Universidade Católica de Brasília

Pró-Reitora de Pós-Graduação e Pesquisa
Lato Sensu em Perícia Digital
Trabalho de Conclusão de Curso

ANÁLISE FORENSE EM TRÁFEGO DE REDES

Autor: Marcos Godinho Velozo
Orientador: Prof. Esp. João Eriberto Mota Filho

Brasília – DF
2012

MARCOS GODINHO VELOZO

ANÁLISE FORENSE EM TRÁFEGO DE REDES

Artigo apresentado ao curso de pós-graduação e pesquisa Lato Sensu em Perícia Digital da Universidade Católica de Brasília, como requisito parcial para obtenção do Título de Especialista em Perícia Digital.

Orientador: Profº Esp. João Eriberto Mota Filho

Brasília
2012

Artigo de autoria de Marcos Godinho Velozo, intitulado "ANÁLISE FORENSE EM TRÁFEGO DE REDES, apresentado como requisito parcial para obtenção do grau de Especialista em Perícia Digital da Universidade Católica de Brasília, em xx/xx/2012, defendido e aprovado pela banca examinadora abaixo assinada:

_____

Profº. Esp. João Eriberto Mota Filho
Orientador
Perícia Digital – UCB

_____

Profº. Msc. Paulo Roberto Corrêa Leão
Membro da Banca
Perícia Digital – UCB

Brasília

ANÁLISE FORENSE EM TRÁFEGO DE REDES

MARCOS GODINHO VELOZO

Resumo:
Com a constante progressão da tecnologia das redes de computadores foi propiciada uma mudança nos padrões para área de computação no qual a informação que antes era reduzida passou a ser comum pelo mundo. As redes estão cada vez maiores, complexas e interconectadas por intermédio da Internet. A facilidade da interconexão mundial favoreceu o acréscimo da complexidade quanto à administração e gerenciamento destas redes. Além de grandes em número de equipamentos, os dados trafegados e a exigência de disponibilidade também acompanharam o crescimento, tornando ainda mais difícil a tarefa de gerenciamento destes ambientes. Ferramentas de apoio ao gerenciamento de redes, assim como a base destas ferramentas (suas bibliotecas), têm papel importante em se tratando das atividades relacionadas à disponibilidade e segurança das redes. Muitas dessas ferramentas de apoio fazem uso de captura de dados como forma de obter informações da rede e, em muitos casos, essa captura é feita através da libpcap, biblioteca que implementa captura passiva de tráfego. Foi conduzida a análise das ferramentas para análise forense em rede, tendo alcançado o objetivo de identificar os fatores que influenciam no desempenho dessas ferramentas, além de chegar a um modelo de previsão em função da combinação dos fatores analisados.
O escopo deste trabalho é descrever as ferramentas que fazem análise forense em tráfego de rede de forma eficiente. Essas informações foram coletadas em livros, revistas especializadas, sites relacionados com o tema e outros trabalhos com conteúdos similares. Além de um estudo de caso com a aplicação prática de uma dessas ferramentas que fazem análise forense em tráfego de redes e o impacto disto na perícia forense.

Palavras-chave: Redes de Computadores. Análise forense em rede. Ferramentas

1 INTRODUÇÃO

A Internet por sua natureza flexível e ágil é um meio que favorece a prática de vários crimes, cuja materialização é viabilizada pela transferência de dados de arquivos sob os mais diversos subterfúgios. A captura e o reconhecimento destes dados são primordiais em processos preventivos ou investigativos que não raramente encontram obstáculos de natureza técnica. O combate a estes crimes é ainda dificultado pela natureza inconstante dos dados ao transitarem pela rede assim como a constante mudança destes dados. Desta forma, uma vez estabelecidos os pontos, a captura do tráfego de rede sob suspeita deve ser feita no menor tempo possível, assim como a posterior extração e o reconhecimento dos objetos e/ou suas alterações. Captura, extração e seleção, estas três etapas devem ser realizadas preferencialmente de maneira simples e transparente para o analista ou investigador.

Muitas aplicações de apoio ao gerenciamento e segurança de redes empregam especificamente a captura passiva de tráfego, fazendo uso de uma biblioteca de software chamada *libpcap*. Esta é uma biblioteca open source portável e que provê funcionalidades

para captura de tráfego das interfaces de rede sem que os desenvolvedores de aplicações precisem implementá-las.

Este formato de arquivo é um formato muito básico para salvar dados capturados de rede. Como a biblioteca libpcap tornou-se de fato padrão de captura de dados de rede no UN * X, (*Unix-like*, por vezes referido como UN * X ou * nix) então se tem um "denominador comum" para arquivos de captura de rede dentro do open source mundial.

Libpcap, e a porta do Windows na *libpcap*, *WinPcap*, usam o mesmo formato de arquivo. Não obstante às vezes é colocado que este formato é adequado somente para redes Ethernet, mas ele pode servir a vários tipos de rede diferentes, exemplos podem ser encontrados na página de suporte de captura do *Wireshark*, onde todos os tipos listados são tratados pelo formato de arquivo Libpcap.

A extensão do arquivo proposto para arquivos *libpcap* é PCAP. A análise forense de rede possui algumas condições, pré- requisitos para fazer sentido. É necessário o tráfego de rede. Basicamente precisa se de um dump de rede já capturado. Quando se fala de análise de rede em modo geral, existe o conceito da volatilidade. As informações são voláteis e cada tipo de informação tem um tipo de volatilidade diferente. Existe o conceito de ordem de volatilidade que pode ser pensado da seguinte forma: quando vai se extrair os dados, necessita–se respeitar essa ordem para minimizar perdas. Perdas essas no sentido de que quando existe a extração de informação de um computador, enquanto extrai de um lado polui do outro. Por exemplo: enquanto alguém está listando um arquivo, também está sobrescrevendo algum dado, nem que sejam instruções no processador etc. Então quando se segue a ordem de volatilidade minimizam-se as perdas. Em se tratando de ordem de volatilidade deve-se considerar coletar primeiramente os dados que forem mais efêmeros.

Os pacotes que passam na rede são extremamente voláteis. São voláteis também dados em memória, cache de processador etc. Então para se fazer a análise forense de rede é importante que a captura esteja sendo feita previamente, por que se precisa desses dados capturados. Então o que será visto será exatamente a análise desses *dumps*. Com um dump de rede é possível analisar todos esses pacotes, seguindo assim o objetivo geral da forense que é reproduzir o que aconteceu, reproduzir o passo a passo. Dessa forma poderemos saber, por exemplo, se foi um ataque ou qualquer coisa do gênero. A idéia é através de um dump de rede conseguir entender o que aconteceu e principalmente daquele dump extrair o que é informação, o que é evidência e o que evidencia um determinado incidente.

A captura e análise de pacotes envolvem conhecimentos TCP, modelo OSI e coisas afins. Quando se comenta em reprodução da sessão capturada, logo se imaginam que o TCP tem facilidade de recuperar sessões, conversas entre cliente e servidor. Algo interessante é o data *carving* em rede, que é a reconstrução de arquivos. Detecta-se que tem arquivo sendo transferido. Exemplos: Anexos de email, FTP reconhece e recupera arquivo original.

Alguns pontos que são considerados tráfegos anômalos que justificam uma investigação forense são:

a) Portas suspeitas - Portas conhecidas de *worm* se propagando, portas conhecidas de determinados ataques, que não são serviços legítimos na sua rede, já podem ser encaradas como um tráfego suspeito ou no mínimo anormal.
b) É importante estar atento aos pacotes contendo comandos e saídas de comandos. Comandos normais de *shell* e de qualquer outra coisa. Saída de comando é

interessante filtrar também, porque é comum filtros bloquearem comandos. Na verdade o filtro deveria bloquear tudo isso, mas é comum ataques desse tipo mascararem o comando para passar os filtros. Então se detecta qualquer tráfego *outbound* que venha com saídas de comandos. Então se pode bloquear, porque já é considerado tráfego suspeito. Alguma instrução chegou de fora que não era para ter chegado.

c) Os Códigos NOPs (No *operation*) são códigos que não fazem nada, são utilizados na anatomia de um ataque de *bufferoverflow* por exemplo. Então quando se detecta Nops passando, pode ser um *exploit*. Um *exploit* sendo utilizado para explorar alguma vulnerabilidade. Então isso também entra nos filtros.
d) *Flood – Flood* de uma maneira geral são suspeitos, *flood* ou está envolvido com força bruta ou DoS ou algo do gênero. *Flood* é um aumento significativo no número de pacotes, com intenção de inundar um determinado ativo de rede para um serviço específico ou para uma máquina específica.

1.1 JUSTIFICATIVA

Os estudos serão demonstrados utilizando algumas ferramentas e suas características tanto para Linux quanto para Windows na análise forense em tráfego de rede. As capturas foram extraídas do site Wireshark.

1.2 DELIMITAÇÃO DO TEMA

A ferramenta é um utilitário que facilita o trabalho, portanto serão demonstradas algumas ferramentas que podem se usadas para análise forense em tráfego de rede.

1.3 PROBLEMA

1.3.1 Contextualização do Problema

Chaudet, Fleury e Rivano (2005, p.71, tradução) afirmam que: “O gerenciamento das redes se torna essencial para avaliação e melhoria de performance e identificação de problemas localizados.” Portanto uma possível melhora em performance e identificação de problemas está relacionado com a segurança da rede que também depende diretamente do modelo de gerência utilizado.

1.3.2 Enunciado do Problema

Ferramentas para análise forense em tráfego de rede podem auxiliar na realização de perícias forenses digitais na coleta e exames de dados em busca da identificação, análise e compreensão das ações realizadas por usuários ou eventos ocorridos em uma rede de computadores?

1.4 HIPÓTESE DA PESQUISA

A imprescindível compreensão e identificação de ações feitas em específicas redes de computadores produz a necessidade de uma preparação de um estudo de caso no decurso de uma pesquisa laboratorial de forma a averiguar a eficácia das ferramentas para perícia computacional em tráfego de rede, provando de forma qualitativa suas características.

1.5 PROPÓSITOS

1.5.1 Objetivo Geral

O objetivo deste trabalho foi realizar uma análise experimental, verificando o desempenho das ferramentas para perícia computacional em tráfego de rede usando a captura de pacotes libpcap aplicando a metodologia de projeto de experimentos, contribuindo para outros projetos que fazem uso dessa biblioteca.

1.5.2 Objetivos Específicos

Os objetivos específicos deste trabalho foram:

a) montagem em um ambiente de testes baseado em Linux e windows para execução de testes baseados na metodologia de projeto de experimentos;
b) identificação dos fatores com maior influência no desempenho das ferramentas para perícia computacional em tráfego de rede para os cenários avaliados;
c) Demonstrar vantagens e desafios do uso das ferramentas para perícia computacional em tráfego de rede;
d) Documentação dos resultados e as análises sobre o experimento realizado.

2 DESENVOLVIMENTO

2.1 REFERENCIAL TEÓRICO

A seguir apresentam-se os conceitos referentes a perícia digital , análise forense de rede, computação forense e ferramentas para perícia computacional em tráfego de rede.

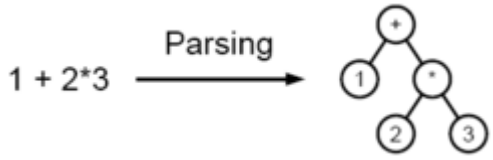
2.1.1 Tcpdump

O Tcpdump é uma ferramenta, criada pela tcpdump.org, utilizada para monitorar os pacotes trafegados em uma rede de computadores. Essa ferramenta mostra os cabeçalhos dos pacotes que passam pela interface de rede. O T*cpdump* é estável. Isso é importante em uma ferramenta de captura porque, para colher dados em uma rede, é necessário, dentre outras coisas, ativar o modo promíscuo. Isto significa colher todo tráfego que pode ser visto, mesmo que este não seja destinado à máquina que está escutando tal rede. Para isso, é necessário que o programa seja executado como root ou como administrador. Portanto é extremamente

complicado ter uma ferramenta que trata pacotes em tempo real, de “n” protocolos diferentes. Isso pode dar margem a vulnerabilidades de parsing. Em ciência da computação e linguística, análise sintática (também conhecida pelo termo em inglês parsing) é o processo de analisar uma sequência de entrada (lida de um arquivo de computador ou do teclado, por exemplo) para determinar sua estrutura gramatical segundo uma determinada gramática formal. Essa análise faz parte de um compilador, junto com análise léxica e análise semântica. (Fonte: pt.wikipedia.org)

A figura 01 mostra o exemplo da análise sintática de uma expressão matemática. O resultado é uma árvore da expressão.

Figura 01 - Árvore da expressão



Fonte: Wikipédia

Pode-se criar um pacote malicioso, que quando determinada ferramenta fizer o *parsing* desse pacote ela apresentará problemas. É o que acontece com o *wireshark* por exemplo. É uma excelente ferramenta (*Wireshark*), mas não é aconselhável usá-la para captura em tempo real por causa dos problemas supracitados. Portanto o *Tcpdump* é mais recomendado para captura de tráfego em tempo real. Ele aceita filtros de expressões, e é bem fácil de filtrar o que se deseja capturar ou não. Usa a biblioteca padrão Libpcap. O interessante de se ter uma biblioteca padrão é que pode se interagir com várias outras ferramentas. Lembrando que a biblioteca *Winpcap* é para Windows.

O comando tcpdump –i fxp0 exposto na figura 02, faz com que seja capturado todo tráfego que chega e sai da interface fxp0. O resultado do comando é apresentado na seqüência do comando, indicando o endereço de origem, destino e hora em que o tráfego foi capturado. Ao final é impresso na tela o total de pacotes capturados.

Figura 02 – TCPDUMP

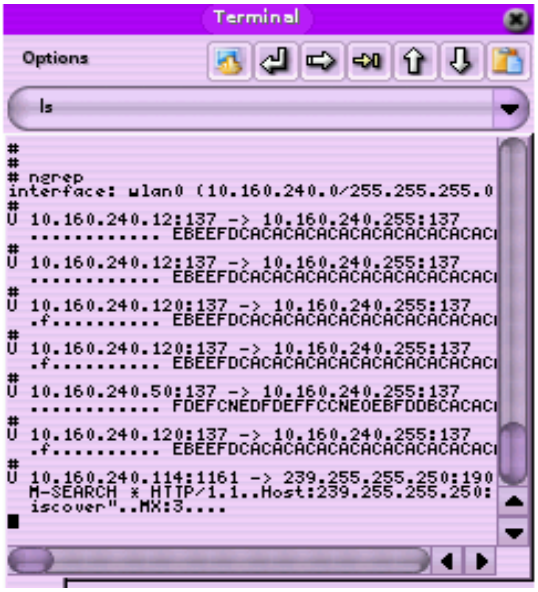
```
192.168.1.253 - PuTTY
web# tcpdump -i fxp0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on fxp0, link-type EN10MB (Ethernet), capture size 96 bytes
17:17:58.096331 IP web.brognoli.br.ssh > 192.168.5.6.3205: P 2020234261:2020234457(196) ack 2038313113 win 65535
17:17:58.454845 IP 192.168.5.6.3205 > web.brognoli.br.ssh: . ack 0 win 15100
17:17:58.674788 IP 192.168.5.6.3205 > web.brognoli.br.ssh: . ack 196 win 16416
17:17:59.111241 IP web.brognoli.br.55242 > 192.168.1.8.domain:  62451+ PTR? 6.5.168.192.in-addr.arpa. (42)
17:17:59.120640 IP 192.168.1.8.domain > web.brognoli.br.55242:  62451 NXDomain 0/1/0 (119)
17:17:59.121484 IP web.brognoli.br.ssh > 192.168.5.6.3205: P 196:520(324) ack 1 win 65535
17:17:59.444729 802.1d unknown version
17:17:59.814581 IP 192.168.5.6.3205 > web.brognoli.br.ssh: . ack 520 win 16092
17:18:00.112780 IP web.brognoli.br.60970 > 192.168.1.8.domain:  62452+ PTR? 8.1.168.192.in-addr.arpa. (42)
17:18:00.121432 IP 192.168.1.8.domain > web.brognoli.br.60970:  62452 NXDomain 0/1/0 (119)
17:18:00.122292 IP web.brognoli.br.ssh > 192.168.5.6.3205: P 520:860(340) ack 1 win 65535
17:18:00.122990 IP web.brognoli.br.ssh > 192.168.5.6.3205: P 860:1024(164) ack 1 win 65535
17:18:00.405193 IP 192.168.5.6.3205 > web.brognoli.br.ssh: . ack 1024 win 15588
17:18:01.123018 IP web.brognoli.br.ssh > 192.168.5.6.3205: P 1024:1540(516) ack 1 win 65535
17:18:01.366938 IP 192.168.4.4.netbios-ns > 192.168.4.255.netbios-ns: NBT UDP PACKET(137): QUERY; REQUEST; BROADCAST
17:18:01.444768 802.1d unknown version
17:18:01.684885 IP 192.168.5.6.3205 > web.brognoli.br.ssh: . ack 1540 win 15072
17:18:02.117012 IP 192.168.4.4.netbios-ns > 192.168.4.255.netbios-ns: NBT UDP PACKET(137): QUERY; REQUEST; BROADCAST
^C
18 packets captured
25 packets received by filter
0 packets dropped by kernel
web#
```

Fonte: Tcpdump.org

2.1.2 Ngrep

Ngrep é um analisador de pacotes de rede, escrito por Jordan Ritter. É executado com linha de comando, e baseia-se na biblioteca pcap. Ngrep permite a correspondência entre padrões ou expressões regulares dentro dos dados do payload do pacote, utilizando a sintaxe Grep do GNU, mostrando assim dados do pacote de forma amigável.
Ngrep busca automatizar certos processos. Trabalha com uma série de protocolos. Exemplo de uso: detecção de tráfego anômalo, composto por sequências de códigos NOPs. Pode ser usado o Shell code do código NOP e usá-lo como expressão regular e assim buscar em um dump inteiro. Ele mostrará todos que possuem códigos NOPs. O Ngrep busca fazer esse tipo de varredura. Ngrep trabalha também em tempo real. É análogo ao comando Grep. É possível mandar o Ngrep ler um arquivo.

Figura 03 – NGREP



Fonte: Irongeek.com

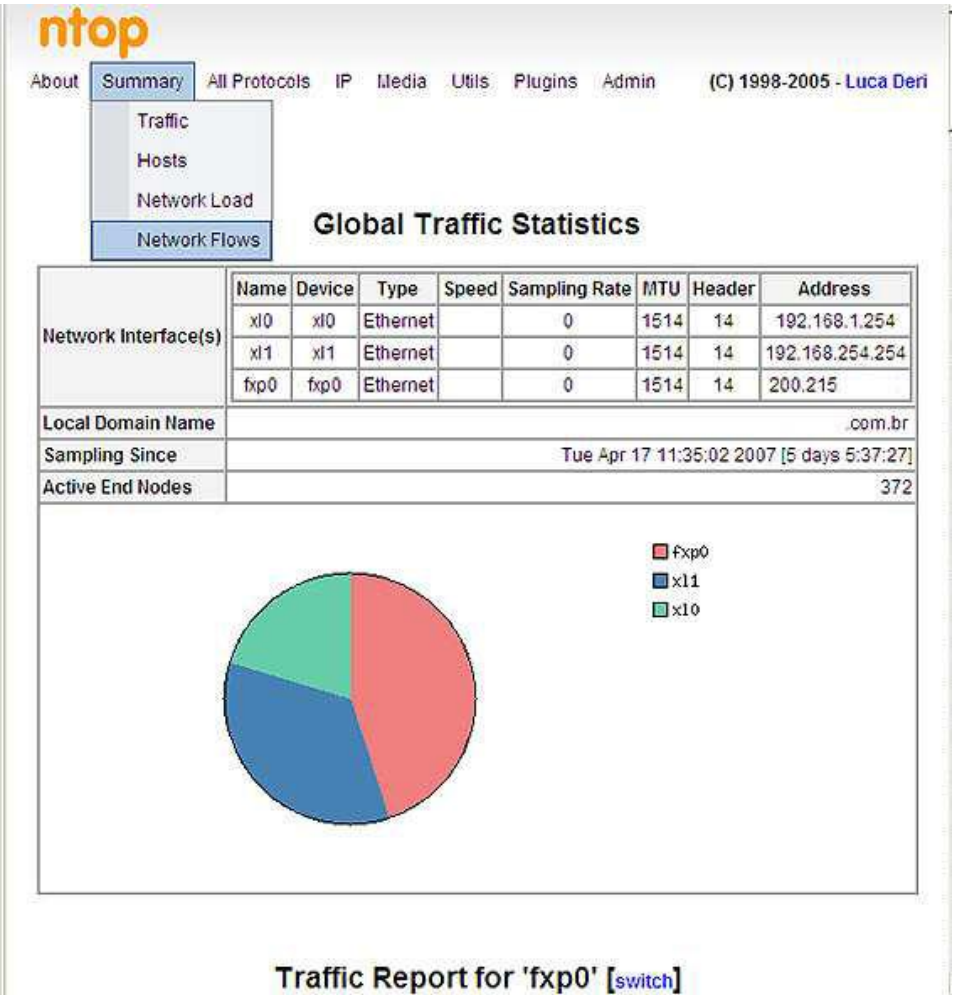
2.1.3 Ntop

NTOP é uma ferramenta, criado pela NTOP.org, interessante para destacar eventos irregulares em uma rede. Ele exibe dados estatísticos sobre protocolos utilizados, atividades de rede dos ativos envolvidos, os destinos e origens etc. Com isso é possível ter uma idéia básica do que é o comportamento normal em uma determinada rede. E quando isso é visualizado em forma de gráficos, rapidamente se conseguirá visualizar um tráfego irregular ou quando determinada máquina excedeu sua utilização, o que pode caracterizar infecção e por último quando se tem um protocolo que não é costume utilizar e que está sendo utilizado em excesso.

Geralmente são protocolos esquecidos que ninguém usa. Exemplo: Existe um protocolo X que ninguém na sua rede faz uso, e então nota-se uma máquina utilizando esse protocolo. Provavelmente existirá um servidor malicioso para fazer um *covert channel* canal de comunicação que permite a um processo transmitir informação de uma forma que viola a política de segurança do sistema, PEIXINHO (2007) e assim fazer uma rede secreta tunelada dentro desse protocolo. Ntop lhe ajuda a ver tudo isso. Ele tira um retrato da sua rede, fazendo com que saiba o que é normal e o que é anormal.

Um dos pontos fortes do NTOP apontado na figura 04 está relacionado à sua interface, quanto a sua facilidade de uso e compreensão. Este ponto pode ser observado na Figura 04 onde se tem a tela principal da ferramenta demonstrando seu menu principal, interfaces monitoradas, o gráfico de distribuição de tráfegos pelas interfaces de redes presentes, tempo de atividade entre outras informações que podem ser observadas no site do fabricante.

Figura 04 – NTOP



Fonte: Ntop.org

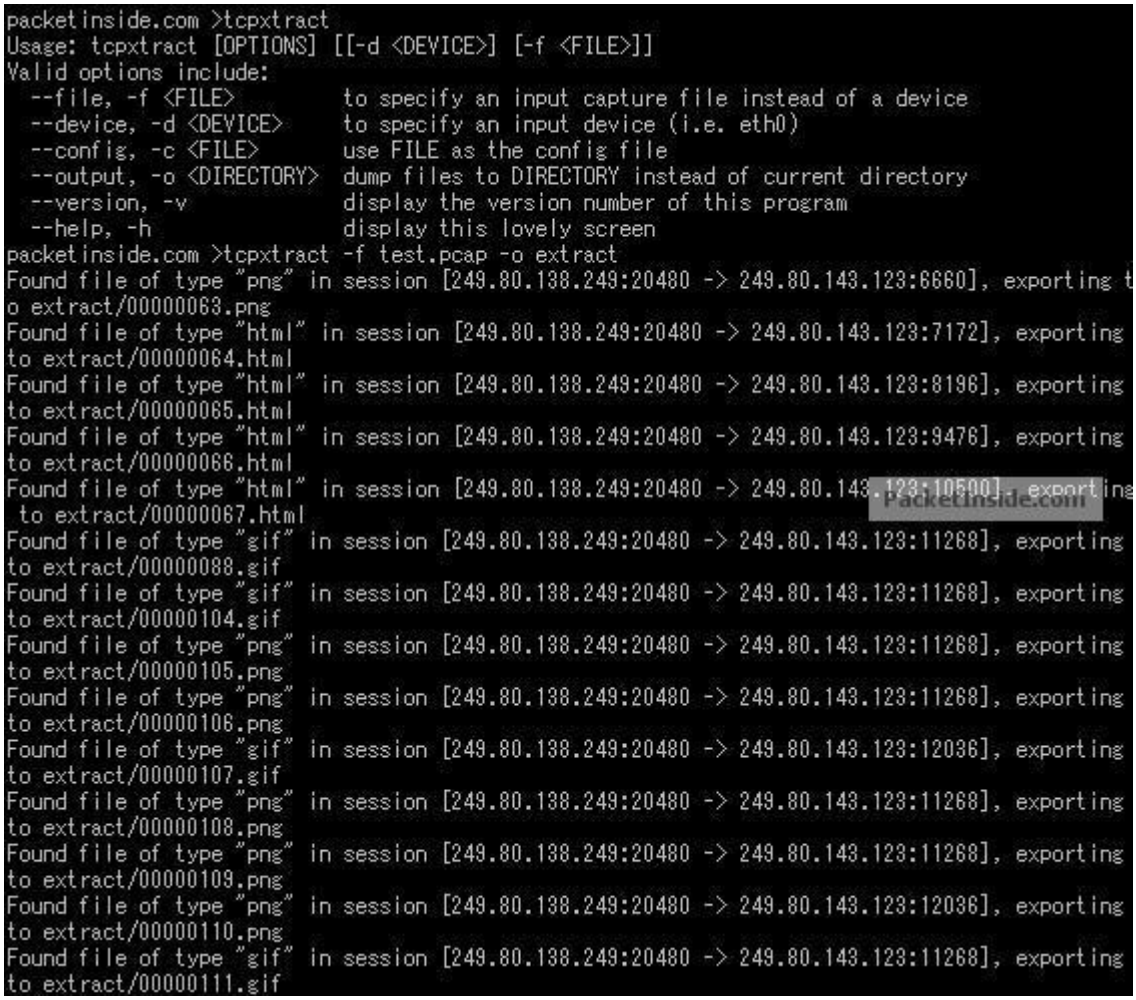
2.1.4 Tcpextract

Tcpextract é uma ferramenta criada para extrair arquivos do tráfego de redes baseadas em assinaturas de arquivo, e foi escrito por Nick Harbour. A idéia do *Tcpextract* é a extração de arquivos, e ele tenta automatizar um processo que se chama de *file carving*. *File Carving* é o reconhecimento em um dump de rede do que é um arquivo. Isso não é algo trivial de automatizar. O *Tcpextract* acha no meio do *dump*, assinaturas, que geralmente são baseadas em Header e *Footers* (cabeçalhos e rodapés). Porém é importante ressaltar que isso depende da arquitetura do arquivo. Tem arquivos que a assinatura se encontra no cabeçalho desses arquivos e em outros casos a assinatura pode ser encontrada no rodapé.

Entretanto acontece que a maioria dos protocolos que envolvem transferência de arquivo envolve algum tipo de codificação. Só observando que codificação não é criptografia. O que dificulta a automatização é a questão da codificação. Exemplo: A maioria dos

mensageiros instantâneos tem protocolos próprios de codificação de envio de arquivos para otimização do envio. Então essa ferramenta é interessante, mas possui limitações que são os tipos de arquivos que ela conhece. Funciona muito bem para mídias de uma maneira geral, os principais *codecs*, documentos do Office, PDFs, e softwares em geral mais reconhecidos. Também apresenta limitação pela maneira como o arquivo veio transferido. NFS, FTP. Coisas simples que passam totalmente em modo RAW são fáceis, mas existem as limitações da automatização.

Figura 05 - TCPEXTRACT



Fonte: Packetinside.com

2.1.5 Wireshark

Esse software originalmente chamado Ethereal foi escrito por Gerald Combs no final dos anos 90. Em maio de 2006, mudou de nome e o projeto foi rebatizado Wireshark devido a problemas de marca registrada. Wireshark possui várias funcionalidades. O destaque é a análise de cabeçalhos em árvore (encapsulamento). Todas as comunicações numa rede começam em uma origem e são enviadas a um destino. As informações enviadas através da rede são conhecidas como dados ou pacotes de dados. Se um computador (host A) desejar enviar dados para outro computador (host B), os dados devem primeiro ser empacotados através de um processo chamado encapsulamento. O encapsulamento empacota as

informações de protocolo necessárias antes que trafeguem pela rede. Assim, à medida que o pacote de dados desce pelas camadas do modelo OSI, ele recebe cabeçalhos, trailers e outras informações. FERREIRA (2007)

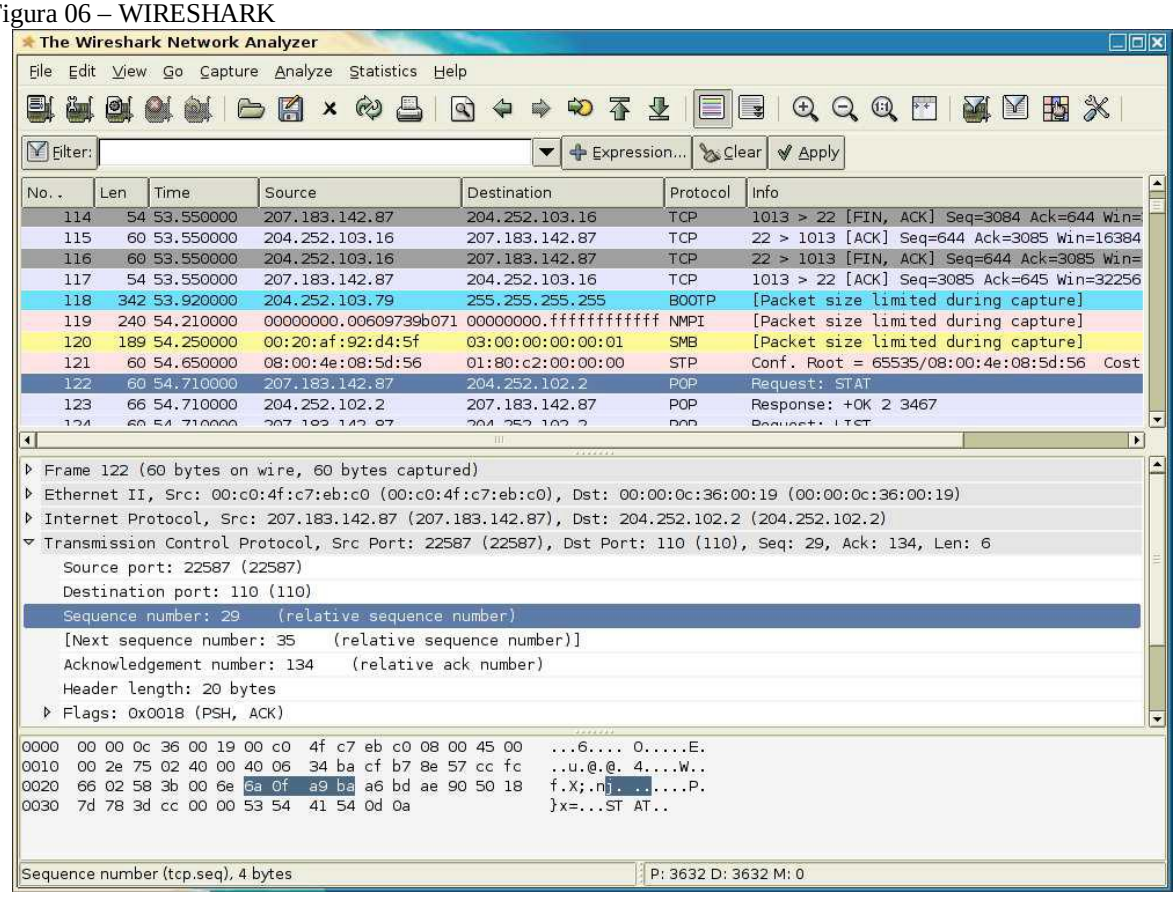
Esse é o processo de encapsulamento. Acontece o inverso quando se recebe o pacote. Pacote chega pela camada física e vai subindo sentido aplicação e para cada camada que sobe vai desencapsulando. *Wireshark* faz essa análise de forma fácil. Basta clicar em cima do pacote e lhe é mostrado em árvore, estilo MS Windows Explorer. Portanto se consegue visualizar *overhead* (informação acompanhada de uma mensagem na rede para garantir a transferência sem erros para o destino desejado) de todas as camadas.

Aplicação de regras e filtros – Os filtros do *Wireshark* não são intuitivos em geral, porque são muitos filtros para muitos protocolos diferentes. Não é tão básico como o *Tcpdump*. Entretanto pode-se criar o filtro de maneira interativa. Basta clicar em um pacote e optar por criar filtro selecionado e o software irá interagindo com a pessoa, sem precisar decorar a sintaxe dos filtros.

O *Wireshark* conta com funcionalidades específicas para análise de tráfego VoIP - Algo interessante que veio de versões anteriores para agora. Essa é uma grande vertente em forense de rede, que seria uma analogia ao grampo telefônico, para que possa se reconstruir sessões VoIP através de um dump de rede. *Wireshark* já vem com funcionalidades para isso.

Esta ferramenta decodifica os frames capturados no formato *libpcap* e os mostra de forma gráfica, como ilustrado na figura a seguir:

Figura 06 – WIRESHARK



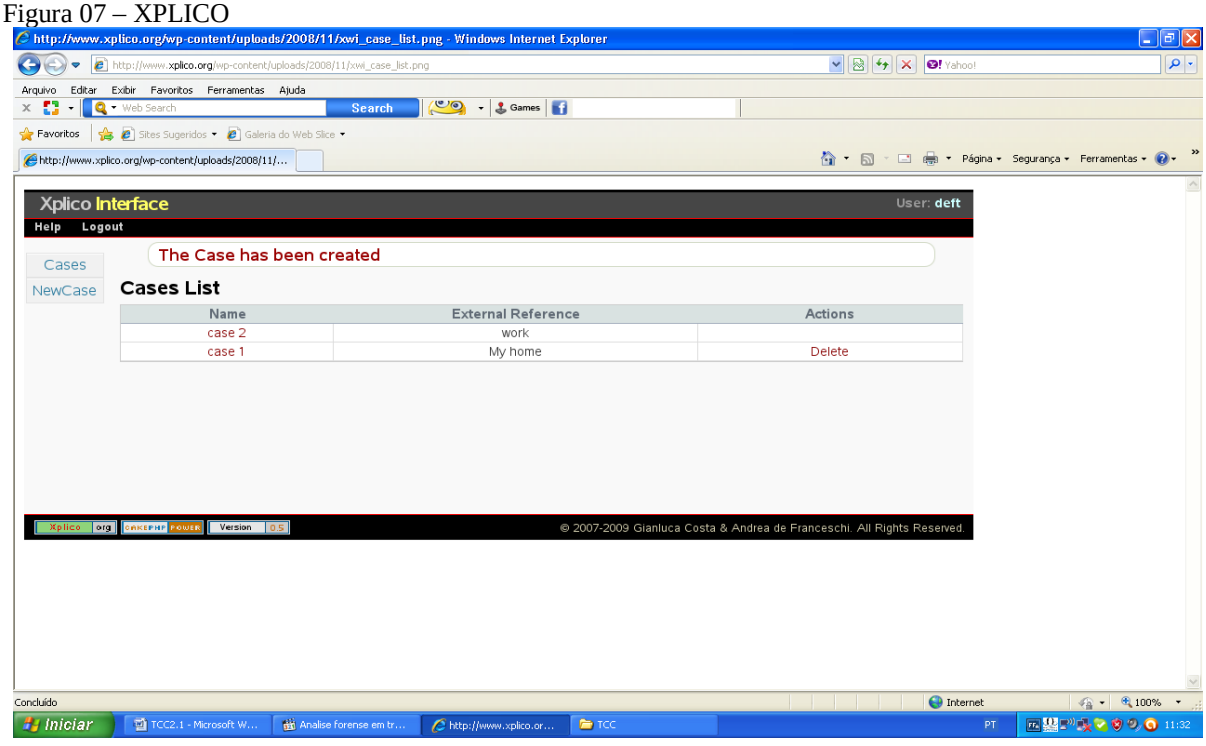
Fonte: Wireshark.org

2.1.6 Xplico

Essa ferramenta, criada pela Xplico.org, lançou o conceito de NFAT. As ferramentas anteriores possuem objetivos específicos e podem ser utilizados em forense. Exemplo: *Wireshark* não é ferramenta de forense, é uma ferramenta de análise de pacote de tráfego, mas pode ser utilizado como ferramenta forense. O *Xplico* nasceu com a idéia de ser uma ferramenta para forense de rede. Ele faz tudo o que se faz com o *Wireshark* e tenta automatizar. O *Xplico* recebe um *dump* de rede e extrai informações. Reconhece por exemplo tudo que é email e já separa como se fosse um *inbox* de um cliente de email. Tudo que for HTTP ele reproduz as páginas. O que for VoIP, pode-se dar um play e ele então reproduz a chamada. No FTP ele mostrará origem e destino de um arquivo e também qual é o tamanho do arquivo.

Xplico é Open Source. Na figura 07 é demonstrado que o *Xplico* é para forense. Ele possui hierarquia, onde se cria um caso e dentro do caso se anexa hosts, dentro dos hosts podem ser incluídas imagens e aí se trabalha dentro das referidas imagens. Essas imagens podem ser as partições 1,2,3 do HD, partição 1,2,3 do *Pendrive* etc.

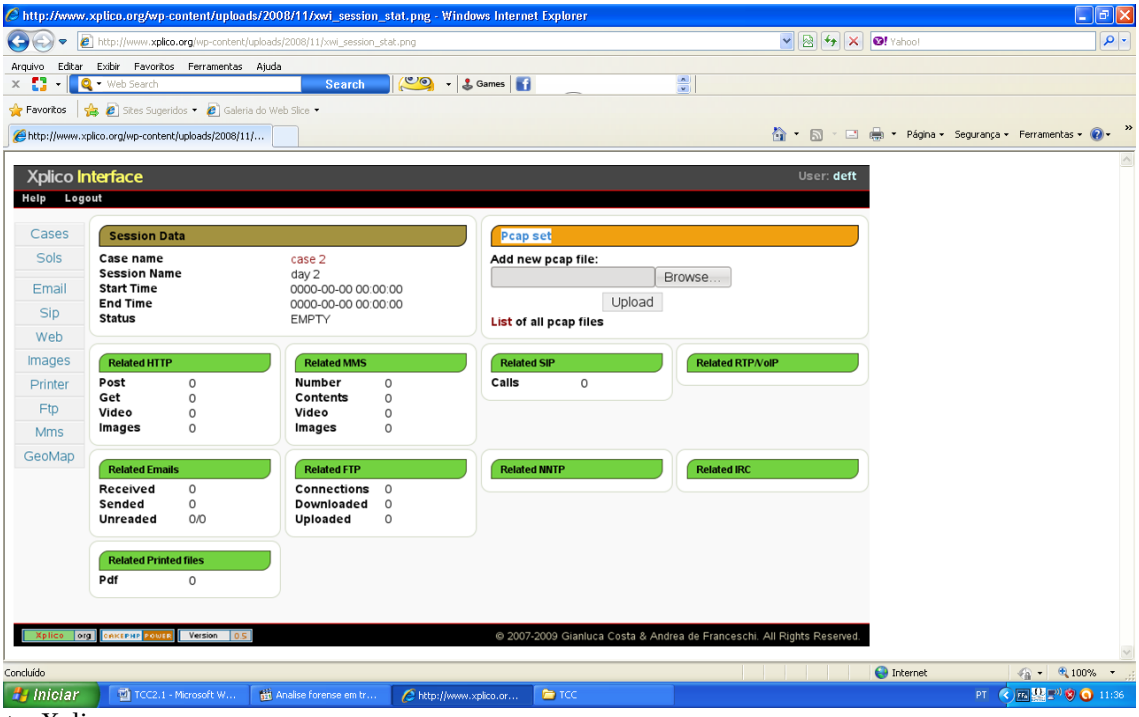
Figura 07 – XPLICO



Fonte: Xplico.org

A figura 08 é a parte mais interessante. É o *dashboard* do *Xplico*. Por exemplo: Pode-se colocar um *Pcap* file e fazer um *upload*. O *Xplico* automaticamente carregará esse arquivo e fará um parse e então essa tela acima ficará com vários dados. O *Xplico* mostra o que é FTP, SIP, MMS, HTTP. E ele atualiza os seus dados em tempo real. Mostra também à hora em que foi iniciado o trabalho.

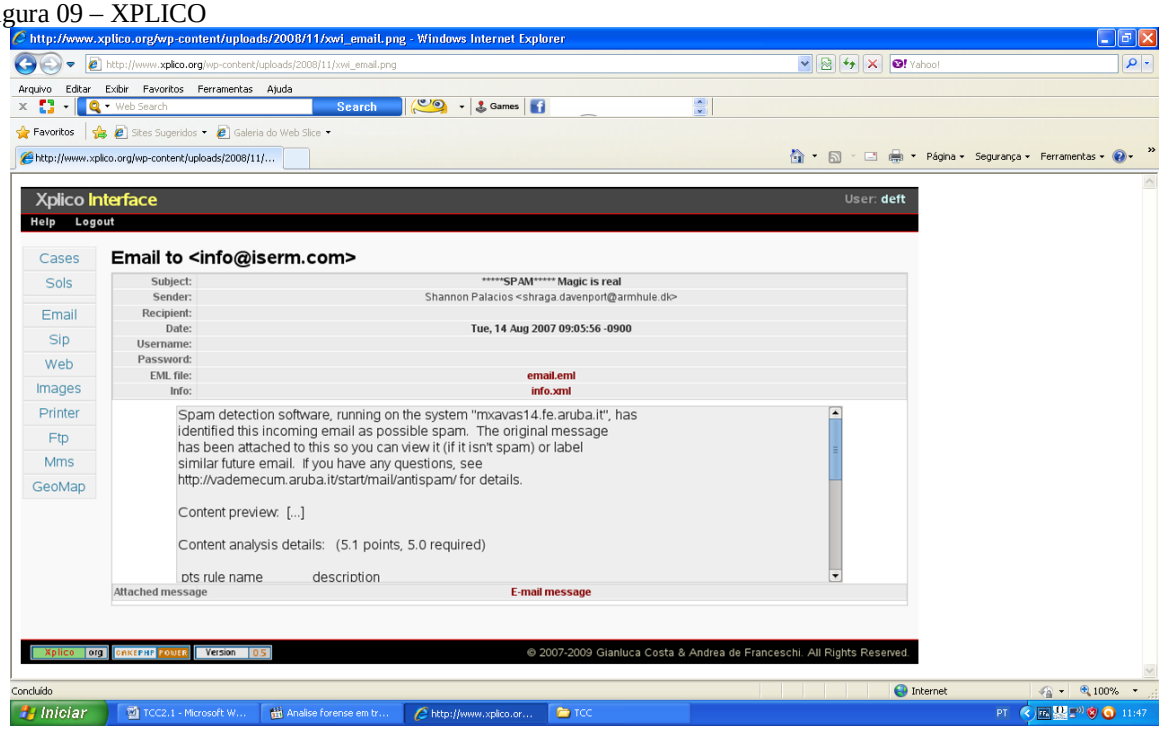
Figura 08 – XPLICO



Fonte: Xplico.org

Tela de email (Figura 09) – Mostra o header da mensagem, o corpo e o anexo. O Xplico faz captura em tempo real.

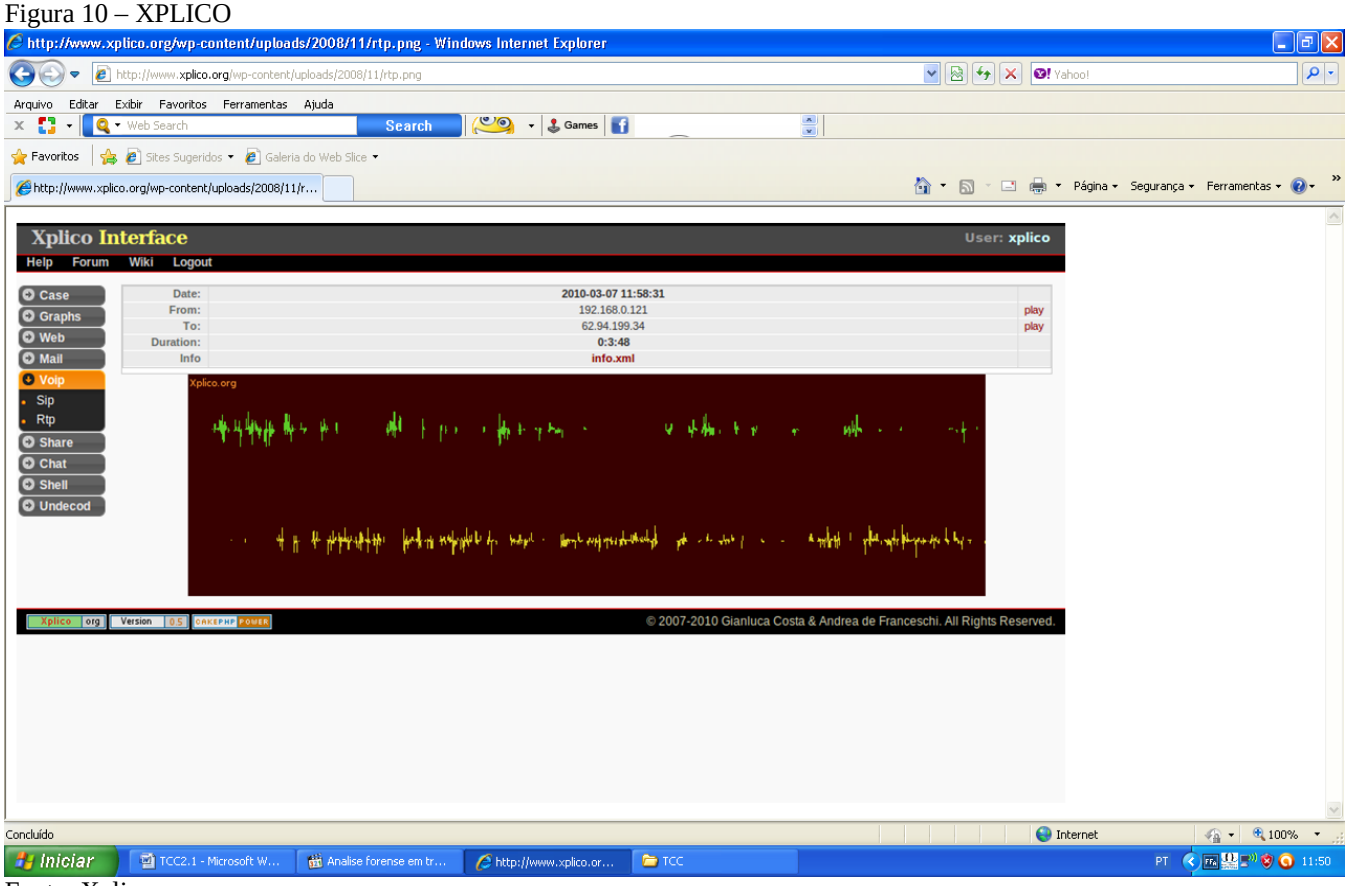
Figura 09 – XPLICO



Fonte: Xplico.org

Tela de Voip (Figura 10) – SIP e RTP (mais usado na fase de teste). SIP é largamente utilizado para voip. O *Xplico* mostra origem e destino e faz uma reconstrução do arquivo de áudio.

Figura 10 – XPLICO



Fonte: Xplico.org

3 RESULTADOS

No quadro 01 é apresentado um comparativo com as ferramentas usadas ao longo desse trabalho, explicitando suas principais funcionalidades.

Quadro 01: Comparativo das Funcionalidades das Ferramentas

Funcionalidades	TCPDUMP	NGREP	NTOP	TCPEXTRACT	WIRESHARK	XPLICO
Plataforma Windows	Sim	Sim	Sim	Não	Sim	Não
Plataforma Linux	Sim	Sim	Sim	Sim	Sim	Sim
Captura de pacotes	Sim	Sim	Sim	Sim	Sim	Sim
Armazenamento dos pacotes	Sim	Sim	Sim	Sim	Sim	Sim
Exibição dos pacotes	Sim	Sim	Sim	Sim	Sim	Sim
Utilização de filtragem de pacotes	Sim	Sim	Sim	Sim	Sim	Sim
Exibição da duração da captura	Não	Não	Sim	Não	Sim	Sim
Dados da utilização da banda utilizada	Não	Não	Sim	Não	Sim	Sim
Contagem de pacotes	Sim	Sim	Sim	Sim	Sim	Sim
Estatística dos protocolos	Não	Não	Sim	Não	Sim	Sim
Estatísticas dos endereços MAC	Não	Não	Sim	Não	Sim	Sim
Estatísticas dos endereços IP	Não	Não	Sim	Não	Sim	Sim
Estatísticas dos endereços MAC	Não	Não	Sim	Não	Sim	Sim
Estatísticas das conversas entre endereço MAC e IP	Não	Não	Sim	Não	Sim	Sim
Estatísticas das conversas TCP e UDP	Não	Não	Sim	Não	Sim	Sim
Geração de gráficos	Não	Não	Sim	Não	Sim	Sim
Geração de Logs	Não	Não	Sim	Não	Não	Sim

Fonte: Revista Eletrônica do Alto Vale do Itajaí. Número 01, agosto 2012.

4 DISCUSSÃO

Ao observar as características das ferramentas, interpreta-se que o *Tcpdump*, apesar de operar em linha de comando, é capaz de capturar os pacotes de rede com maior integridade,

evitando possíveis perdas desses pacotes. Entretanto, o *Wireshark*, por possuir interface gráfica, facilita a visualização e entendimento dos pacotes de rede capturados através de gráficos. Assim, para obter uma análise mais precisa, sugere-se a utilização em conjunto dessas duas ferramentas. Notou-se que as funcionalidades disponíveis nas demais ferramentas podem atender com eficácia a realização das atividades de captura e análise de pacotes.

5 CONCLUSÃO

Segundo (VENEMA, 2007) a computação forense é um campo que talvez devesse ser levado mais a sério do que outras disciplinas na ciência da computação. Eles citam ainda que, os programas e as pessoas envolvidas na coleta e análise dos vestígios devem ter cuidado especial, porque seus resultados podem influenciar seriamente a liberdade individual, a vida, o emprego das pessoas e muito mais.

O presente artigo teve como objetivo a realização de um estudo com a ferramenta de análise forense em tráfego de redes (*Xplico*) e com as demais ferramentas que são fundamentalmente para análise de tráfego em rede, mas que executam também a análise forense em tráfego de redes e assim podem identificar vulnerabilidades nos sistemas computacionais. No decorrer do estudo, foram realizadas investigações forenses usando capturas de tráfego de rede e todas as ferramentas apresentaram resultados satisfatórios que possibilitam um efetivo diagnóstico em uma rede de computadores.

A análise de tráfego permite, dentre outras possibilidades: Encontrar pontos de bloqueio na rede; Detectar anomalias na rede; Descobrir equipamentos e cabeamento defeituosos; Observar importantes mensagens de sistema não mostradas pelas aplicações. A análise dependerá, principalmente, do conhecimento a respeito de protocolos de rede e modelo OSI.

Através do uso adequado das ferramentas computacionais para o processo de investigação forense, concluí-se que apesar das diferenças apresentadas entre as ferramentas, todas apresentam condições de oferecer um exame adequado em um tráfego de redes. Dessa forma, a afirmação feita na linha anterior foi confirmada, considerando que análise forense em tráfego de redes realizada nesse estudo pode contribuir de forma eficaz para o trabalho, por proporcionar a quem possa interessar uma melhor visão sobre os procedimentos de uso dessas ferramentas.

De todo o exposto, concluiu-se que a análise forense em tráfego de redes quando feita de forma correta, por meio de ferramentas adequadas, é de grande valia para qualquer organização por poder deixá-la imune a erros futuros em seu sistema computacional. Evidencia, além disso, que a perícia forense tem uma função substancial quando se adapta constantemente às mudanças tecnológicas para obter resultados desejados, sendo o papel do perito de suma importância nesse processo. Espera-se com esse trabalho contribuir para a área forense computacional.

5 1 TRABALHOS FUTUROS

Como referência para futuros trabalhos, proponho uma pesquisa mais ampla usando uma gama maior de ferramentas que fazem a análise forense em tráfego de redes, assegurando assim um trabalho futuro com maior abrangência.

FORENSIC ANALYSIS ON NETWORK TRAFFIC

MARCOS GODINHO VELOZO

Abstract:
Due to the increasing advance of computer networks technology it has been fostered a change in the standards for the computing field where information that was previously reduced, became common worldwide. The networks are becoming larger, more complex and interconnected through the Internet. Global interconnection easiness has favored the increase of complexity regarding the administration and management of those networks. Besides being large the amount of devices, the transmitted data and the demand for availability have accompanied the growth as well, making the task of managing those environments even more difficult. Network management support tools, as well as the core of these tools (their libraries), play an important role concerning the activities related to network availability and security. Many of those support tools use data capture as a way of obtaining information from the network, and in many cases that caption is made through libpcap, which is a library that implements the passive capture of traffic. The analysis of tools for forensic analysis was conducted in the network and it reached the objective of identifying the factors that influence the performance of those tools besides reaching a forecasting model based on a combination of the analyzed factors.

Keywords: Computer Networks. Network forensics. Tools

REFERÊNCIAS BIBLIOGRÁFICAS

BARBOSA, Angelo Mota; MEIRELLES, Davi; PEIXOTO, Holanda Osvaldo**. Análise de Tráfego no Xplico utilizando arquivo pcap gerado pelo tcpdump**. Disponível em:<http://osvaldohp.blogspot.com.br/2011/08/analise-de-trafego-no-xplico-utilizando.html>. Acessado em: 05/10/2012.

COUTO, Felipe Santos; SANTOS, Alex Ferreira; LOVATO, Agnaldo Volpe. **Estudo comparativo de ferramentas analisadoras de pacotes em rede TCP/IP**. Revista Eletrônica do Alto Vale do Itajaí, n. 1, agosto 2012. Disponível em: <www.site.com.br>. Acesso em 20/10/2012.

FERREIRA, Rafael Soares. **Análise Forense de tráfego de rede**. Disponível em:<http://www.blog.clavis.com.br/webinar-2-analise-forense-em-trafego-de-rede/>. Acessado em: 10/09/2012.

FEBRERO, Borja Merino. **Traffic analysis with wireshark: Inteco-cert**. Feb. 2011. Disponível em:< http://cert.inteco.es/extfrontinteco/img/File/intecocert/EstudiosInformes/cert_trafficwireshark.pdf>. Acessado em 23/08/2012.

FILHO, João Eriberto Mota. **Análise de tráfego em redes TCP/IP com tcpdump**. São Paulo: Novatec, 2012.

LIBPCAP Documentation. Disponível em:< http://www.tcpdump.org/>. Acessado em 25/09/2012.

MARCHETTI, Breno Rangel Borges. **Um método simples para detecção on-the-fly de arquivos e suas mutações aplicado ao combate à pedofilia e outros crimes na Internet**. Disponível em:< http://www.icofcs.org/2008/ICoFCS2008-pp03.pdf >. Acessado em: 01/10/2012.

MILAGRE, José Antonio. **Perícia eletrônica, computação forense, forense digital ou perícia digital: uma proposta para padronização da terminologia**. Disponível em: <http://josemilagre.com.br/blog/wp-content/uploads/2011/10/Artigo-Forense-Digital-Uma-proposta-para-padroniza%C3%A7%C3%A3o-da-terminologia-Jose-Milagre-01-02-2012-v1.pdf>. Acessado em 15/08/2012.

NTOP Documentation. Disponível em:<http://sourceforge.net/projects/ntop>. Acessado em: 01/10/2012.

PEIXINHO, Ivo de Carvalho. **Covert Channels**: o que são, o que fazem e como se prevenir contra eles. Dez. 2006. Disponível em:< ftp://ftp.registro.br/pub/gts/gts08/02-CovertChannels.pdf>. Acessado em 24/08/2012.

SEGURA, Colorado Rubén. **Monitorización de tráfico hacia internet**. Disponível em: <http://upcommons.upc.edu/pfc/bitstream/2099.1/11644/1/69621.pdf>. Acessado em: 09/10/2012.

SILVA, Ramicés dos Santos. **Análise de desempenho da biblioteca libpcap**. Disponível em:< http://siaibib01.univali.br/pdf/Ramices%20dos%20Santos%20Silva.pdf>. Acessado em: 04/09/2012.

TEIXEIRA, Adriana Ribeiro. **Perícia forense de rede: teoria e prática**. Nov. 2009. Disponível em:< http://www.ginux.ufla.br/files/mono-AdrianaTeixeira.pdf>. Acessado em: 30/08/2012

WIRESHARK Documentation. Disponível em:<http://www.wireshark.org/docs>. Acessado em: 25/09/2012.

XPLICO. Disponível em: <http://www.xplico.org/screenshot>. Acessado em: 07/10/2012.

AS 75 MELHORES Ferramentas de Segurança para Sistemas em Rede. Disponível em: <http://insecure.org/tools/tools-pt.html>. Acessado em: 10/10/2012.

DARKSTAT : um analisador de tráfego. Disponível em: <http://www.linuxfocus.org/Portugues/September2004/article346.meta.shtml>. Acessado em: 10/10/2012.

DEMONSTRAÇÕES e Gabarito do Desafio Prático "Análise Forense em Tráfego de Rede". Disponível em: <http://www.seginfo.com.br/demonstracoes-e-gabarito-do-desafio-pratico-analise-forense-em-trafego-de-rede/>. Acessado em: 10/10/2012.

EXTRAIR ficheiros de Capturas de rede. Disponível em:< http://miguellopes.net/extrair-ficheiros-de-capturas-de-rede>. Acessado em: 07/10/2012.

ISTF. Disponível em: <http://www.istf.com.br/showthread.php/11682-NGrep-Sniffer-com-caracter%C3%ADsticas-do-Grep>. Acessado em: 10/10/2012.

LINUX Security. Using NGREP. Disponível em: <http://www.linuxsecurity.com.br/article.php?sid=435>. Acessado em: 10/10/2012.

MAPEAMENTO de Redes com nmap: ferramenta de código aberto com diversas funcionalidades. Disponível em: <http://www.seginfo.com.br/category/artigos/>. Acessado em: 10/10/2012.

MONITORIZACIÓN de red con NTOP. Disponível em: <http://www.solid-rock-it.com/web-solid-rock/blog/index.php/2008/03/11/35-monitorizacion-de-red-con-ntop>. Acessado em: 10/10/2012.

MONITORANDO redes utilizando NTOP. Disponível em: <http://www.vivaolinux.com.br/artigo/Monitorando-redes-utilizando-Ntop>. Acessado em: 10/10/2012.

NGREP : um filtro de rede avançado. Disponível em: <http://tecnologoderedes.blogspot.com.br/2012/01/ngrep-um-filtro-de-rede-avancado.html>. Acessado em: 10/10/2012.

NTOP : administración de la Red. Disponível em: <http://administradores.educarex.es/wiki/index.php/Ntop_(Administraci%C3%B3n_de_la_Red>. Acessado em: 10/10/2012.

SEGURIDAD y Redes. Tcpxtract. Extrayendo ficheros del tráfico de red. Disponível em: <http://seguridadyredes.wordpress.com/2010/10/26/tcpxtract-extrayendo-ficheros-del-trafico-de-red>. Acessado em: 09/10/2012.

SSEGURANÇA: Expressões Regulares em Resposta a Incidentes. Disponível em: <http://sseguranca.blogspot.com.br/search/label/ngrep>. Acessado em: 10/10/2012.

TUTORIAL sobre o wireshark. Disponível em: <http://pratesdicas.wordpress.com/2011/02/28/tutorial-sobre-o-wireshark/>. Acessado em: 10/10/2012.