FACULDADE DE INFORMÁTICA DE
PRESIDENTE PRUDENTE

SOFTWARE PARA SCANNERS DE REDES - WIRESHARK

VINICIUS SOUZA VASCONCELOS DOS SANTOS
VITOR FONSECA VERONEZI

Presidente Prudente – SP

2017

SUMÁRIO

1 – INTRODUÇÃO

Tanenbaum (2003) em seu livro “Redes de Computadores”, no capítulo 8 – “Segurança de redes” cita que durante as primeiras décadas da existência das redes de computadores, elas eram utilizadas principalmente por pesquisadores universitários, com a finalidade de enviar mensagens e também por funcionários de grandes empresas, para compartilhar impressoras e documentos. Sob estas condições, a segurança nunca precisou ter maiores cuidados. Hoje em dia, com milhões de usuários, transações bancarias, *softwares* de receita federal e sistemas que necessitam de cuidados maiores, as redes precisaram evoluir, gerando assim melhoras significativas em seus protocolos e algoritmos, tornando-a mais segura.

Tanenbaum (2003) e também Kurose e Ross (2013) abordam em seus livros, um capítulo direcionado a segurança de redes, onde abordam temas como: criptografias, algoritmos de chave assimétrica, assinaturas digitais, *Firewalls*, segurança em redes sem fio entre outros tópicos. Eles explicam como estes atingem as camadas e seu funcionamento interno. Ao final de cada capítulo no livro “*Redes de Computadores e a internet: uma abordagem top-down”* (6ª Edição) de Kurose e Ross (2013) é tratado um tópico chamado *Wireshark Lab,* onde este aborda uma ferramenta de *scanner* de rede.

Scanners de redes são programas utilizados para varrer os computadores em uma rede à procura de vulnerabilidade e/ou portas abertas, que possam receber possíveis ataques, podendo ocorrer tanto em servidores como em *hosts* finais. Eles buscam sistemas desprotegidos e trabalham de duas formas: (1) os *scanners* de portas (*portscanner*), que verificam as portas TCP/IP abertas de um sistema e os de (2) vulnerabilidades, que fazem a verificação das vulnerabilidades conhecidas nos programas que rodam no computador. Estes fazem testes necessários e com os resultados o administrador pode procurar melhorias no sistema.

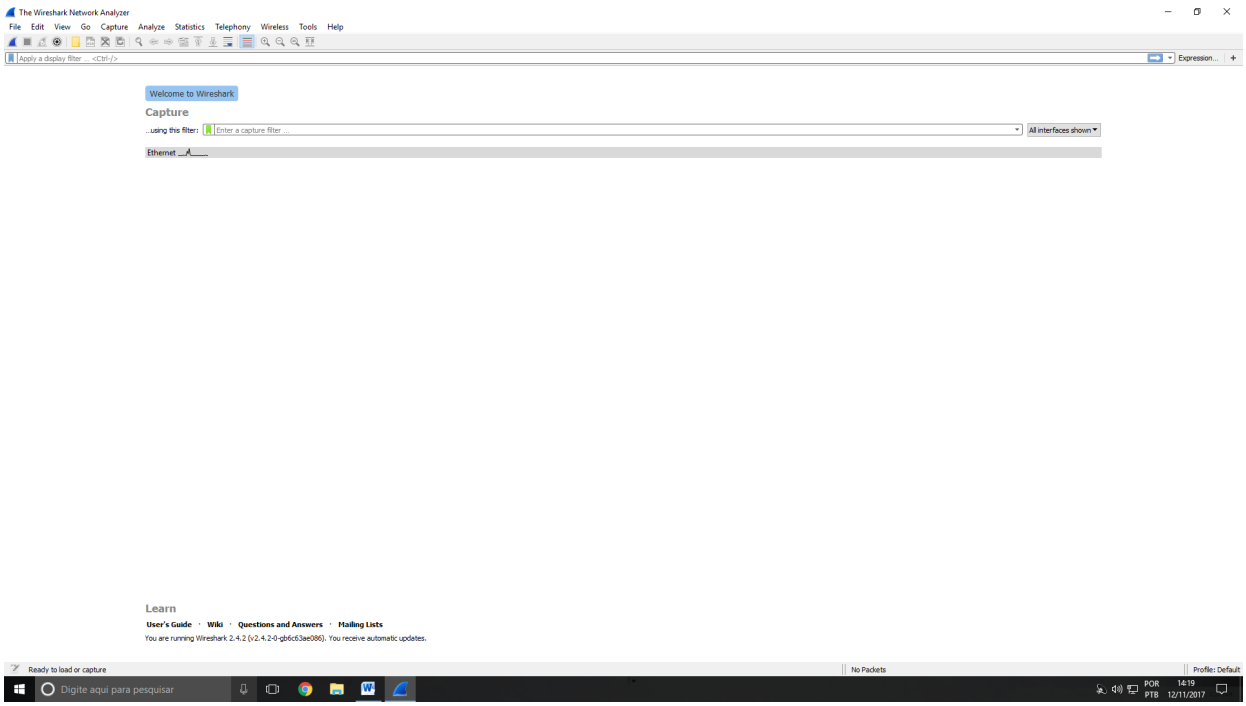
Como citado acima, estas ferramentas não são apenas para uso a fins de invadir e roubar dados pessoais, mas também tem o intuito de testar as vulnerabilidades de um programa, dando *feedbacks* necessários para fazer uma renovação na segurança de determinados sistemas.

Uma ferramenta muito utilizada no mundo *Linux* que também é citada no livro de Kurose e Ross (2013) é o *Wireshark.* Uma ferramenta muito poderosa e

utilizada mundialmente. Esta ferramenta permite verificar minuciosamente o que esta acontecendo em uma rede a nível microscópico de detalhes. Hoje em dia utilizam-se *Switches* e não mais *Hubs* dificultando um pouco mais a captura de dados (pacotes) em uma rede, já que os *Switches* não enviam dados para todas as portas (*broadcast*) como o *Hub* fazia.

Abaixo na Figura 1, mostra-se a tela inicial da ferramenta *Wireshark*. No capítulo 2 será abordado o seu funcionamento, terminando com uma conclusão no capítulo 3.

FIGURA 1 – Tela inicial da ferramenta *Wireshark*.



Fonte: Autor (2017).

2 – EMBASAMENTO TEÓRICO

A ferramenta *Wireshark* possui um rico conjunto de recursos como:

- Inspeção profunda de centenas de protocolos.
- Analise de captura ao vivo e off-line.
- Navegador padrão de pacotes de três painéis.
- Multi-plataforma: funciona no *Windows*, *Linux*, *MacOS*, *Solaris*, *FreeBSD*, *NetBSD* entre outros.
- Os filtros de exibição mais poderosos da indústria.
- Os dados ao vivo podem ser lidos de *Ethernet*, IEEE 802.11, PPP / HDLC, ATM, *Bluetooth*, USB, *Token Ring*, *Frame Relay*, FDDI e outros (dependendo da sua plataforma).
- Suporte de encriptação para muitos protocolos.
- Coloração pode ser aplicada na lista de pacotes para uma análise rápida e intuitiva.
- A saída pode ser exportada para XML, *PostScript*, CSV ou texto sem formatação.

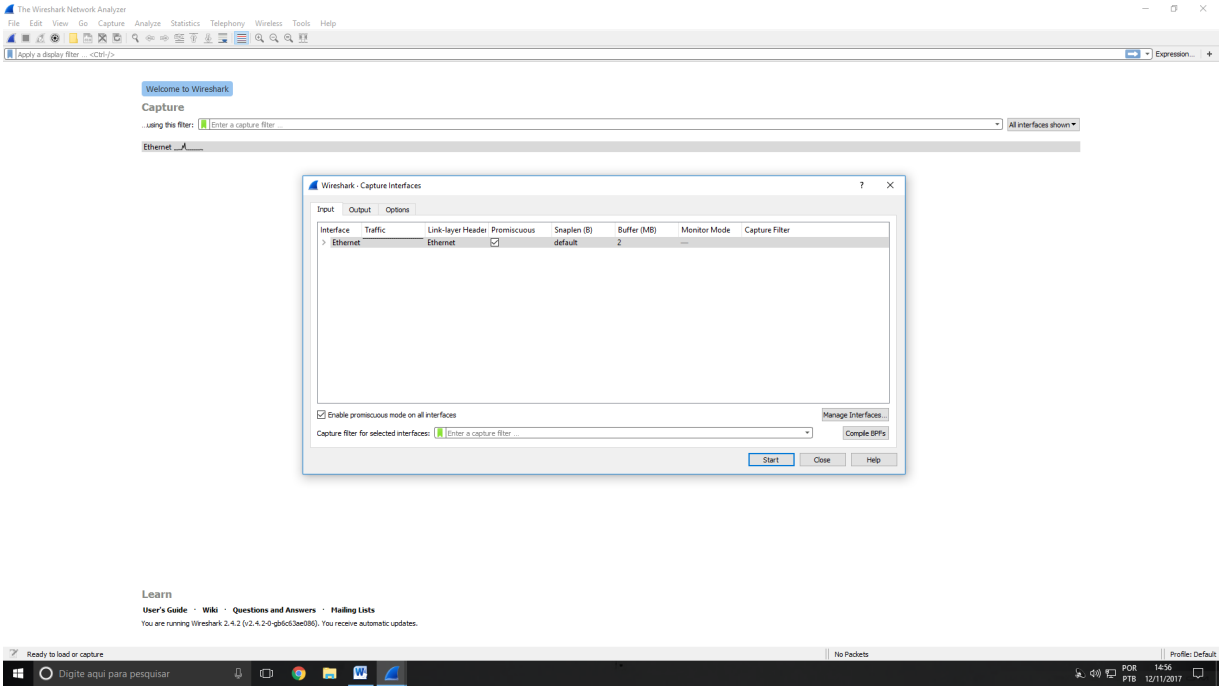
Algumas destas características poderão ser vistas nos tópicos abaixo.

2.1 – Funcionamento da ferramenta.

Após realizada a instalação da ferramenta, pode-se testa-la capturando alguns pacotes em uma rede em tempo real. A pesquisa mostra como a ferramenta faz a separação das camadas, diferenciação dos protocolos e o seu poder nas mãos de usuários mal-intencionados.

Para fazer uma varredura em uma rede com a ferramenta *Wireshark,* é necessário primeiramente escolher a interface de rede na qual se deseja iniciar o procedimento (encontra-se no menu superior, *Capture* e depois *Options*), como mostra a Figura 2. Após selecionar a(s) interface(s) (o computador pode ter mais de uma, exemplo, interface cabeada e uma placa *Wireless*), pressione o botão *Start* que o procedimento ira começar.

FIGURA 2 – Tela de escolha de interface(s).

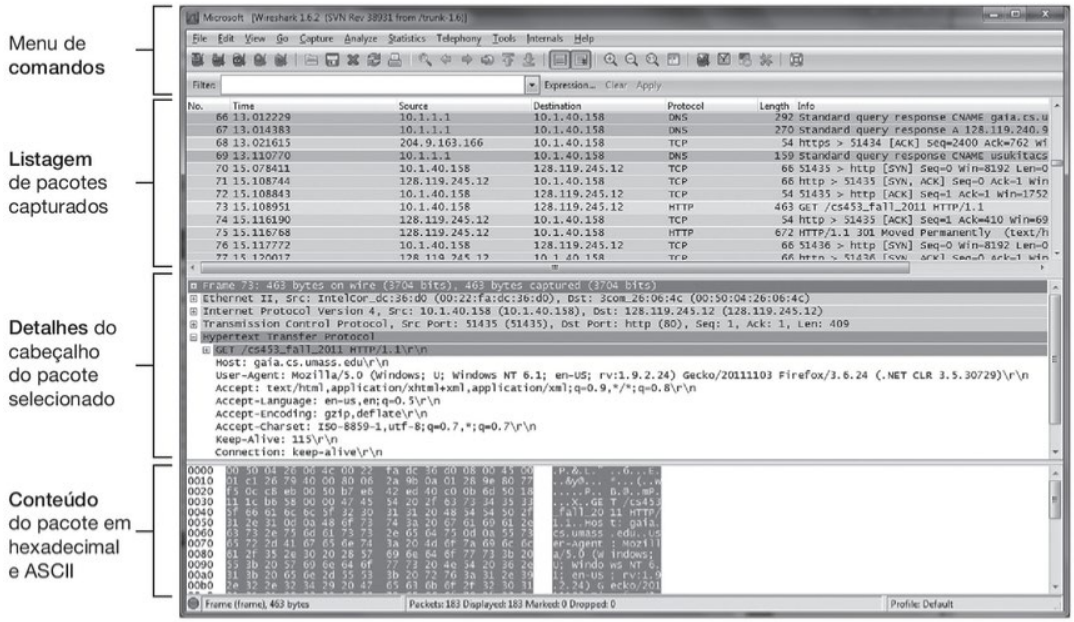


Fonte: Autor (2017).

Assim que a ferramenta começar a “farejar” os pacotes em sua rede, a seguinte tela irá aparecer como mostra a Figura 3. Kurose e Ross (2013) explica que a ferramenta básica para observar as mensagens trocadas entre entidades de protocolos em execução é chamada *de packet sniffer* (analisador de pacotes). Ela simplesmente “fareja” passivamente mensagens enviadas e recebidas por um computador e também exibe o conteúdo de vários campos de protocolos das mensagens capturadas.

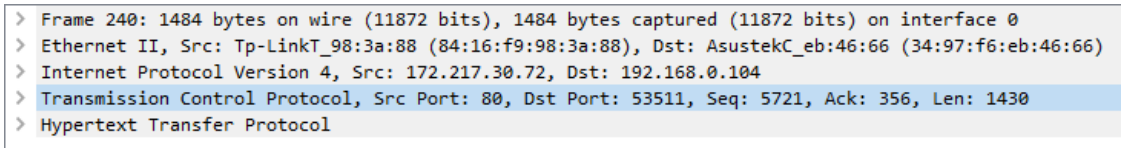
Na parte de “listagem de pacotes capturados”, estão todos os pacotes que foram capturados durante a varredura. Ela traz uma tabela com os campos de *Time* (tempo), *Source* (origem), *Destination* (destino), *Protocol* (protocolo), *lenght* (tamanho) e *info* (informação), onde pode-se observar o IP de origem e destino, o protocolo que esta sendo usado (TCP, UDP, HTTP...), o tamanho e informações básicas. Assim que selecionada uma linha desta tabela, pode-se observar mais informações sobre este pacote através do menu “detalhes do cabeçalho do pacote selecionado” como mostra a Figura 4.

FIGURA 3 – Tela do analisador de pacotes.



Fonte: Livro do Kurose | Ross, Redes de computadores e a internet: uma abordagem top-down (6.ed, pág. 58, 2013).

FIGURA 4 – Menu de exibição do cabeçalho do pacote escolhido.



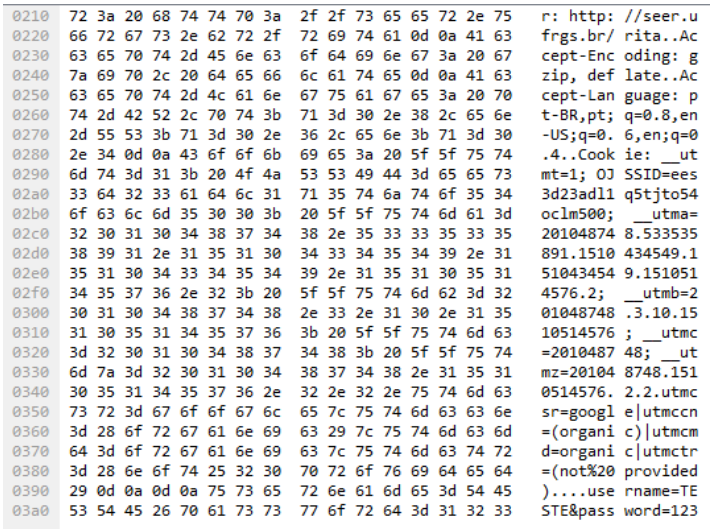
Fonte: Autor (2017).

Aqui mostra com detalhes minuciosos e microscópicos todas as informações que estão contidas dentro de um determinado pacote, os protocolos que ele utilizou, seu IP de origem e destino, endereço MAC, portas e muito mais. Esta parte da ferramenta traz as informações separadas em camadas como mostra a Figura 4, desde a camada de aplicação (*Hypertext Transfer Protocol* - HTTP) até a camada de enlace (na Figura 4, escrito *Frame*), passando pela camada de transporte e rede. Nos próximos tópicos a pesquisa vai abordar com mais detalhes a captura e extração de informação de um determinado pacote e como a segurança em redes é importante quando o assunto são dados pessoais dos usuários. O tópico abaixo traz um pequeno exemplo de como capturar uma senha com a ferramenta *Wireshark*.

2.2 – Capturando senhas.

Percebemos na Figura 3, uma área denominada “conteúdo de um pacote em hexadecimal e ASCII”. Nesta parte se traz o conteúdo que esta dentro de um determinado pacote escolhido, como o protocolo da camada de aplicação que ele utiliza, seu HTML e TAGS, site acessado, cookies e alguns valores um pouco bagunçados como podemos ver na Figura 5.

FIGURA 5 – Conteúdo de um pacote em hexadecimal.

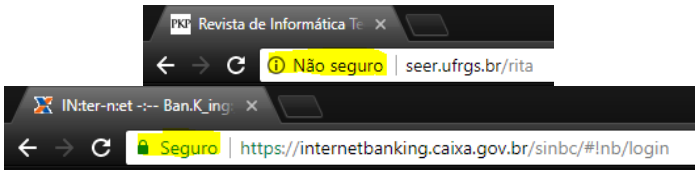


```
0210  72 3a 20 68 74 74 70 3a  2f 2f 73 65 65 72 2e 75   r: http: //seer.u
0220  66 72 67 73 2e 62 72 2f  72 69 74 61 0d 0a 41 63   frgs.br/ rita..Ac
0230  63 65 70 74 2d 45 6e 63  6f 64 69 6e 67 3a 20 67   cept-Enc oding: g
0240  7a 69 70 2c 20 64 65 66  6c 61 74 65 0d 0a 41 63   zip, def late..Ac
0250  63 65 70 74 2d 4c 61 6e  67 75 61 67 65 3a 20 70   cept-Lan guage: p
0260  74 2d 42 52 2c 70 74 3b  71 3d 30 2e 38 2c 65 6e   t-BR,pt; q=0.8,en
0270  2d 55 53 3b 71 3d 30 2e  36 2c 65 6e 3b 71 3d 30   -US;q=0. 6,en;q=0
0280  2e 34 0d 0a 43 6f 6f 6b  69 65 3a 20 5f 5f 75 74   .4..Cook ie: __ut
0290  6d 74 3d 31 3b 20 4f 4a  53 53 49 44 3d 65 65 73   mt=1; OJ SSID=ees
02a0  33 64 32 33 61 64 6c 31  71 35 74 6a 74 6f 35 34   3d23adl1 q5tjto54
02b0  6f 63 6c 6d 35 30 30 3b  20 5f 5f 75 74 6d 61 3d   oclm500;  __utma=
02c0  32 30 31 30 34 38 37 34  38 2e 35 33 33 35 33 35   20104874 8.533535
02d0  38 39 31 2e 31 35 31 30  34 33 34 35 34 39 2e 31   891.1510 434549.1
02e0  35 31 30 34 33 34 35 34  39 2e 31 35 31 30 35 31   51043454 9.151051
02f0  34 35 37 36 2e 32 3b 20  5f 5f 75 74 6d 62 3d 32   4576.2;  __utmb=2
0300  30 31 30 34 38 37 34 38  2e 33 2e 31 30 2e 31 35   01048748 .3.10.15
0310  31 30 35 31 34 35 37 36  3b 20 5f 5f 75 74 6d 63   10514576 ; __utmc
0320  3d 32 30 31 30 34 38 37  34 38 3b 20 5f 5f 75 74   =2010487 48; __ut
0330  6d 7a 3d 32 30 31 30 34  38 37 34 38 2e 31 35 31   mz=20104 8748.151
0340  30 35 31 34 35 37 36 2e  32 2e 32 2e 75 74 6d 63   0514576. 2.2.utmc
0350  73 72 3d 67 6f 6f 67 6c  65 7c 75 74 6d 63 63 6e   sr=googl e|utmccn
0360  3d 28 6f 72 67 61 6e 69  63 29 7c 75 74 6d 63 6d   =(organi c)|utmcm
0370  64 3d 6f 72 67 61 6e 69  63 7c 75 74 6d 63 74 72   d=organi c|utmctr
0380  3d 28 6e 6f 74 25 32 30  70 72 6f 76 69 64 65 64   =(not%20 provided
0390  29 0d 0a 0d 0a 75 73 65  72 6e 61 6d 65 3d 54 45   ).....use rname=TE
03a0  53 54 45 26 70 61 73 73  77 6f 72 64 3d 31 32 33   STE&pass word=123
```

Fonte: Autor (2017).

No meio desta “bagunça” que se obtêm informações cruciais. Foram acessados dois sites para uma análise destes dados que são trazidos pela ferramenta. O primeiro, um site qualquer de uma revista de informática que não contem segurança de acesso. O site utiliza protocolo HTTP simples, sem criptografia dos dados. E o segundo, o site do *Internet Banking* da Caixa que contém protocolo HTTPS, certificados digitais, segurança com encriptação dos dados, entre outras técnicas, como mostra a Figura 6.

FIGURA 6 – Sites acessados, com e sem segurança de encriptação dos dados.

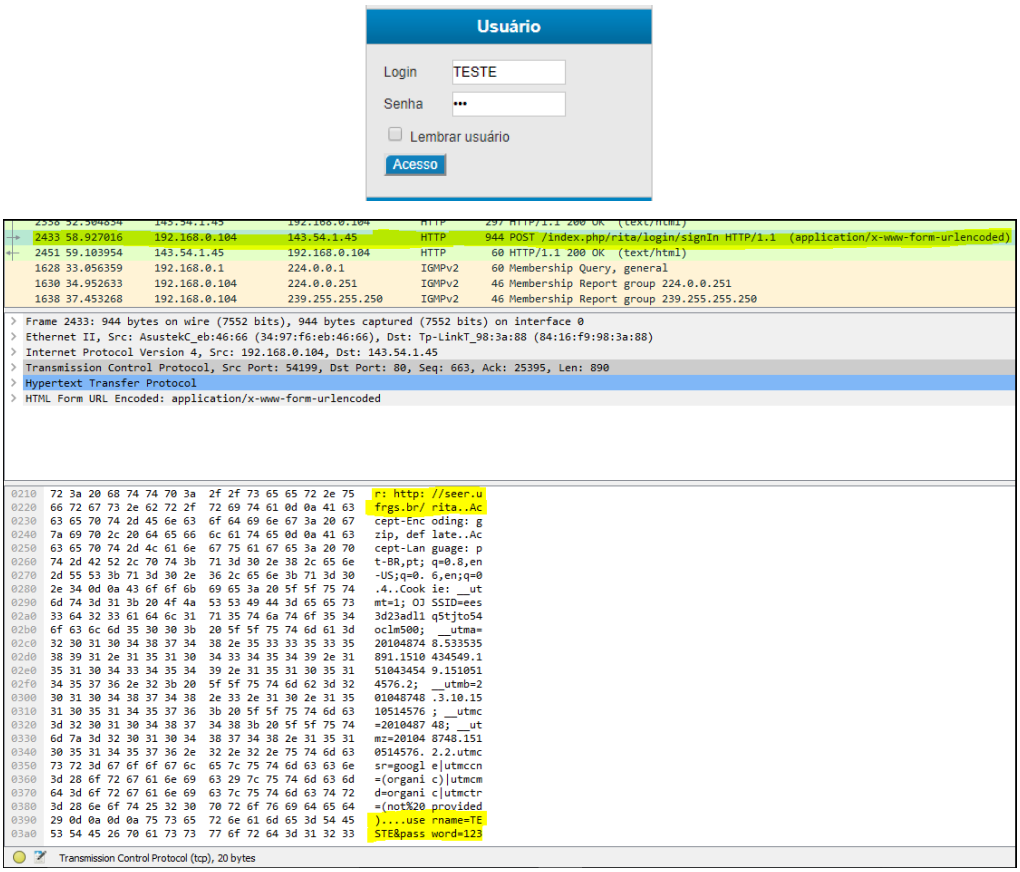


Fonte: Autor (2017).

Os sites que não contem encriptação de dados transferem seus dados em um arquivo de texto normal. Estes quando capturados podem ser facilmente abertos e lidos por qualquer um. Neles contem informação como o site que foi acessado e se tentar logar em uma conta, aparecerá o *login* e senha do usuário, como mostra a Figura 7, com um arquivo de dados capturado através da pesquisa feita no primeiro site da revista de informática.

Após a tentativa de logar com um usuário chamado **TESTE** e com a senha **123**, podemos analisar que estes aparecem de forma explicita no dado capturado pela ferramenta *Wireshark,* enquanto estamos efetuando uma varredura na rede, e utilizando o computador para acessar os sites qualquer.

FIGURA 7 – *Login* e senha capturados.



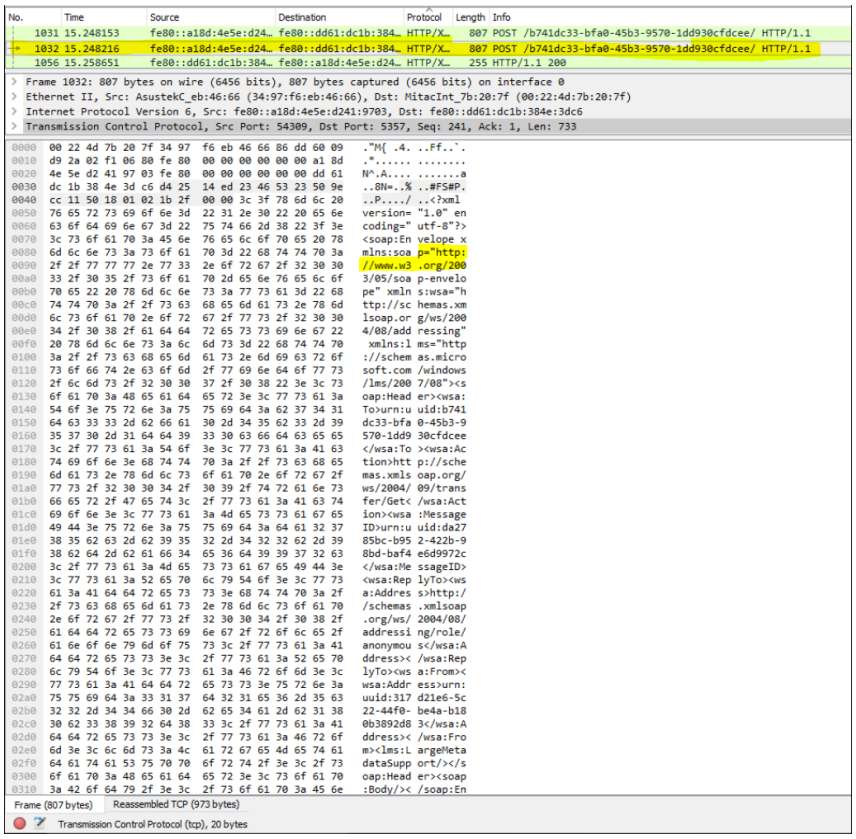
Fonte: Autor (2017).

Selecionada a linha da tabela do protocolo HTTP (da camada de aplicação) do tipo POST, tipo que envia dados para um determinado formulário gerando uma requisição, no caso da Figura 7, a tentativa de logar no site da revista de informática. Observando a imagem percebe-se facilmente os respectivos *login* e senha, marcados de marca texto na tela do *Wireshark*. Porém isto já não ocorre no

segundo exemplo de tentativa de acesso ao *Internet Banking* da Caixa, como mostrado na Figura 6. O site da Caixa oferece o protocolo na camada de aplicação chamado de HTTPS que tem encriptação de dados e certificados digitais, para prevenir contra este tipo de problema de roubos de dados pessoais.

Observando a Figura 8, percebe-se que com os dados criptografados a leitura e extração de informações não são tão simples, chegando a ser impossível. Mesmo que estes dados caiam nas mãos de uma potente máquina, é quase impossível (se não for impossível) descriptografar estes dados. Pegando o exemplo dos bancos, eles têm criptografias próprias que só a base de dados deles entende, com algoritmos poderosos e específicos, tornando as redes mais seguras.

FIGURA 8 – Tentativa de leitura dos dados do site da Caixa – *Internet Banking*.



Fonte: Autor (2017).

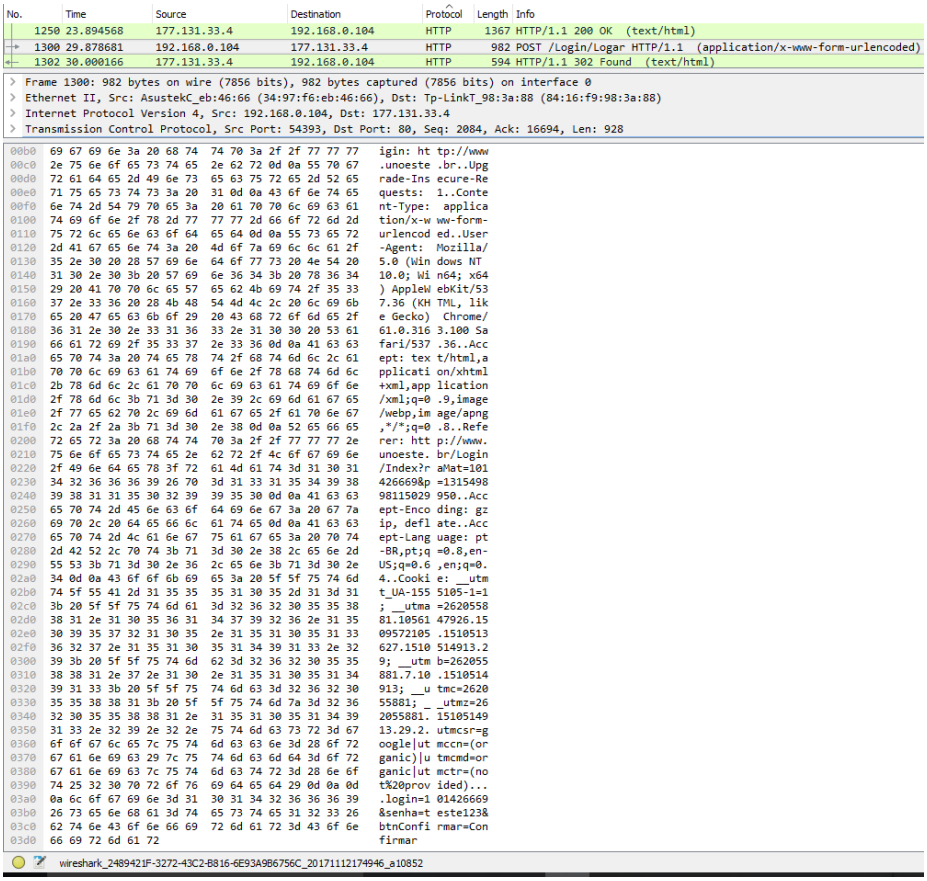
Após observar estes dois exemplos das falhas de segurança em determinados sites, um terceiro exemplo foi escolhido para explicar melhor como a ferramenta *Wireshark* trabalha sobre as camadas do modelo TCP/IP (o mais usado mundialmente) e como ela trata e exibe as informações que são trazidas dentro dos

pacotes. O próximo tópico mostrará isto na tentativa de acesso ao Aprender Unoeste no site oficial da Unoeste.

2.3 – Detalhes dos cabeçalhos de dados entre camadas.

Após acessar o site www.unoeste.br e logar com um usuário e senha na área do aluno (Aprender Unoeste) obtivemos os seguintes dados mostrados na Figura 9.

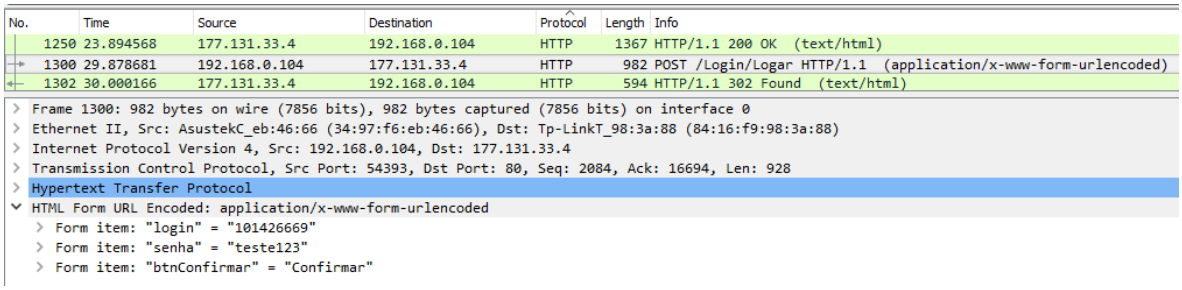
FIGURA 9 – Cabeçalho de dados do site da Unoeste.



Fonte: Autor (2017).

A primeira coisa que podemos observar é que este site não contém segurança como já vimos no tópico “2.2 – Capturando senhas”, facilmente podemos observar o campo *login* com o valor 101426669 e o campo senha com o valor teste123. Isto também pode ser observado na Figura 10, localizada nos “detalhes do cabeçalho do pacote selecionado” na aba *HTML Form URL Encoded*.

FIGURA 10 – Detalhes do pacote.



Fonte: Autor (2017).

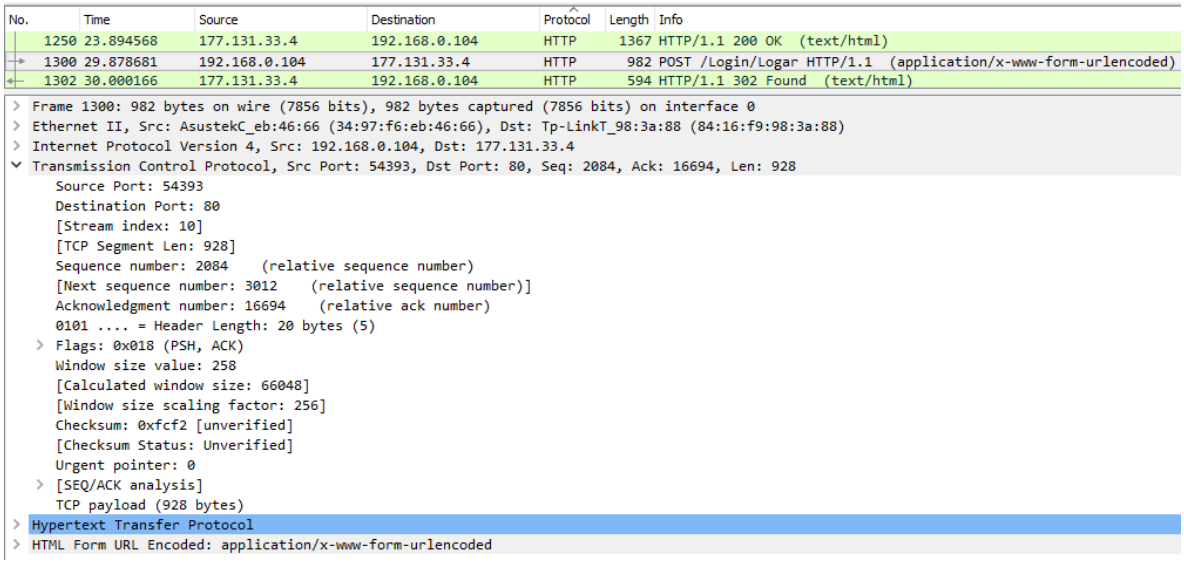
A próxima camada a ser analisada, é a camada de aplicação na aba *Hypertext Transfer Protocol* mostrado na Figura 11. Esta aba traz algumas informações como o protocolo utilizado (no caso da Figura 11 o HTTP), o tipo (POST ou GET (entre outros)), algumas *tags* HTML, o site acessado, a sua linguagem e o tamanho do dado em *bytes*.

FIGURA 11 - *Hypertext Transfer Protocol.*

Fonte: Autor (2017).

A Figura 12 mostra a camada de transporte. Além de especificar o protocolo utilizado (o TCP como já mostra no nome da aba *Transmission Control Protocol*) ele especifica as portas de origem e destino, seu ACK e tamanho, *checksum* e *flags* além de outras informações complementares.

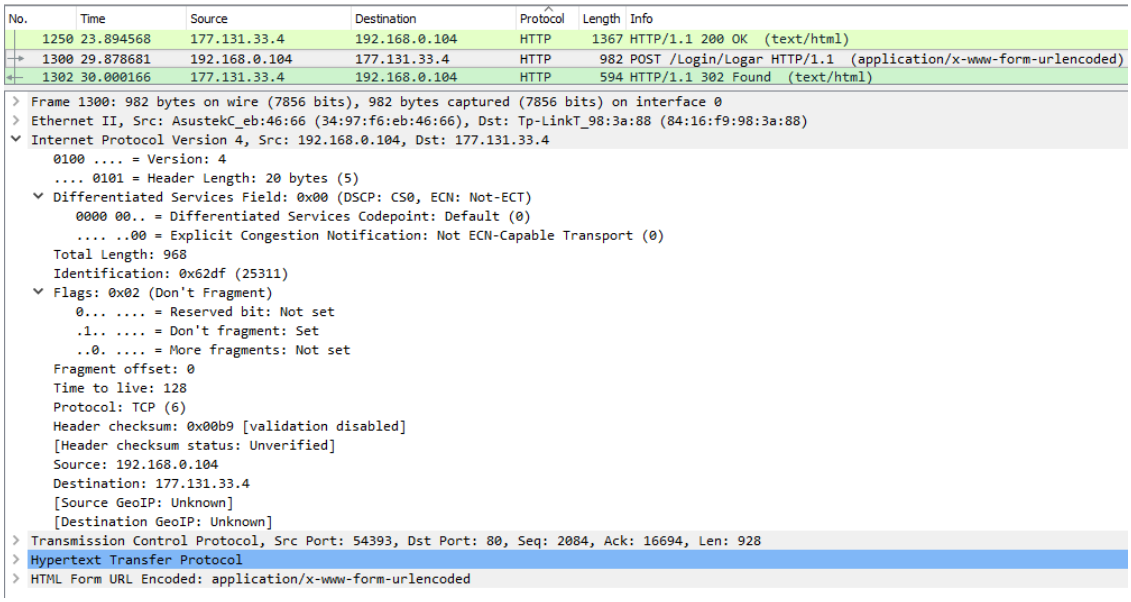
FIGURA 12 – Aba *Transmission Control Protocol*.



Fonte: Autor (2017).

A aba *Internet Protocol Version* 4, referente a camada de rede, podemos facilmente ver o protocolo utilizado o IPV4, o IP de origem (minha máquina) e destino (Servidor da Unoeste) da requisição, algumas *flags*, *offset* e os respectivos tamanhos dos dados em *bytes* exibidos na Figura 13.

FIGURA 13 – Dados da camada de rede.

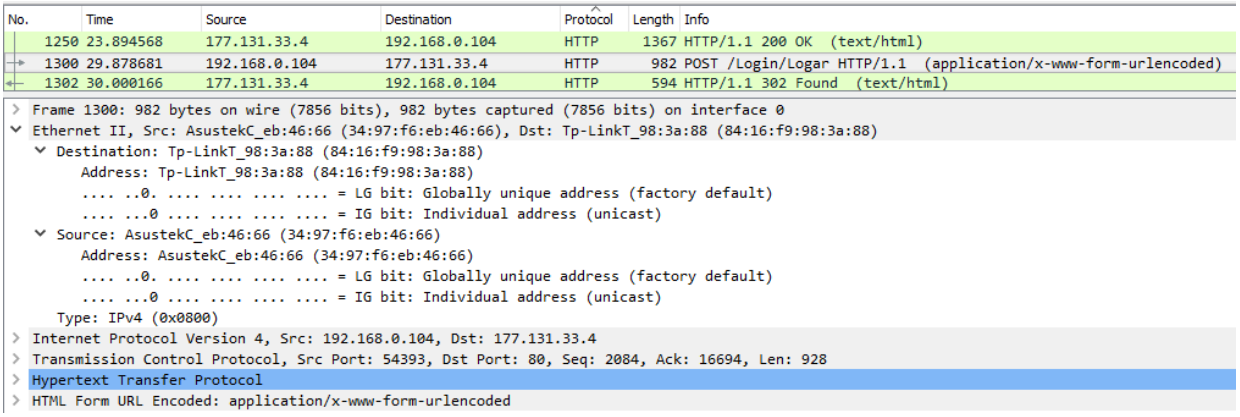


Fonte: Autor (2017).

Por ultimo a camada de enlace na aba denominada de *Ethernet II,* onde pode se notar os endereços MAC das máquinas e seus respectivos roteadores /

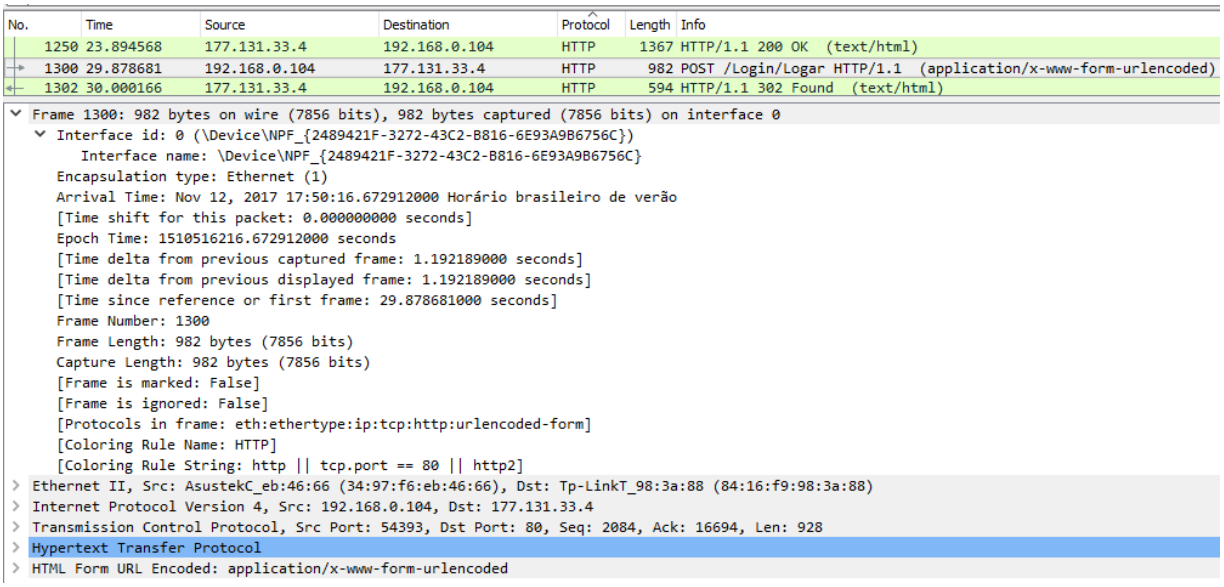
comutadores, tanto de origem quanto destino todos em hexadecimal como mostra a Figura 14. E na aba *Frame* como mostra a Figura 15 são exibidas as interfaces, tamanho dos dados e tempos, alguns protocolos e também portas utilizadas podendo fazer uma analogia com a camada física.

FIGURA 14 – Detalhes dos MAC.



Fonte: Autor (2017).

FIGURA 15 – Detalhes dos *frames.*



Fonte: Autor (2017).

3 – CONCLUSÃO

Esta pesquisa abordou um pouco sobre a ferramenta de escaneamento de redes *Wireshark,* trazendo alguns exemplos de sites sem segurança e como a ferramenta tratou isto. Em meio ao mundo quase totalmente digital, e com um fluxo de informação exorbitante passando todos os dias nas redes de computadores espalhadas pelo mundo, vimos que a segurança é algo indispensável, pois os dados certos nas mãos de pessoas erradas causam um grande estrago.

Utilizando a ferramenta observamos o poder de detalhamento que ela traz dentro dela, muito bem organizada, com o mínimo de falhas possível e muito fácil de utilizar. A pesquisa abordou alguns tópicos simples, mas foi o suficiente para se observar o quão completo é este programa.

REFERÊNCIAS

KUROSE, J. F; ROSS, K. W. **Redes de computadores e a internet**: uma abordagem top-down.6.ed. São Paulo: Pearson Education do Brasil, 2013.

TANENBAUM, A. S. **Redes de computadores**. 4.ed. Rio de Janeiro: Elsevier Editora, 2003.

DIEGO MACÊDO – ANALISTA DE T.I. **Introdução ao Wireshark**: Detecção e captura de tráfego em redes. 2016. Disponível em <http://www.diegomacedo.com.br/introducao-ao-wireshark-deteccao-e-captura-de-trafego-em-redes/>. Acessado em: 11 Nov. 2017.

LOGANDO TI. **Scanners de rede (software)**: Portas e vulnerabilidades. 2011. Disponível em: <http://www.logandoti.com/scanners-de-rede-software-portas-e-vulnerabilidades/>. Acessado em: 11 Nov. 2017.

UNDER-LINUX.ORG. **Wireshark – Parte 1 – Análise de Tráfego e Captura de Senhas**. 2012. Disponível em: < https://under-linux.org/entry.php?b=2969>. Acessado em: 11 Nov. 2017.

WIRESHARK. **Wireshark**. 2017. Disponível em: <https://www.wireshark.org/>. Acessado em: 11 Nov. 2017.