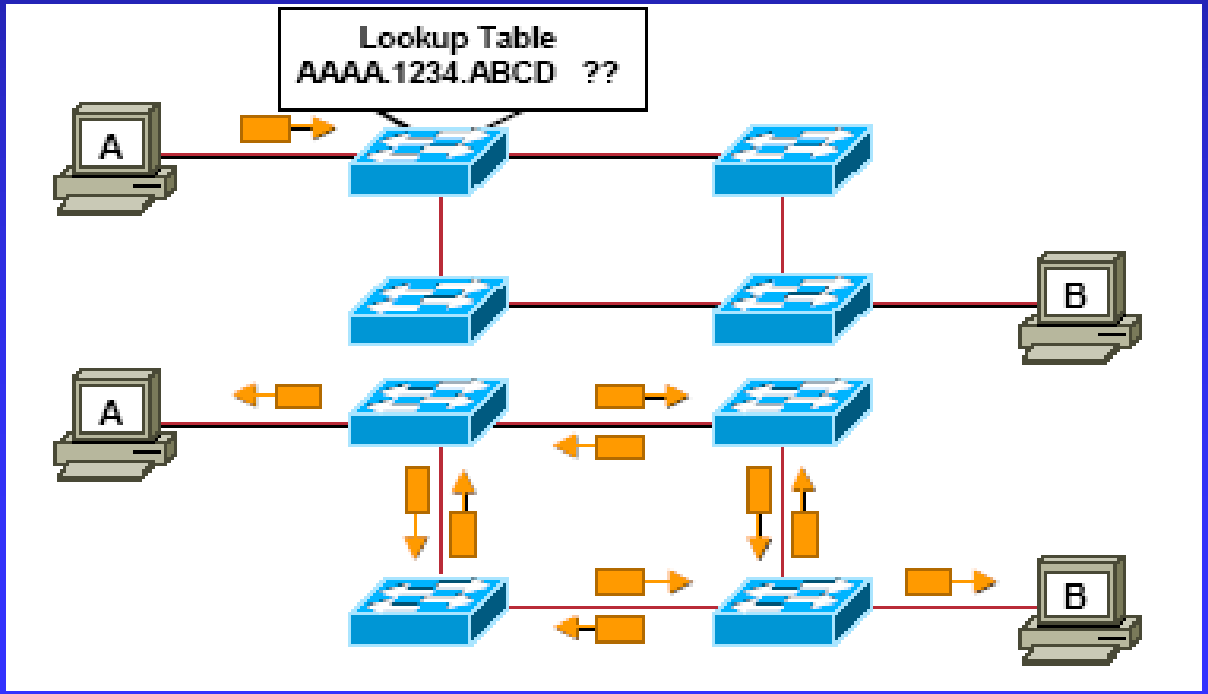
Inner Tag, Victim VLAN

Contra-medidas

- Desabilitar auto-trunk, mas depende do equipamento
- Use uma VLAN só para trunking
- Desabilite as porta fora de uso e as ponha em uma VLAN
- Todas as portas de usuário devem estar com trunking e DTP em OFF

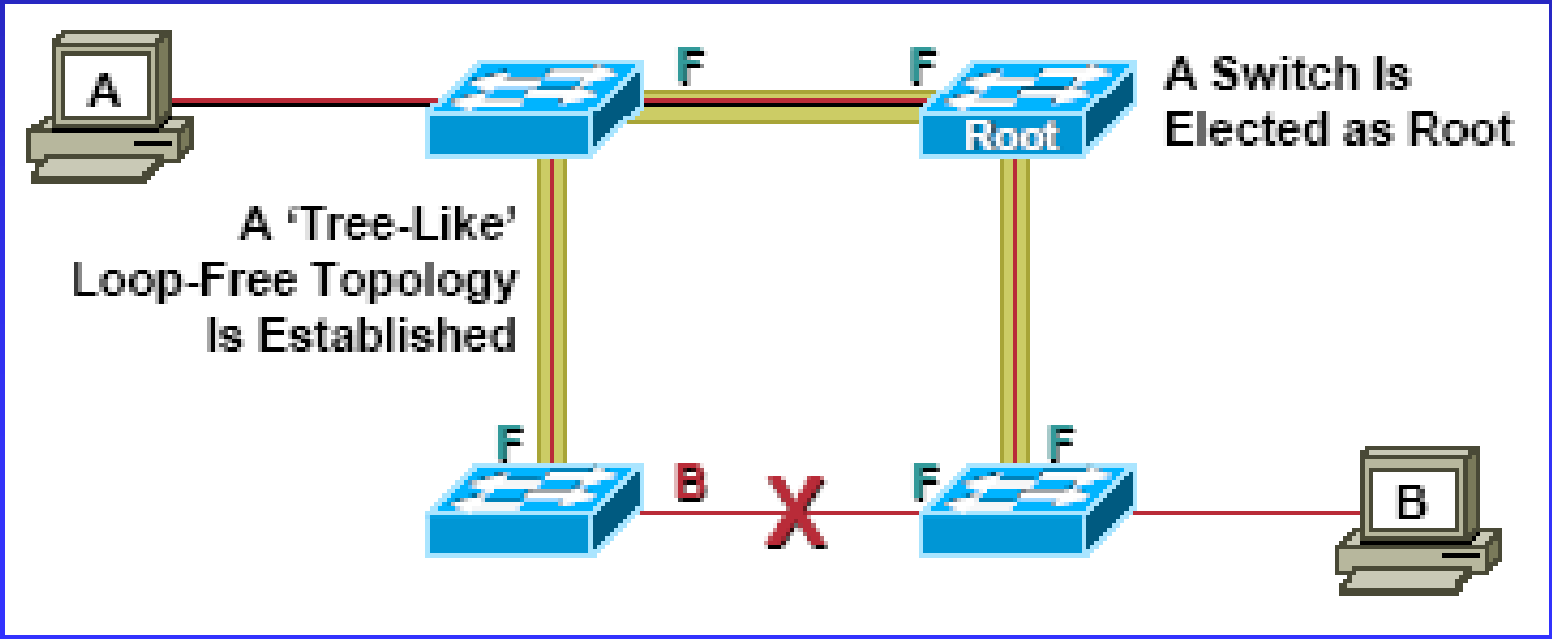
STP – Spanning Tree Protocol

- IEEE 802.1d
  - Protocolo destinado a manter topologias redundantes sem loops
- Loops causam broadcast storm

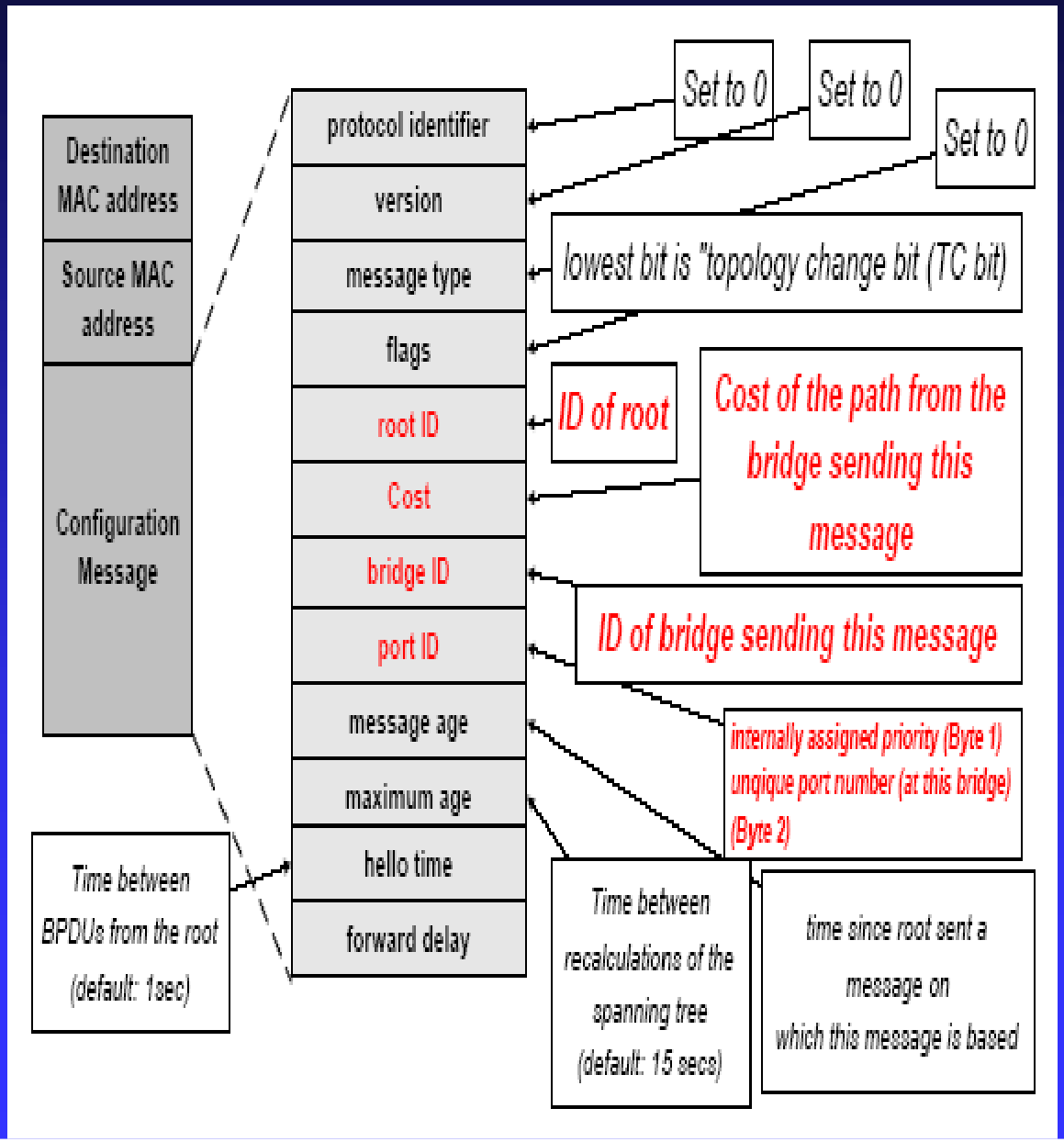


STP – Spanning Tree Protocol

- Baseado na velocidade do link e em um número arbitrário (prioridade) é calculado um custo para cada link
- Com o custo, o algoritmo de Dijkstra define uma árvore que é implementada com o bloqueio seletivo dos enlaces



STP – Spanning Tree Protocol

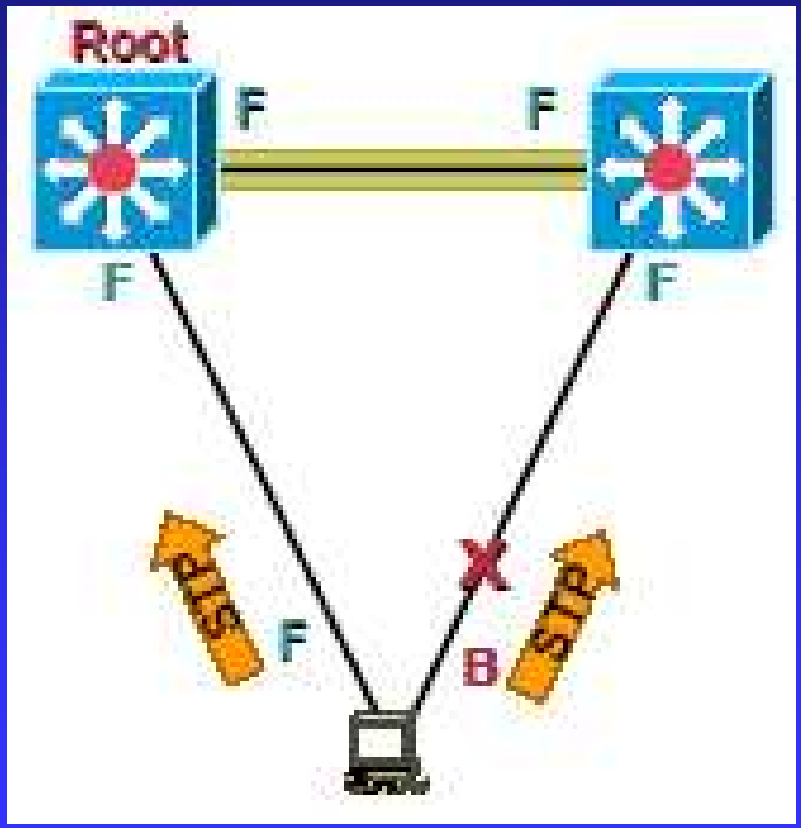


STP – ataques e métodos

- Quedas de enlace ou da raíz levam 30-45 seg para se propagar
- Injeção de BPDUs falsos
  - DoS
  - Tornar-se a raíz da árvore
  - Combinado com MAC flooding permite escutar tráfego entre switches

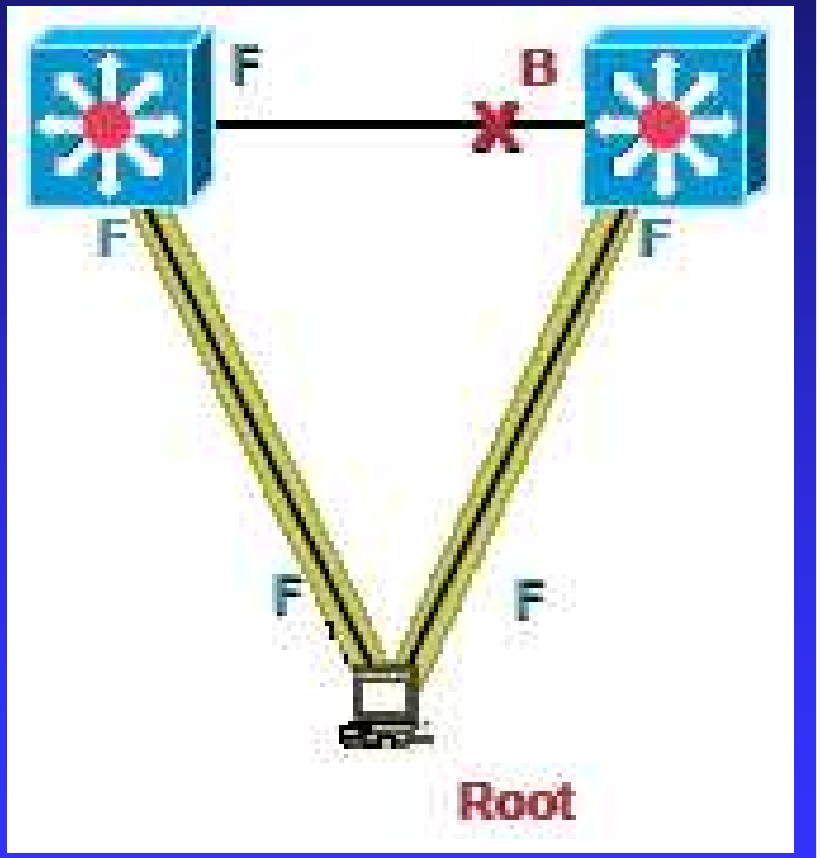
Tornando-se root 1/2

- Envia-se um BPDU para se tornar root



Tornando-se root 2/2

- Após a árvore ser recalculada, pode-se fazer:
  - MITM
  - DoS



Juntando os pedaços...

- Passo 1:
  - Faça MAC flooding
- Passo 2:
  - Envie BPDUs com prioridade igual a zero
- Passo 3:
  - Espere um pouquinho
- You´ve got the power!

Contra-medidas

- Não desabilite o STP!
  - Self inflicted DoS...
- BPDU guard
  - Desabilita a porta se ela mandar BPDUs
- Root guard
  - Desabilita a porta que se tornaria root com anúncio de BPDUs

Comentários finais

- A camada 2 está desprotegida
- Criptografar todo o tráfego talvez seja a única opção para sigilo e privacidade, como proposto com tecnologias wireless.
- Mas ainda se fica exposto a várias formas de DoS
- É preciso repensar toda a camada 2 para se obter um modelo de segurança.