Análise do comportamento de um ataque de negação de serviço via SYN-FLOOD a um servidor web

Diego da Silva Pereira[1], Alexandre Henrique Dantas[2], Harrison Gustavo Silva de Medeiros[2]

[1]Professor de Redes de Computadores – IFRN Câmpus Currais Novos. e-mail: diego.pereira@ifrn.edu.br
[2]Aluno do 3º ano do Curso Técnico Integrado em Informática – IFRN Câmpus Currais Novos. e-mail: netbuntuedul@gmail.com

Resumo: O artigo tratou de um dos ataques mais comuns a servidores *web*, o ataque de negação de serviço. Este consiste em deixar um serviço de rede temporariamente inoperante devido ao alto consumo dos recursos dos servidores. Trouxe de maneira detalhada todo o funcionamento do ataque, dando ênfase aos protocolos envolvidos e como eles se relacionam. Para isso foi elaborado um cenário, contendo um servidor *web* Apache2, um observador capturando o tráfego da rede e um cliente/atacante, o qual executou as duas situações, uma conexão normal e um ataque. Foi observado o potencial deste ataque e a maneira que o servidor se comporta, afinal este não consegue distinguir qual solicitação de conexão é falsa. O trabalho se faz necessário para aprender com requintes de detalhe cada mensagem que participa do processo, para a partir daí poder traçar uma estratégia de defesa, seja ela pré ou pós ataque. Outro fator de destaque é a abordagem em um trabalho futuro do ataque DDoS, ou seja, ataque de negação de serviço distribuído, o qual não possui um único atacantes, e sim diversas máquinas comandadas por uma única entidade através de *softwares* instalados de maneira quase transparente ou de difícil percepção pelos administradores e demais usuários.

Palavras–chave: ataque, DoS, segurança, *syn-flood*

1. INTRODUÇÃO

Atualmente a rede mundial de computadores, popularmente conhecida como Internet, é responsável pela comunicação entre centenas de milhares de dispositivos nos mais diversos locais do planeta, entretanto, esta grande estrutura está vulnerável a inúmeros problemas, os mais preocupantes são aqueles que violam os princípios básicos da segurança da informação, como a integridade, confidencialidade e disponibilidade, sendo este último o alvo principal deste trabalho.

Dentre a variedade de serviços ofertados nessa grande rede, com certeza o serviço de hospedagem de páginas *web* é o mais amplamente utilizado, portanto, é o que sofre o maior número de ataques com intuito de comprometer o funcionamento adequado do serviço, dentre várias formas de ataques possíveis, uma será objeto de estudo neste trabalho, o DoS (*Denial of Service*) ou Negação de Serviço.

O DoS consiste em tornar o serviço ofertado indisponível devido ao esgotamento de recursos do servidor, uma forma de ataque comum é o *syn-flood*, conforme TANENBAUM (2003) o ataque DoS via *syn-flood* consiste "envio de pacotes legítimos em grande quantidade, até o site entrar em colapso com a carga".

Logo, o presente trabalho terá como objetivo principal detalhar o funcionamento de um ataque de negação de serviço dando ênfase ao comportamento do servidor, pois a partir daí será possível traçar as melhores estratégias para executar táticas que possam garantir uma melhor margem de segurança para os diversos servidores *web* presentes na Internet.

As seções estão dispostas de maneira que possa ser feito uma leitura gradativa do tema proposto e um bom embasamento teórico sobre redes de computadores e suas aplicações. Na seção 2 é feito o embasamento teórico sobre redes de computadores, dando ênfase as formas de comunicação entre dispositivos de uma rede, as aplicações que essas redes podem oferecer e ainda como ocorre o ataque de negação de serviço. A seção 3 apresenta a infraestrutura do cenário utilizado para realização do experimento e a maneira que foi capturado os dados. Em seguida, na seção 4 é realizada a apresentação dos dados e a discussão sobre os resultados obtidos. Por fim, na seção 5 é feito a conclusão do trabalho, frisando os apontamentos principais e perspectivas para trabalhos futuros.

ISBN 978-85-62830-10-5

2. REFERENCIAL TEÓRICO

A comunicação entre dispositivos de uma rede de computadores é regida por uma pilha de protocolos, de acordo com KUROSE e ROSS (2010) "Um protocolo define o formato e a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou no recebimento de uma mensagem ou outro evento.", conforme a pilha de protocolos adotada pela rede no processo de comunicação, cada protocolo terá sua função específica em cada etapa deste processo. Atualmente, a pilha de protocolos mais amplamente utilizada na Internet é a TCP/IP, entretanto ela sofreu uma leve adaptação, trabalhando em 5 camadas, e não apenas 4 como é definida originalmente. A Figura 1 ilustra a arquitetura de camadas da pilha TCP/IP e da pilha adaptada utilizada na Internet.

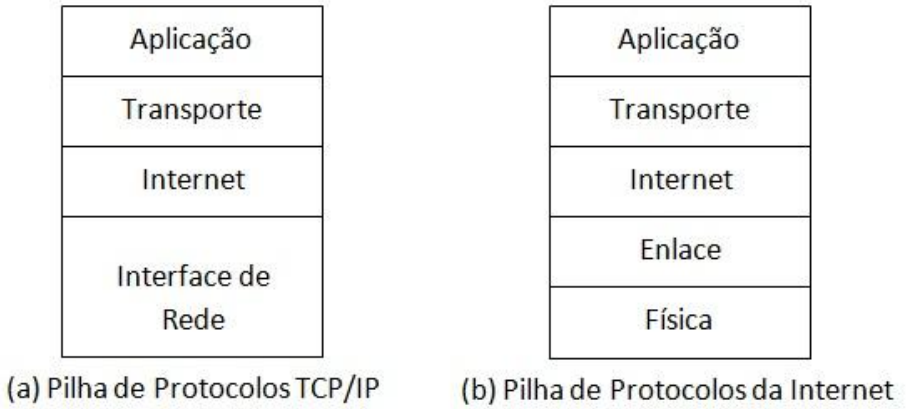

Figura 1 – Pilha de protocolos TCP/IP (a) e a pilha de protocolos utilizada na Internet (b)

2.1. PILHA DE PROTOCOLOS DA INTERNET

A pilha de protocolos adotada na Internet é formada por cinco camadas bem definidas, onde cada uma delas terá uma função específica dentro do processo de comunicação, e, além disso, contará com a presença de um protocolo. Com intuito de facilitar o entendimento do trabalho, é necessário detalhar o funcionamento de cada camada presente neste modelo.

2.1.1. Camada de aplicação

Essa camada se encontra acima da camada de transporte. Ela tem função de fornecer ao usuário final o serviço solicitado pelo mesmo, seja de uma página da Internet (HTTP), de um correio eletrônico (SMTP) ou até mesmo para transferência de arquivos (FTP). TANENBAUM (2003) menciona que novos protocolos de aplicação foram inseridos com o passar do tempo, entre eles estão o DNS (*Domain Name Service*) que controla os nomes de hosts para locais de rede, o DHCP (*Dynamic Host Configuration Protocol*) que atua na configuração dinâmica de estações da rede e até mesmo o próprio HTTP, afinal os protocolos foram sendo criados de acordo com a necessidade que os usuários tinham por determinados serviços.

2.1.2. Camada de transporte

A finalidade desta camada é permitir que as entidades origem e destino do processo de comunicação mantenham uma conversação. Dois protocolos principais foram definidos para atuar nessa camada, o TCP (*Transmission Control Protocol* – Protocolo de Controle de Transmissão) e o UDP (*User Datagram Protocol* – Protocolo de Datagramas de Usuário). Cada um deles é recomendado em determinada situação, neste trabalho só será necessário abordar o TCP, pois este provê serviços orientados à conexão para os protocolos da camada de aplicação, que neste caso será o HTTP. O UDP é amplamente utilizado em situações que não é necessário um controle da conexão entre os participantes do processo de comunicação.

ISBN 978-85-62830-10-5

2.1.3. Camada de rede

Conforme KUROSE e ROSS (2010) "a camada de rede da Internet é responsável pela movimentação, de um dispositivo para outro, de pacotes chamados datagramas", o processo de escolha do percurso que o datagrama deve seguir na rede é chamado de roteamento e o protocolo responsável em atuar nesta camada é o IP (*Internet Protocol* – Protocolo Internet).

2.1.4. Camada de enlace

Os serviços prestados pela camada de enlace dependem do protocolo específico utilizado no enlace de dados, entretanto conforme TANENBAUM (2003) "a principal tarefa da camada de enlace é transformar um canal de transmissão bruta em uma linha livre para transmissão".

2.1.5. Camada física

A camada física trata da transmissão de *bits* por um determinado canal de comunicação, a principal preocupação desta camada é movimentar esses *bits* individualmente de um nó para outro.

2.2. O ESTABELECIMENTO DE UMA CONEXÃO TCP

O protocolo TCP é dito orientado a conexão, pois antes do início da transmissão de dados por um processo da camada de aplicação, é feito um contato prévio pela camada de transporte para estabelecer critérios adequados para o sucesso da comunicação.

O processo citado acima é chamado de *three-way handshake*, ou, quando traduzido para português, aperto de mãos de três vias, para maioria dos autores apenas *handshake* de três vias ou, ainda, apresentação de três vias conforme KUROSE e ROSS (2010).

De acordo com STEVENS, FENNER e RUDOFF (2005) apresentação de três vias é composta por três segmentos TCP e ocorre conforme descrito a seguir.

Antes de iniciar o processo de estabelecimento da conexão, o servidor deve estar preparado para aceitar uma conexão entrante (*passive open* – abertura passiva). Nesse contexto, o cliente requer uma abertura ativa (*active open*), onde o cliente envia um seguimento de sincronização (SYN – synchronize) que informa ao servidor o seu número inicial de sequência TCP.

No segundo momento, o servidor reconhece (ACK – acknowledge) o SYN do cliente e também envia seu próprio SYN, este deve conter o seu número de sequência TCP, o detalhe é que o SYN e o ACK são enviados em um único segmento.

A terceira e última etapa trata apenas do reconhecimento (ACK) do cliente pelo recebimento do SYN enviado pelo servidor anteriormente, nesse momento o cliente concretiza o estabelecimento da conexão. Acrescenta-se que apenas no último segmento enviado pelo cliente é possível enviar carga útil, ou seja, transporte real de dados da camada de aplicação, pois até o então a conexão não estava realmente estabelecida. Todo o processo é exibido em detalhes conforme a Figura 2.

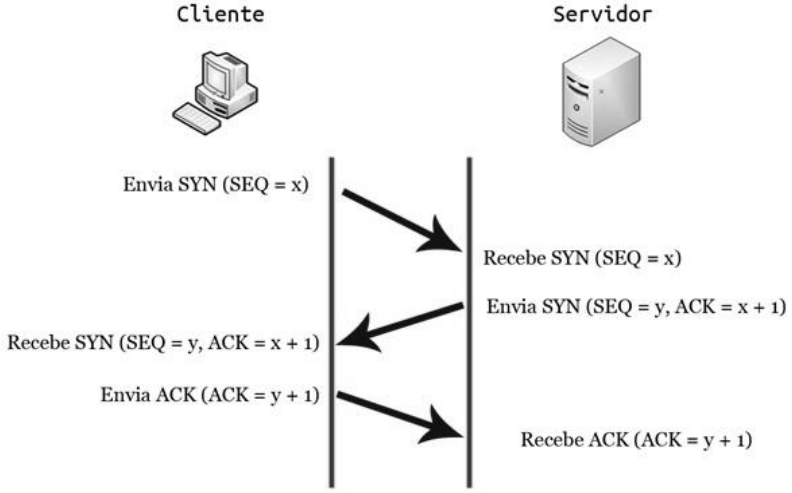

Figura 2 – Estabelecimento de uma conexão TCP (three-way handshake)

ISBN 978-85-62830-10-5

2.3. O SERVIÇO HTTP (Hypertext Transfer Protocol)

Conforme TORRES (2009) a transferência de documentos hipermídia é feita através do protocolo HTTP. Um servidor *www* hospeda o site, enquanto um cliente (*browser*) faz a requisição dos documentos lá contidos. No servidor essa transferência é feita usando a porta 80 do protocolo TCP.

A navegação *web*, como foi dito anteriormente, também faz uso do paradigma cliente-servidor, ou seja, sempre terá um ofertante (servidor) e um consumidor(cliente) para o serviço oferecido, sendo assim, devido as necessidades da camada de aplicação (HTTP) o protocolo da camada de transporte é o TCP. Com isso, sempre que o cliente HTTP desejar se conectar a um servidor terá que efetuar o *three-way* handshake.

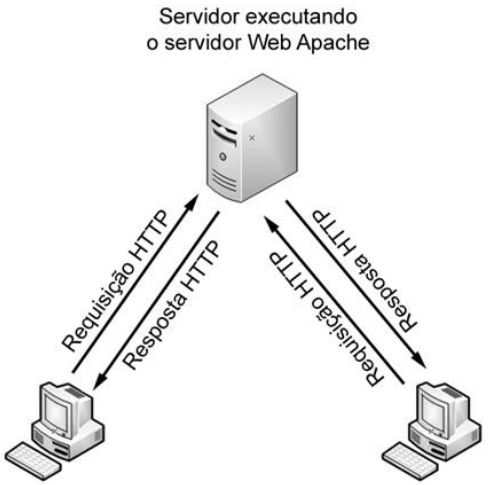

Figura 3 – Exemplo de comunicação via HTTP

2.4. ATAQUE DE NEGAÇÃO DE SERVIÇO

Nas palavras de CHESWICK, BELLOVIN e RUBIN (2005) os ataques de negação de serviço (DoS – *Denial of Service*) constituem a simples superutilização de um serviço – levando *software*, *hardware* ou enlaces de rede além da capacidade para eles concebida. A intenção é desativar ou degradar a qualidade de um serviço, e, geralmente, esse é um objetivo modesto.

Esses ataques são de fácil percepção, afinal ocorre o comprometimento do serviço ofertado, a grande dificuldade está em localizar a origem, na maioria dos casos eles envolvem pacotes gerados com endereços de retorno falsificados e aleatórios.

Uma forma de ataque de negação de serviço ainda mais complexa para identificar a origem é quando ele ocorre de maneira distribuída, neste caso ele recebe o nome de DDoS (*Distributed Denial of Service* – Negação de Serviço Distribuído). Em um DDoS vários *hosts* da internet participam do ataque, entretanto a grande maioria deles são cúmplices inconscientes, pois foram comprometidos de alguma maneira e equipados com o código malicioso necessário para efetuar o ataque. Em situações assim, a dificuldade para recuperar-se do ataque é ainda maior, afinal são diversas origens, contrário ao DoS simples, que possui apenas um atacante.

Esse tipo de ataque pode ocorrer de diversas maneiras e comprometer o serviço em qualquer parte do processo de comunicação, pois basta uma única camada da pilha de protocolos está comprometida para ocorrer falhas de comunicação entre os dispositivos de rede. Eles podem travar um sistema, esgotar a capacidade de memória principal ou secundária (discos rígidos), exaurir a tabela de processos ou a disponibilidade da CPU, exceder o número de conexões TCP, comprometer o desempenho do enlace da rede com tráfego excessivo, entre outras.

Para que esses ataques ocorram no enlace de rede, basta que o enlace de conexão do atacante tenha uma capacidade maior que o do alvo, a inundação ocorre através de pacotes aleatórios, sendo apenas necessário que o campo de destino e a soma de verificação estejam corretos.

ISBN 978-85-62830-10-5

O ataque pode ocorrer na camada de rede, isso ocorre através da manipulação dos protocolos de roteamento, que podem simplesmente redefinir a máscara de sub-rede do alvo, nesse caso, o servidor também fica impossibilitado de comunicar-se com a rede.

Diversas formas de ataque de negação de serviço podem ser abordadas aqui, entretanto a que receberá ênfase e é alvo da pesquisa deste trabalho é a *syn-flood* ou inundação SYN que ocorre na camada de transporte com o protocolo TCP.

2.4.1. Ataque DoS via SYN-FLOOD TCP

O ataque DoS tem como objetivo principal consumir os recursos do alvo de modo que ela não possa oferecer um determinado serviço. Os ataques podem ser classificados de acordo com o tipo de recurso consumido, ou seja, se o recurso é interno ou pertence à capacidade de transmissão de dados (enlace). No caso do *syn-flood* TCP trata-se de um ataque a recursos internos do alvo.

STALLINGS (2008) descreve o ataque de negação de serviço via *syn-flood* conforme os passos a seguir:

1) O atacante começa a enviar para o alvo (servidor *web*) diversos segmentos SYN (sincronismo) com informações de endereço IP de retorno falso;

2) Cada segmento SYN é uma solicitação de abertura de conexão TCP. Para cada segmento desse tipo recebido, o servidor *web* responde com um segmento SYN/ACK (sincronismo/confirmação), tentando estabelecer uma conexão com a entidade solicitante;

3) O servidor *web* mantém uma estrutura de dados para cada solicitação SYN esperando por uma resposta (ACK) do cliente, entretanto, neste caso a resposta não chegará, afinal trata-se de um IP de origem falso e não interessa ao atacante concretizar a conexão. O protocolo TCP aguarda por um intervalo de tempo, e é justamente durante esse intervalo que solicitações de conexões legítimas são negadas, devido ao servidor está aguardando para completar as falsas conexões "meio abertas". Todo o processo é ilustrado na Figura 4 adiante.

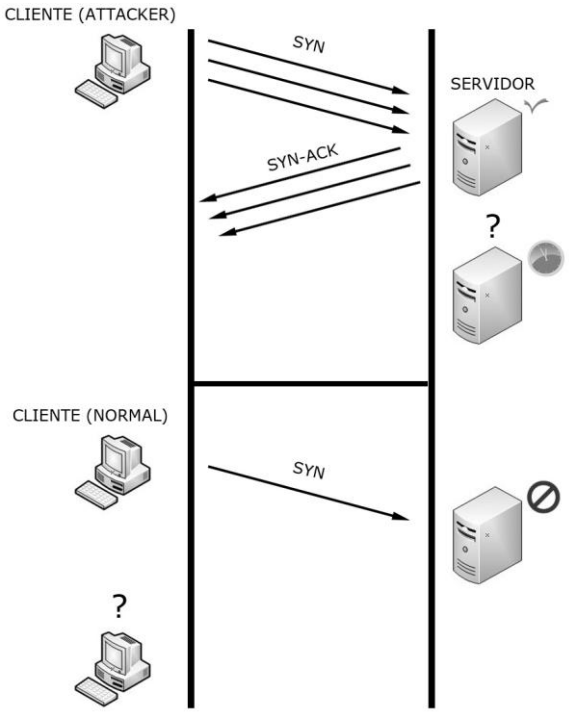

Figura 4 – Ataque de negação de serviço via SYN-FLOOD a servidor *web*

ISBN 978-85-62830-10-5

3. MATERIAIS E MÉTODOS

Com objetivo de observar todo o comportamento do ataque de negação de serviço via *syn-flood* foi criado um cenário controlado e restrito em uma única rede local sem acesso a internet. Este foi desenvolvido no laboratório de redes de computadores do IFRN – Câmpus Currais Novos.

3.1 CENÁRIO PROPOSTO

Para realizar o ataque de negação de serviço via *syn-flood* a um servidor web foi necessário utilizar três estações dentro da mesma rede local, sendo uma delas um servidor *web* Debian 6.0.5 Apache2 versão 2.2.16, outra máquina Windows XP *Service Pack* 4 com o analisador de protocolos de rede Wireshark versão 1.8.1, e uma terceira com sistema operacional Linux Ubuntu 11.10 com o sofware HPING versão 3, este último será utilizado para executar o ataque DoS ao servidor *web*. Os endereços ips das máquinas e a sua descrição estão de acordo com a Tabela 1.

Tabela 1 – Descrição geral da máquinas presentes no cenário proposto

Identificação	Endereço IP	Sistema Operacional	Configuração Básica
Servidor *web*	172.16.20.1/24	Debian 6.0.5	Processador Dual Core 2.6 GHz com 512 Mb de RAM
Observador	172.16.20.2/24	Windows XP SP4	Processador Dual Core 2.6 GHz com 512 Mb de RAM
Atacante	172.16.20.3/24	Ubuntu 11.10	Processador Dual Core 2.6 GHz com 1 Gb de RAM

3.2 METODOLOGIA

A execução do experimento consistiu em utilizar o programa Hping3 via terminal de comando no atacante ao servidor *web*. Foram executados 40 ataques, sendo o cenário retomado para o estado inicial após cada um deles, ou seja, tanto o servidor *web* quanto o atacante tinham seus sistemas operacionais reiniciados após a conclusão do ataque. Nesse universo de 40 amostras, em apenas 01 execução o servidor *web* não teve seu serviço comprometido totalmente, ou seja, 97,5% dos ataques obtiveram êxito.

4. RESULTADOS E DISCUSSÃO

Com intuito de acompanhar todo o ataque ao servidor, a máquina observador teve a função de capturar todo o tráfego de dados entre o servidor *web* e o atacante. A Figura 5 mostra a conexão realizada normalmente entre o cliente e o servidor. Observa-se que o *three-way handshake* é realizado na abertura e no fechamento da conexão, entre eles ocorre o envio dos dados do protocolo da camada de aplicação, no caso HTTP.

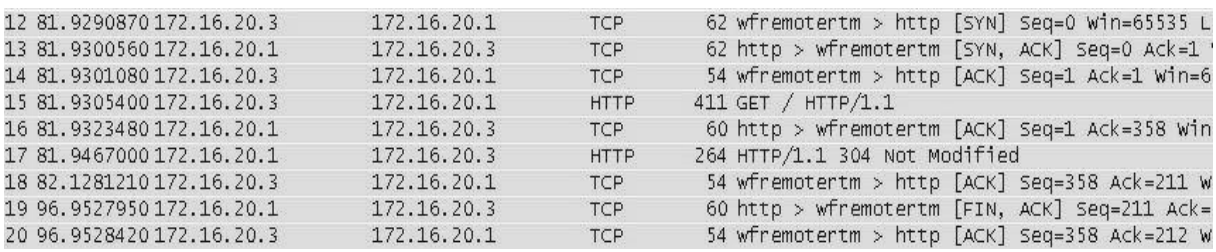

Figura 5 – Conexão TCP entre o cliente e o servidor

O software HPING possui diversas finalidades e uma potencialidade muito grande, pois permite ao usuário manipular de inúmeras maneiras as informações, independente da camada que deseja-se manipular. Neste caso, como já foi dito anteriormente, o objetivo foi capturar o ataque de negação de serviço na íntegra, e este poder ser observado na Figura 6. Como abordado na seção 2, o ataque consiste na solicitação de aberturas de conexões falsas, ou seja, conexões que não serão concluídas.

ISBN 978-85-62830-10-5

6983	57.414483	172.16.20.3	172.16.20.1	TCP	54	4584 > http [SYN] Seq=0 Win=512 Len=0
6984	57.414606	172.16.20.3	172.16.20.1	TCP	54	4585 > http [SYN] Seq=0 Win=512 Len=0
6985	57.414726	172.16.20.3	172.16.20.1	TCP	54	4586 > http [SYN] Seq=0 Win=512 Len=0
6986	57.414873	172.16.20.3	172.16.20.1	TCP	54	4587 > http [SYN] Seq=0 Win=512 Len=0
6987	57.414994	172.16.20.3	172.16.20.1	TCP	54	4588 > http [SYN] Seq=0 Win=512 Len=0
6988	57.415113	172.16.20.3	172.16.20.1	TCP	54	4589 > http [SYN] Seq=0 Win=512 Len=0

Figura 6 – Sequência de solicitações de abertura de conexão via *syn-flood*

Uma forma de comprovar a existência dessas conexões no servidor *web* é utilizar o comando *netstat* no terminal de comando, que este exibirá as conexões ativas, o estado em que eles se encontram e outras informações relevantes, como endereços ip envolvidos, protocolo da camada de transporte e porta da camada de aplicação utilizada, isto pode ser observado na Figura 7.

Um ponto de extrema relevância é observar que o estado de todas as conexões é SYN_RECEBIDO, ou seja, o servidor recebeu a solicitação de abertura, enviou o SYN_ACK de confirmação para estabelecimento da conexão, entretanto, o cliente que neste caso é um atacante, não enviou o ACK de confirmação para concretizar a conexão em definitivo e com isso o servidor aguarda a chegada desse segmento. Isso acarreta uma sobrecarga na tentativa de estabelecimento de conexões, afinal o cliente envia mais solicitações incessantemente. Vale destacar que a Figura 7 não traz todas as tentativas de conexões devido ao grande espaço que a imagem iria ocupar, tornando assim isso inviável.

```
Conexões Internet Ativas (sem os servidores)
Proto Recv-Q Send-Q Endereço Local          Endereço Remoto         Estado
tcp        0      0 172.16.20.1:80          172.16.20.3:48341       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48338       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48407       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48348       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48421       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48342       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48354       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48511       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48477       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48316       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48318       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48479       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48376       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48425       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48395       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48383       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48320       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48403       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48404       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48422       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48388       SYN_RECEBIDO
tcp        0      0 172.16.20.1:80          172.16.20.3:48411       SYN_RECEBIDO
--Mais--
```

Figura 7 – Conexões ativas no servidor *web* apresentadas através do comando *netstat*

Conforme dito anteriormente, cada segmento TCP que chega ao servidor é uma solicitação para iniciar uma nova conexão, isso também é possível de ser observado com a utilização do analisador de protocolos Wireshark. A Figura 8 ilustra justamente os detalhes de um dos milhares de segmentos TCP que chegaram ao servidor. Merece ser destacado a porta de destino, no caso 80, porta padrão do protocolo HTTP, e dentre todas as *flags* do TCP, a única que está ativa é a SYN, o que determina abertura de conexão com o pedido de sincronização (*synchronize*).

ISBN 978-85-62830-10-5

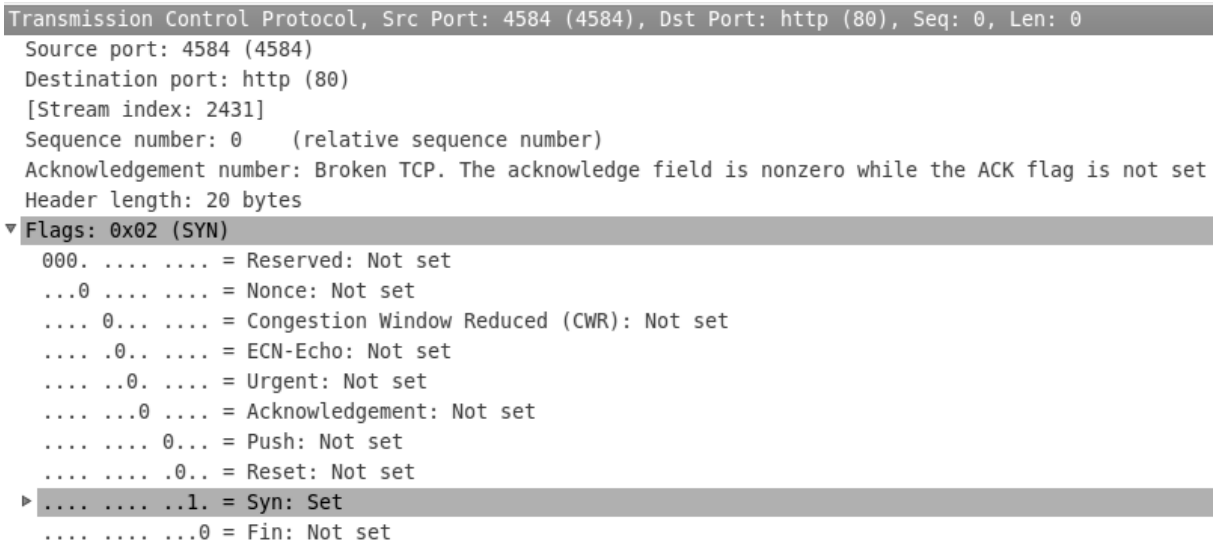

Figura 8 – Detalhes de um segmento SYN enviado pelo atacante

5. CONCLUSÕES

Atualmente a área de segurança da informação está cada vez recebendo mais cobrança, pois sempre existirá algum indivíduo na rede com objetivo de obter algum tipo de vantagem ou apenas prejudicar determinada organização. Por isso, o objetivo deste trabalho foi ilustrar de maneira detalhada o comportamento de um ataque de negação de serviço via *syn-flood*, dando ênfase à parte teórica através da visualização de todos os pacotes que transitam na rede durante o processo, e também a parte prática, afinal precisa-se entender o que está acontecendo para poder conseguir elaborar uma estratégia de defesa.

Com a conclusão do trabalho ficou claro que tipo de protocolos fazem parte da execução do ataque e que para combatê-lo é necessário atuar na camada de transporte, evitando que o servidor esgote seus recursos com o processamento dessas solicitações de conexão falsas. Para um próximo cenário será abordado o ataque DDoS, ou seja, o ataque de negação de serviço de maneira distribuída. Também é pretendido desenvolver estratégias básicas para tratar da prevenção desses ataques e ainda procedimentos que venham a ser adotados durante a execução dos mesmo.

REFERÊNCIAS

CHESWICK, W. R.; BELLOVIN, S. M.; RUBIN, A. D. **Firewalls e Segurança na Internet: repelindo o hacker ardiloso.** 2. ed. São Paulo: Bookman, 2005.

COMER, Douglas E. **Redes de Computadores e Internet.** 4. ed. Porto Alegre: Bookman, 2007.

KUROSE, J. F.; ROSS, K. W.; **Redes de Computadores e a Internet: uma abordagem top-down.** 5. ed. São Paulo: Pearson Education do Brasil, 2010.

RNP. Escola Superior de Redes. **Segurança da Informação de Redes e Sistemas.** Rio de janeiro: RNP, 2010.

STALLINGS, William. **Criptografia e segurança de redes: princípios e práticas.** 4. ed. São Paulo: Pearson Prentice Hall, 2008.

STEVENS, W. R.; FENNER, B.; RUDOFF, A. M. **Programação de Rede Unix: api para soquetes de rede.** 3. ed. Porto Alegre: Bookman, 2005.

TORRES, Gabriel. **Redes de Computadores.** Rio de Janeiro: Novaterra, 2009.

ISBN 978-85-62830-10-5