Pontifícia Universidade Católica do Rio Grande do Sul

Faculdade de Engenharia

Programa de Graduação em Engenharia da Computação

**ARP-*Poisoning***

Gabriel Chieza Chiele

Maiki Buffet

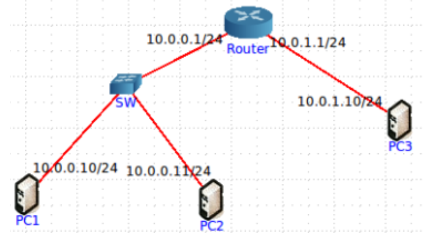
Porto Alegre, 18 de setembro de 2017

**Sumário**

1. Tarefa.................................................................................................3
2. Protocolo ARP....................................................................................4
3. Execução do Ataque...........................................................................5
4. Referências Bibliográficas...................................................................8
5. **Tarefa**

O objetivo geral deste trabalho consiste em desenvolver uma aplicação utilizando *raw sockets* para que seja realizado um ataque de *ARP poisoning* combinado com *man-in-the-middle.*

A tarefa foi realizada na linguagem de programação C, utilizando a seguinte topologia.



**Figura 1: Topologia – PC1 e *Router* são as vítimas e PC3 o atacante.**

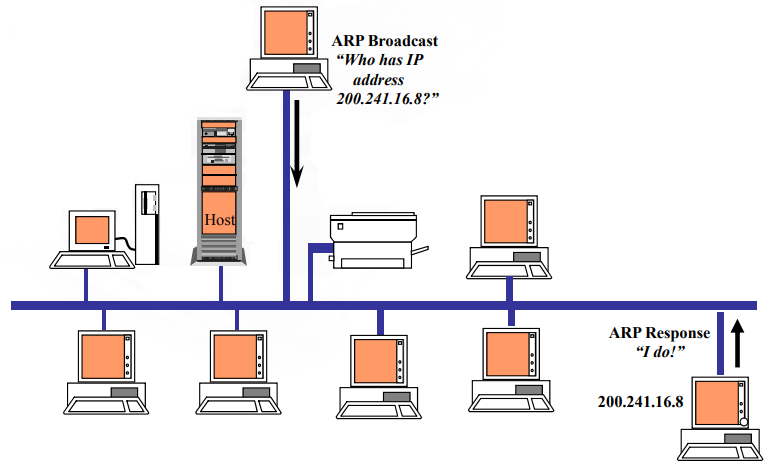
1. **Protocolo ARP**

O protocolo ARP (Address *Resolution Protocol*), é um protocolo da camada de enlace da pilha de protocolos (camada 2), pois ele opera apenas dentro dos limites de uma única rede, nunca sendo roteado entre nós de redes.

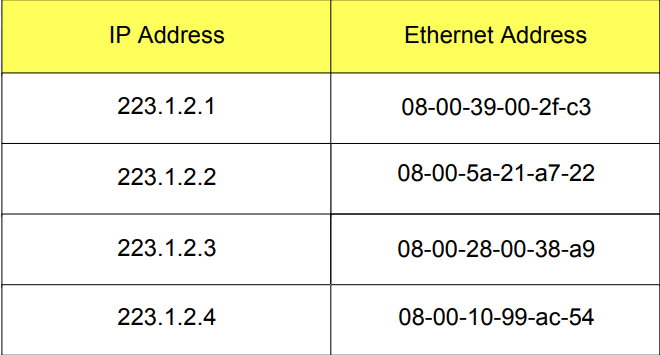
O ARP é utilizado para mapear endereços de rede para um endereço físico (*MAC – Media Access Control*) em redes IPV4. Para redes IPV6 este protocolo foi substituído pelo protocolo NDP (*Neighbor Discovery Protocol*).

**Figura 2: Pilha de protocolos.**

O processo de tradução de endereços que o protocolo ARP implementa é baseado no uso de mensagens *broadcast* e caches para armazenamento dos endereços resolvidos. Estas caches são chamadas de tabelas ARP, e elas contém os endereços IP e MAC referentes a uma máquina. Estas tabelas estão presentes em todas as máquinas que implementam este protocolo.

As tabelas ARP são preenchidas da seguinte maneira: um host A envia uma mensagem ARP *Request* que contém o endereço IP que ele necessita, então a máquina que possui o endereço IP requisitado responde com uma mensagem ARP *Reply*. Durante este processo de requisições, todas as máquinas conectadas a rede em questão, atualizam suas tabelas ARP com o endereço *MAC* da máquina que respondeu.

**Figura 3: Ilustração do processo de tradução ARP.**



**Figura 4: Exemplo de tabela ARP.**

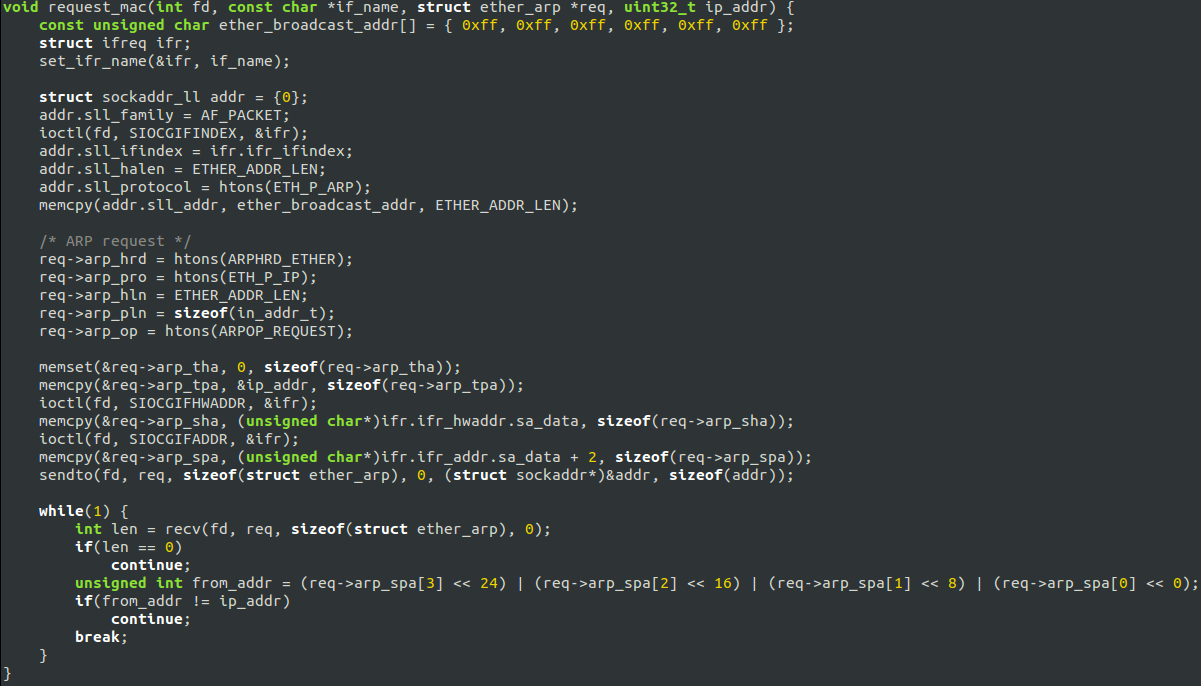
1. **Execução do Ataque**

O ataque de ARP *Poisoning* é realizado explorando a funcionalidade do protocolo ARP, como este protocolo utiliza tabelas cache internas, um usuário malicioso ao alterar o valor desta tabela, pode se passar por outra máquina, redirecionando o fluxo de mensagens de uma rede.

Para realizar um ataque ARP *Poisoning*, dividimos o processo em três etapas, estas que são:

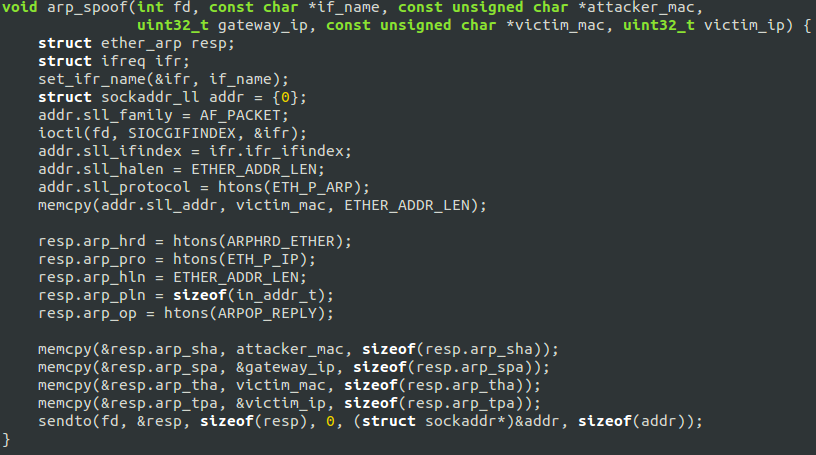
1. Descobrir o endereço *MAC* das máquinas que serão atacadas:

Para descobrirmos os endereços *MAC* das máquinas que serão atacadas, é realizado um envio de mensagens ARP *Request* com o endereço IP das máquinas em questão. Para isto utilizamos o método *request\_mac*, cuja implementação se encontra na figura abaixo.



**Figura 5: Código de criação, envio e recebimento de resposta de mensagens ARP *Request***.

1. Alterar a tabela ARP de um *host* na rede:

Com o endereço MAC das máquinas, podemos então, criar e enviar mensagens ARP *Reply* para a vítima do ataque. Para isto utilizamos o método *arp\_spoof*, cuja implementação se encontra na figura abaixo.

**Figura 6: Código de criação e envio de mensagens ARP *Reply*.**

1. Alterar a tabela ARP do *gateway* da rede:

Após termos realizado a modificação da tabela ARP da vítima, utilizamos novamente o método *arp\_spoof*, agora para o *gateway* da rede.

**Figura 7: Chamada consecutivas do método *arp\_spoof*, invertendo os campos *gateway\_ip*, *victim\_mac* e *victim\_ip*.**

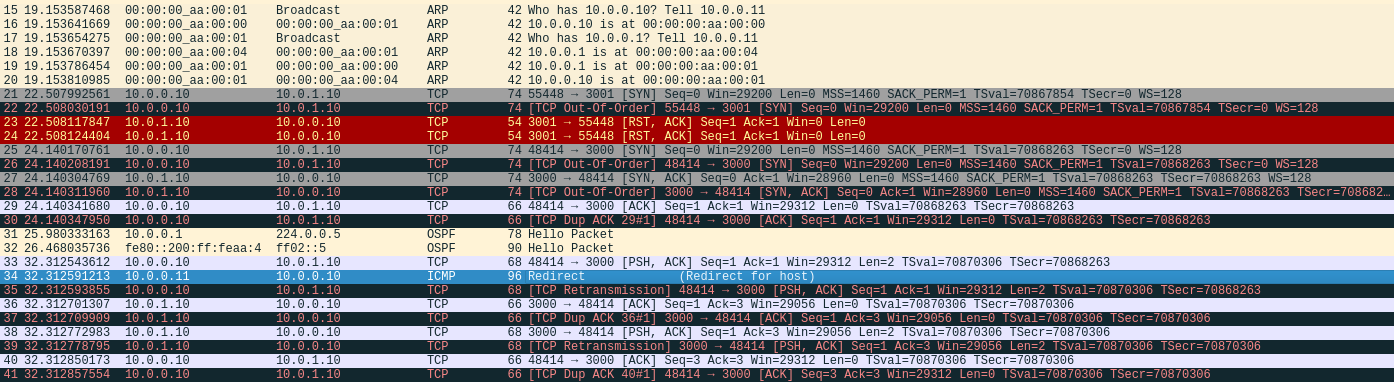
**Detalhamento das etapas:**

1 - Através da máquina atacante, acionamos o *arpspoof* para que sejam continuamente atualizadas (de forma forçada), as tabelas arp das máquinas Cliente e Gateway – ver figura 8.

2 – Após o *envenenamento* das tabelas, e com o encaminhamento de IP ativo, a máquina Cliente abre uma conexão com a máquina Servidor, sendo o trajeto do pacote TCP o seguinte:

Cliente → Atacante → Gateway → Servidor

E o retorno:

 Servidor → Gateway → Atacante → Cliente

**Figura 8: Imagem detalhada do tráfego através da rede, passando pela máquina atacante – via Wireshark.**

1. **Referências Bibliográficas**

techrepublic.com/article/tcp-hijacking/

exploit-db.com/papers/13587/

ce.sharif.edu/courses/79-80/2/ce443/projects/delivered/6/index.html

ettercap.github.io/ettercap/

datenterrorist.wordpress.com/2007/07/06/programming-tcp-hijackingtools-in-perl/

cecs.wright.edu/~pmateti/InternetSecurity/Lectures/TCPexploits/

github.com/yo2seol/mininet\_tcp\_hijacking

github.com/NickStephens/WRATH

github.com/r00t-3xp10it/Morpheus

securiteam.com/tools/5QP0P0K40M.html

tenouk.com/Module43a.html

https://pt.wikipedia.org/wiki/Endere%C3%A7o\_MAC

https://www.inf.ufes.br/~zegonc/material/Redes\_de\_Computadores/O%20Protocolo%20ARP.pdf