UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

SIT E CLIT: FERRAMENTAS E METODOLOGIA PARA APRIMORAMENTO DE INVESTIGAÇÕES CRIMINAIS UTILIZANDO INTERCEPTAÇÕES DE CONEXÃO À INTERNET

ANDRÉ PERON

ORIENTADOR: FLÁVIO ELIAS GOMES DE DEUS

DISSERTAÇÃO DE MESTRADO PROFISSIONAL EM ENGENHARIA ELÉTRICA
ÁREA DE CONCENTRAÇÃO INFORMÁTICA FORENSE E SEGURANÇA DA INFORMAÇÃO

PUBLICAÇÃO: PPGENE.DM – 096/2012
BRASÍLIA/DF: FEVEREIRO – 2012

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

SIT E CLIT: FERRAMENTAS E METODOLOGIA PARA APRIMORAMENTO DE INVESTIGAÇÕES CRIMINAIS UTILIZANDO INTERCEPTAÇÕES DE CONEXÃO À INTERNET

ANDRÉ PERON

DISSERTAÇÃO DE MESTRADO PROFISSIONALIZANTE SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE.

APROVADA POR:

_____

Flávio Elias Gomes de Deus, Doutor, ENE/FT
(Orientador)

_____

Rafael Timóteo de Sousa Júnior, Doutor, ENE/FT
(Examinador Interno)

_____

Robson de Oliveira Albuquerque, Doutor, ABIN
(Examinador Externo)

BRASÍLIA/DF, 02 DE FEVEREIRO DE 2012

FICHA CATALOGRÁFICA

PERON, ANDRE
SIT E CLIT: Ferramentas e Metodologia para Aprimoramento de Investigações Criminais Utilizando Interceptações de Conexão à Internet [Distrito Federal] 2012.
xv, 132p., 297 mm (ENE/FT/UnB, Mestre, Engenharia Elétrica, 2012). Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia. Departamento de Engenharia Elétrica.

1. Interceptação de Internet
2. Interceptação legal
3. NFAT
4. Tráfego de rede
I. ENE/FT/UnB
II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

PERON, A. (2012). SIT E CLIT: Ferramentas e Metodologia para Aprimoramento de Investigações Criminais Utilizando Interceptações de Conexão à Internet. Dissertação de Mestrado em Engenharia Elétrica, Publicação PPGENE.DM – 096/2012, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 132p.

CESSÃO DE DIREITOS

AUTOR: André Peron.

TÍTULO: SIT E CLIT: Ferramentas e Metodologia para Aprimoramento de Investigações Criminais Utilizando Interceptações de Conexão à Internet.

GRAU: Mestre ANO: 2012

_____

André Peron
EQSW 103/104 Lote 1 Bloco B, Sudoeste
70670-350 Brasília – DF – Brasil.

AGRADECIMENTOS

O presente trabalho foi realizado com o apoio do Departamento Polícia Federal – DPF, com recursos do Programa Nacional de Segurança Pública com Cidadania – PRONASCI, do Ministério da Justiça.

RESUMO

SIT E CLIT: FERRAMENTAS E METODOLOGIA PARA APRIMORAMENTO DE INVESTIGAÇÕES CRIMINAIS UTILIZANDO INTERCEPTAÇÕES DE CONEXÃO À INTERNET

Autor: André Peron
Orientador: Flávio Elias Gomes de Deus
Programa de Pós-graduação em Engenharia Elétrica
Brasília, Fevereiro de 2012

A interceptação de conexões à Internet é uma técnica de investigação criminal prevista pela legislação brasileira sob determinadas condições, que consiste na captura e análise do tráfego de rede de determinada conexão sem o conhecimento dos investigados.

Diversas dificuldades têm inibido seu uso, tais como falta de padronização na disponibilização do tráfego pelos provedores de conexão, infraestrutura inadequada para recebimento do tráfego e ferramentas ineficientes para tratamento e análise do tráfego.

Neste trabalho são propostos dois sistemas (SIT e CLIT) e metodologia de investigação utilizando-os. O SIT (Servidor de Interceptação Telemática) é uma infraestrutura central para recebimento do tráfego e o CLIT (Cliente de Interceptação Telemática) trata-se de ferramenta de obtenção, tratamento, importação e análise de tráfego. Os sistemas visam à simplificação do uso de interceptações de conexões de Internet nas investigações e, em consequência, o aumento do seu uso.

Os resultados obtidos nos experimentos demonstram que a metodologia de investigação utilizando os sistemas desenvolvidos apresentou, além da simplificação do processo, diversos benefícios que potencializam o uso das interceptações de conexão à Internet pelos órgãos de investigação criminal brasileiros.

ABSTRACT

SIT AND CLIT: TOOLS AND METHODOLOGY FOR IMPROVING CRIMINAL INVESTIGATIONS USING INTERCEPTION OF INTERNET CONNECTION

Author: André Peron
Supervisor: Flávio Elias Gomes de Deus
Programa de Pós-graduação em Engenharia Elétrica
Brasília, February of 2011

The interception of Internet connections is a technique of criminal investigation admitted by Brazilian legislation under certain conditions, which consists of capturing and analyzing network traffic of a given connection without the knowledge of the person being investigated.

Several problems have inhibited its use, such as lack of standardization in the delivery of traffic by connecting providers, inadequate infrastructure for receiving traffic and inefficient tools for treatment and traffic analysis.

This paper proposes two systems (SIT and CLIT) and investigation methodology using them. The SIT (Telematics Interception Server) is a central infrastructure for receiving traffic and CLIT (Telematics Interception Client) it is a tool for obtaining, processing, importing and analyzing the traffic. Systems aimed at simplifying the use of interception of Internet connections in the investigations and, consequently, increased their use.

The results obtained in the experiments demonstrate that the investigation methodology using the designed systems presented, as well as simplifying the process, several benefits that enhance the use of interception of Internet connection by Brazilian criminal investigation agencies.

SUMÁRIO

LISTA DE TABELAS

LISTA DE FIGURAS

LISTA DE SÍMBOLOS, NOMENCLATURA E ABREVIAÇÕES

ADSL – *Asymmetric Digital Subscriber Line*

AF – Arquivo de Fluxo

AJAX – *Asynchronous JavaScript and XML*

ANATEL – Agência Nacional de Telecomunicações

ASN.1 – *Abstract Syntax Notation One*

BCC – *Blind Carbon Copy*

CALEA – *Communications Assistance for Law Enforcement Act*

CC – *Carbon Copy*

CDMA2000 – *Code Division Multiple Access 2000*

CLIT – Cliente de Interceptação Telemática

CNJ – Conselho Nacional de Justiça

CPI – Comissão Parlamentar de Inquérito

DHCP – *Dynamic Host Configuration Protocol*

DNS – *Domain Name System*

EDGE – *Enhanced Data Rates for GSM Evolution*

ETSI – *European Telecommunication Standards Institute*

ETSI TC LI – *ETSI Technical Committee on Lawful Interception*

FTP – *File Transfer Protocol*

GPRS – *General Packet Radio Service*

GRE – *Generic Routing Encapsulation*

HDSPA – *High-Speed Downlink Packet Access*

HTML – *HyperText Markup Language*

HTTP – *HyperText Transfer Protocol*

ICI – Interceptação de Conexão à Internet

IP – *Internet Protocol*

IRC – *Internet Relay Chat*

ITU-T – *International Telecommunication Union - Telecommunication Standardization Sector*

JPEG – *Joint Photographic Experts Group*

JSON – *JavaScript Object Notation*

LCC – *Linux Cooked Capture*

MSNM – *MicroSoft Network Messenger*

MTU – *Max Transfer Unit*

NFAT – *Network Forensic Analysis Tool*

NIC – *Network Interface Card*

NNTP – *Network News Transfer Protocol*

NSM – *Network Security and Monitoring*

OFT – *OSCAR File Transfer*

OICB – Órgão de Investigação Criminal Brasileiro

OSCAR – *Open System for CommunicAtion in Real-time*

P2P – *Peer-to-Peer*

PA – Ponto de Análise

PCLI – *Packet Cable Lawful Interception*

POP – *Post Office Protocol*

POP3 – *Post Office Protocol Version 3*

PPPoE – *Point-to-Point Protocol over Ethernet*

RFC – *Request for Comments*

RIP – *Routing Information Protocol*

RTCP – *RTP Control Protocol*

RTP – *Real-time Transport Protocol*

SCP – *Secure CoPy*

SFTP – *Secure File Transfer Protocol*

SIP – *Session Initiation Protocol*

SIT – Servidor de Interceptação Telemática

SMTP – *Simple Mail Transfer Protocol*

SNMP – *Simple Network Management Protocol*

SO – Sistema Operacional

TCP – *Transmission Control Protocol*

TI – Tecnologia da Informação

TZSP – *TaZmen Sniffer Protocol*

UDP – *User Datagram Protocol*

URL – *Unified Resource Locator*

VDSL – *Very-high-bit-rate Digital Subscriber Line*

VLAN – *Virtual Local Area Network*

VO – Visualizador de Objetos

VoIP – *Voice over IP*

WAN – *Wide Area Network*

WCDMA – *Wideband Code Division Multiple Access*

WiMAX – *Worldwide Interoperability for Microwave Access*

WLM – *Windows Live Messenger*

XML – e*Xtensible Markup Language*

YahooMSG – *Yahoo!Messenger*

1 INTRODUÇÃO

A interceptação de comunicação de dados para fins de prova em investigação criminal e em instrução processual é prevista pela legislação brasileira. Para se utilizar deste artifício essencial para o sucesso de muitas investigações nos dias de hoje, os órgãos policiais necessitam, além de infraestrutura de *software* e *hardware*, metodologia adequada para a produção de provas para que não venham a ser questionadas na esfera judicial.

Quanto à interceptação telemática, mais especificamente, a Interceptação de Conexão à Internet (ICI), diferentemente da interceptação de comunicações telefônicas que é regida pela mesma lei e já se encontra bem sedimentada nos órgãos policiais, tem se encontrado dificuldade para obter soluções informatizadas que atendam este fim considerando a casuística brasileira.

O uso de infraestrutura inadequada por parte dos órgãos de investigação para recebimento do tráfego capturado pelas operadoras, a incompatibilidade de formatos de arquivos ou pacotes entregues com as ferramentas de tratamento e análise disponíveis e a impossibilidade de automatização dos processos provocam perda de informações e dificultam o uso de ICIs nas investigações.

Neste trabalho são apresentados os sistemas SIT (Servidor de Interceptação Telemática) e CLIT (Cliente de Interceptação Telemática) e a metodologia de investigação aplicada com o uso deles. O SIT é uma infraestrutura central para recebimento do tráfego das operadoras (provedores de conexão à Internet) brasileiras. O CLIT é uma ferramenta de análise de tráfego voltada para os Investigadores dos Órgãos de Investigação Criminais Brasileiros (OICBs) e compatível com os diferentes formatos de arquivos e pacotes entregues pelas operadoras brasileiras. Os resultados produzidos por este estudo objetivam principalmente a simplificação e o aumento do uso de ICIs nas investigações.

1.1 DEFINIÇÃO DO PROBLEMA

As ICIs são solicitadas pelos OICBs para as operadoras através de mandados judiciais. As operadoras disponibilizam o tráfego coletado da conexão solicitada em diversas formas de

entrega, formatos de arquivos de captura e formatos de pacotes (enlaces e encapsulamentos IP – *Internet Protocol*).

A análise do tráfego é conduzida pelo Investigador, policial especializado no crime sob investigação (tráfico de drogas e armas, corrupção, pedofilia, contrabando e descaminho, etc.) que possui conhecimentos básicos de informática. Seu objetivo principal é coletar as comunicações entre os investigados, tais como *e-mails*, conversas, arquivos trocados e outras informações que permitam autoria, materialidade e motivação das infrações penais e a estrutura da organização criminosa. Por força legal, as investigações são conduzidas pela unidade descentralizada responsável pela circunscrição onde ocorreu o crime.

Para a análise do trafego, o Investigador utiliza ferramentas forenses de análise de tráfego (NFATs – *Network Forensic Analysis Tools*). Devido à necessidade de acompanhamento em tempo real[1], uma NFAT é adotada na investigação, pois normalmente não há tempo nem recursos humanos disponíveis para o uso de um conjunto de NFATs.

O Investigador precisa ainda manipular ferramentas para captura de pacotes, servidores de transferência segura, programas de conversão de formato e editores de pacotes, já que as NFATs não são compatíveis com todas as formas de disponibilização de tráfego (formas de entrega, formatos de arquivos e pacotes) das operadoras brasileiras. As ICIs podem envolver também contratação de conexão de Internet com endereço de IP fixo e configuração de *modems* por parte do Investigador.

Além das complexidades citadas inerentes às diferentes formas de disponibilização do tráfego, as NFATs disponíveis não têm atendido as necessidades das investigações brasileiras, a saber:

- Quanto ao tráfego: incompatibilidade com formatos de arquivo ou de pacotes impedem análise parcial ou total do tráfego, ausência de mecanismo de importação automática de arquivos de captura e de gerenciamento de arquivos “já importados” / “não importados”, impedindo em determinados casos o acompanhamento em tempo real do investigado;

[1] O termo “tempo real” é utilizado neste trabalho no sentido de atraso não superior a alguns minutos.

- Quanto à decodificação do tráfego: não decodificação ou decodificação parcial de protocolos/aplicativos de uso comum entre os internautas brasileiros, tais como *chats* através da *web* em sítios de redes sociais e de comunicadores instantâneos e *webmails* brasileiros e impossibilidade de implementação de novos decodificadores externos específicos;
- Quanto à interface com o usuário (no caso, o Investigador): ausência de mecanismos de marcação de objetos analisados, sua classificação e exportação para relatório, nos moldes de ferramentas de análise de interceptação telefônica.

Segundo dados da Agência Nacional de Telecomunicações (ANATEL) de março de 2011, o Brasil possui aproximadamente 16 milhões de conexões à Internet[2] (Serviço de Comunicação Multimídia), 30 milhões de telefones fixos[3] (Acessos Fixos Individuais), 210 milhões de telefones móveis[4] (Serviço Móvel Privativo), sendo que destes, 24 milhões são terminais de banda larga móveis (tecnologias WCDMA - *Wideband Code Division Multiple Access*, CDMA2000 - *Code Division Multiple Access 2000* e dados), ou seja, para cada conexão de Internet existem 6 telefones. Considerando que, independente do crime, a interceptação legal é uma medida de afastamento do sigilo das comunicações do investigado, seria esperado que pelo menos 14% das interceptações legais sejam ICIs.

Pelos dados do mês de agosto de 2011[5] do Conselho Nacional de Justiça (CNJ), que controla o número de interceptações telefônicas e telemáticas autorizadas pelo Poder Judiciário brasileiro, foram monitoradas 17.872 linhas telefônicas (fixas, móveis e VoIP – *Voice over IP*) e apenas 320 endereços eletrônicos ou sistemas de informática/temática, ou seja, apenas 1,83% das interceptações legais são telemáticas e, portanto, números menores ainda são especificamente de ICIs.

---

[2] Disponível em http://www.anatel.gov.br/Portal/verificaDocumentos/documento.asp?numeroPublicacao=261089&pub=original&filtro=1&documentoPath=261089.pdf

[3] Disponível em http://sistemas.anatel.gov.br/sgmu/Localidade/Consolidado/frmListagem.asp?SISQSModulo=17507

[4] Disponível em http://sistemas.anatel.gov.br/SMP/Administracao/Consulta/AcessosPrePosUF/telaConsulta.asp

[5] Disponível em http://www.cnj.jus.br/noticias/cnj/15962-17-mil-linhas-telefonicas-foram-monitoradas-em-2011

As complexidades envolvidas nos processos de recebimento, tratamento e importação do tráfego recebido das operadoras e as limitações das ferramentas disponíveis determinam a baixa utilização de ICIs nas investigações.

1.2 JUSTIFICATIVA

Criminosos, especialmente quando em grupos organizados, necessitam de formas eficientes de comunicação, sendo o telefone e a Internet os meios mais comumente utilizados. A interceptação legal destes meios trata-se de ferramenta fundamental para o sucesso das investigações.

Os principais OICBs já dispõem de tecnologia que permitem o uso da interceptação telefônica de forma cotidiana nas investigações. O mesmo não ocorre com as interceptações telemáticas, mais especificamente das ICIs, que são atualmente realizadas apenas em investigações específicas com recursos humanos e materiais abundantes.

A grande disponibilidade de conexões de Internet de baixo custo, banda larga e de diversas tecnologias, principalmente com a popularização de dispositivos móveis, aliada a ampla divulgação na mídia de investigações que alcançaram sucesso com o uso da interceptação telefônica, tem feito com que criminosos passem a utilizar cada vez mais a Internet e menos o telefone para articulação de suas ações, sendo imprescindível aos OICBs o investimento em pesquisa e desenvolvimento de tecnologias que facilitem a interceptação de comunicações realizadas pela Internet.

1.3 OBJETIVO DA DISSERTAÇÃO

Este trabalho tem por objetivo principal a simplificação do uso de ICI para os Investigadores nos OICBs e, em consequência, tornar seu uso tão frequente quanto as interceptações telefônicas.

Para alcançar esse objetivo principal, os seguintes objetivos específicos foram definidos:

1. Aumentar a abrangência dos dados analisados através da compatibilização com formatos de arquivos de captura e pacotes entregues pelas operadoras brasileiras;
2. Aumentar a confiabilidade na recepção dos dados da operadora;

3. Automatizar processos manuais de *download*, conversão de formatos, extração de enlaces e encapsulamentos, importação de arquivos, etc.;
4. Eliminar controles manuais hoje necessários;
5. Reduzir a quantidade de programas externos a serem utilizadas pelo Investigador;
6. Permitir acompanhamento em tempo real do tráfego do investigado;
7. Padronizar os procedimentos realizados pelo Investigador, que atualmente variam muito devido às diversas formas de disponibilização de tráfego pelas operadoras;
8. Aumentar a quantidade de vestígios decodificados do tráfego (páginas *web*, *e-mails*, *chats*, conversas VoIP, *webmails*, etc.), focando principalmente em protocolos e aplicativos de uso mais comum dos internautas brasileiros, tais como *webmails* brasileiros e *chats* através de redes sociais.

1.4 METODOLOGIA

Para a realização deste trabalho foram realizadas diversas etapas, onde se destacam:

- Consulta de referências bibliográficas para análise global da interceptação telemática e verificação de trabalhos correlatos;
- Análise da legislação brasileira quanto ao uso da interceptação para fins de investigação criminal;
- Trabalho junto aos Investigadores em um OICB para entender as formas de disponibilização de tráfego, infraestrutura e ferramentas utilizadas, a metodologia de trabalho e as dificuldades enfrentadas no dia a dia e diagnosticar o baixo uso das ICIs nas investigações comparadas com as interceptações telefônicas;
- Levantamento da estrutura de Tecnologia da Informação (TI) de um OICB, objetivando propor uma infraestrutura para recebimento do tráfego interceptado;
- Estudo de diversos protocolos e trabalhos publicados sobre os mesmos para o desenvolvimento da NFAT que decodifica os principais protocolos de interesse das investigações dos OICBs e outras deficiências levantadas em outras NFATs atualmente em uso;
- Foram definidos oito critérios de comparação que foram avaliados em seis diferentes cenários de ICIs, a fim de testar a infraestrutura, a ferramenta e a metodologia proposta.

1.5 TRABALHOS CORRELATOS

Embora não tenham sido encontrados durante a pesquisa trabalhos específicos sobre o uso de ICIs em órgãos de investigação que detalhem todas as etapas do processo, ferramentas e metodologia utilizadas (documentos dessa natureza normalmente não são públicos), foram encontrados estudos que envolvem partes do objetivo proposto detalhado no Capítulo 3.

Diversos trabalhos sobre caracterização de tráfego são encontrados na literatura, tais como Dianotti et al. (2009), Hyun-Chul et al. (2008) e Ponec et al. (2007), que podem auxiliar na redução do tráfego, essencial quando o volume do tráfego for elevado e que o impeça de ser analisado em sua totalidade. São encontrados também diversos trabalhos para caracterização de tráfego específico, tais como P2P (*Peer-to-Peer*) em Constantinou e Mavrommatis (2006), VoIP (*Voice over IP*) em Li et al. (2010) e SMTP (*Simple Mail Transfer Protocol*) em WenQi e WeiGuang (2009).

Han et al. (2002) e Fusco e Deri (2010) apresentam arquiteturas de sistemas para captura e monitoramento de redes de alta velocidade (acima de 1Gbps), demonstrando as limitações das ferramentas de captura nessas situações.

Cohen (2008) apresenta PyFlag, uma ferramenta *open source* com múltiplas funções forenses, entre elas a análise de tráfego de rede. Seu uso é voltado para o especialista de informática forense e análise *off-line* de tráfego (tráfego previamente capturado).

Esses três grupos de trabalhos contribuíram principalmente no desenvolvimento de módulos de decodificação de tráfego, infraestrutura de recebimento de tráfego e de apresentação do tráfego decodificado, respectivamente.

1.6 ORGANIZAÇÃO DO TEXTO

Os demais capítulos estão organizados da seguinte forma: no Capítulo 2 é apresentada a Interceptação Telemática nos seus aspectos gerais e os aspectos específicos da sua aplicação pelos órgãos de investigação no Brasil; no Capítulo 3 é detalhada a proposta deste trabalho; no Capítulo 4 são demonstrados os experimentos realizados e a análise dos resultados obtidos e no Capítulo 5 são apresentadas as conclusões e os trabalhos futuros.

2 INTERCEPTAÇÃO DE CONEXÃO À INTERNET

Neste capítulo são apresentados inicialmente os conceitos da interceptação legal e sua contextualização na Informática Forense e Ciências Forenses, os aspectos jurídicos envolvidos e padronização por órgãos de normatização. Então são apresentados o processo de captura e disponibilização do tráfego interceptado pelas operadoras, os usuários da interceptação, a infraestrutura utilizada para recebimento e análise de tráfego e a metodologia de análise e de investigação aplicada. Por fim são apresentados o funcionamento da interceptação telefônica e os desafios enfrentados pelos OICBs para diminuir a complexidade das interceptações telemáticas para aumentar seu uso nas investigações.

2.1 INTERCEPTAÇÃO LEGAL NAS CIÊNCIAS FORENSES

Interceptação legal é o processo de interceptar comunicações entre investigados de agências da lei em uma rede, autorizadas por juiz competente e sem que os investigados tenham conhecimento (Branch et al., 2004). Pode ser de ligações telefônicas (interceptação telefônica) e de comunicação de dados (interceptação telemática). Segundo o ITU-T (2008b) são requisitos gerais da interceptação legal: o sistema de interceptação deve prover interceptação transparente apenas dos dados solicitados, o sujeito interceptado não deve perceber a interceptação e o serviço provido para os não envolvidos não deve ser afetado.

A interceptação telemática é estudada pela Forense de Rede (do termo inglês *Network Forensics*), que, segundo Pilli (2010), é a ciência que lida com captura, registro e análise de tráfego de rede interceptado para detecção de intrusões e sua investigação. Palmer (2001) adota uma definição mais abrangente, indicando a Forense de Rede como um sub-ramo da Forense Digital relativo ao monitoramento e análise de tráfego de rede de computadores para fins de coleta de informações, provas legais ou de detecção de intrusão. Dentre as definições da Forense de Rede, as mais abrangentes, ou seja, as que definem que a busca de informações de interesse forense parecem ser mais adequadas, já que no tráfego de rede pode haver vestígios de todo o tipo de crime e não apenas atos delituosos contra sistemas computacionais.

A Forense Digital, também chamada de Informática Forense ou Computação Forense, é uma Ciência Forense que lida com a identificação, preservação, análise e apresentação de vestígios digitais de forma legalmente aceitável através da aplicação de tecnologia para investigação de crimes (McKemmish, 1999), sendo que vestígio digital pode ser definido como qualquer informação com valor probatório armazenado ou transmitido de forma digital (Huebner et al., 2003).

Ciência Forense é "o uso da ciência e tecnologia para investigar e estabelecer fatos em cortes de justiça criminal ou civil" de acordo com o *American Heritage Dictionary of the English Language*.

2.2 INTERCEPTAÇÃO TELEFÔNICA *VERSUS* INTERCEPTAÇÃO TELEMÁTICA

A interceptação telefônica nos OICBs já se encontra bastante sedimentada, sendo utilizada como técnica de investigação em diversos trabalhos do dia-a-dia destes órgãos.

Os sistemas mais utilizados são o Guardião da empresa Dígitro[6] e o Sombra da empresa Federal Tecnologia[7]. São sistemas compostos por servidor com armazenamento, *links* de voz (responsáveis por receber o conteúdo das ligações) e de dados (responsáveis pelo recebimento das informações das ligações tais como horário de início, duração, número discador e número discado) e estações de trabalho chamadas de PA (Ponto de Análise). Trata-se de sistemas passivos, ou seja, só armazenam informações que chegam aos seus *links*, que são encaminhadas mediante autorização judicial pelas operadoras, não interagindo ativamente com os equipamentos de telefonia. Os sistemas possuem basicamente:

- Módulo de gerenciamento: permite o cadastramento de operações, telefones interceptados e seu respectivo canal de desvio e os usuários;
- Módulo de análise: permite que o Investigador escute as ligações, classifique-as conforme sua importância para a investigação (novas chamadas são classificadas como "não analisada") e faça relato das conversas. Permite também a exportação

[6] Disponível em http://www.digitro.com/
[7] Disponível em http://www.federaltecnologia.com/

de informações para relatório com *links* para arquivos de áudio. O acompanhamento das ligações é feito em tempo real e em ordem cronológica.

Para ICIs não existe uma padronização de infraestrutura definida. Cada unidade descentralizada monta sua própria solução de *hardware*, *software* e *links* conforme suas necessidades imediatas.

Conforme os dados apresentados na Introdução deste trabalho, embora as conexões de Internet representem mais de 14% dos terminais disponíveis de telefonia e Internet, menos de 2% das interceptações legais são telemáticas. Levantamentos informais realizados com Investigadores em OICBs indicam que:

- ICIs são consideradas muito complexas pelos Investigadores;
- A ICI é utilizada apenas em grandes investigações onde recursos financeiros e humanos são abundantes ou em investigações em que nenhuma outra técnica de investigação obteve sucesso;
- ICIs trazem bons resultados paras as investigações quando bem empregadas;
- O uso de interceptações telefônicas nas investigações tem se tornado cada vez menos eficiente devido sua ampla divulgação na mídia, enquanto a Internet é ainda considerada pelos criminosos com meio de comunicação seguro e anônimo.

2.3 ASPECTOS JURÍDICOS

O uso da interceptação legal depende de leis específicas de cada país. Nos Estados Unidos, é prevista por "*Communications Assistance for Law Enforcement Act* (CALEA)", na França por "*Commission Nationale de Controle des Interceptions de Sécurité - La loi 91-636*" e "*Loi sur la Securite Quotidienne*", na Alemanha por "*G-10*" e "*The Counter terrorism Act*", no Reino Unido por "*Regulation of Investigatory Powers Act 2000*" e "*Anti-terrorism, Crime and Security Act 2001"* (Baloo, 2003).

A interceptação telefônica e de dados (telemática) é prevista pela legislação brasileira como meio legal de prova em processos criminais. Era inicialmente regida pelo Código Brasileiro de Comunicações (Lei nº 4.117/1962 – Brasil, 1962), depois pela Constituição de 1988 (Brasil, 1988) e pela Lei da Interceptação (Lei nº 9.296/1996 – Brasil, 1996). Em

2008, o CNJ publicou a Resolução Nº 59/2008 (Brasil, 2008) a fim de disciplinar e uniformizar procedimentos. No momento tramitam no Congresso Nacional projetos modificando a Lei nº 9.296/1996.

2.3.1 Constituição de 1988

No seu artigo 5º (trata dos direitos e garantias fundamentais e direitos e deveres individuais e coletivos) inciso XII, a Constituição da República Federativa do Brasil de 1988 estabelece:

> É inviolável o sigilo da correspondência e das comunicações telegráficas, de dados e das comunicações telefônicas, salvo, no último caso, por ordem judicial, nas hipóteses e na forma que a lei estabelecer para fins de investigação criminal ou instrução processual penal. (Brasil, 1988).

Este inciso concede ao cidadão o direito ao sigilo de suas comunicações telefônicas e de dados, dentre outras, relativisando-o quando seu uso for para fins criminais. O inciso indica ainda que uma lei deve ser editada regulamentando em quais situações esse direito pode ser "quebrado" mediante ordem judicial.

O artigo 5º é uma cláusula pétrea, ou seja, só pode ser modificado com a convocação de uma Assembléia Constituinte.

2.3.2 Lei da Interceptação (Lei nº 9.296, de 1996)

Somente após oito anos da promulgação da Constituição de 1988 (Brasil, 1988) foi regulamentado o seu inciso XII do artigo 5º através da sanção da Lei da Interceptação (lei nº 9296/1996 - Brasil, 1996). A Lei da Interceptação trata da interceptação telefônica e telemática e estabelece, dentre outras providências, que:

- Dependerá de ordem de juiz competente (art. 1º);
- Apenas para crimes com pena superior a detenção (art. 2º inciso III);
- Pode ser requerida por autoridade policial ou ministério público (art. 3º incisos I e II);
- Decisão judicial deve ser fundamentada, ter duração máxima de 15 dias, renováveis (art. 5º);

- Resultado relatado ao juiz em auto circunstanciado (art. 6º §2º);
- Autoridade policial poderá requisitar serviços técnicos especializados às concessionárias de serviço público (art. 7º);
- Constitui crime realizar interceptações sem autorização judicial sujeito à pena de reclusão de dois a quatro anos e multa (art. 10º).

2.3.3 Resolução nº 59/2008-CNJ

Em 2008, o Conselho Nacional de Justiça (CNJ), "órgão voltado à reformulação de quadros e meios no Judiciário, sobretudo no que diz respeito ao controle e à transparência administrativa e processual" (CNJ, 2011), publicou a Resolução nº 59/2008 (Brasil, 2008) que:

> Disciplina e uniformiza as rotinas visando ao aperfeiçoamento do procedimento de interceptação de comunicações telefônicas e de sistemas de informática e telemática nos órgãos jurisdicionais do Poder Judiciário, a que se refere a Lei nº 9.296, de 24 de julho de 1996. (Brasil, 2008).

A Resolução nº 59/2008 trata de aspectos da movimentação de documentos para garantir o sigilo das medidas judiciais e estabelecimento de controles a fim de coibir abusos no uso desse meio de prova.

Em suas disposições transitórias, mais precisamente no art. 20º, estabelece que "O Conselho Nacional de Justiça desenvolverá, conjuntamente com a Agência Nacional de Telecomunicações - ANATEL, estudos para implementar rotinas e procedimentos inteiramente informatizados, assegurando o sigilo e segurança dos sistemas no âmbito do Judiciário e das operadoras" (Brasil, 2008), demonstrando a preocupação do órgão na possibilidade de vazamento de informações de interceptações causada pela circulação de documentos em papel, bem como a agilização do processo através da informatização.

2.3.4 Crimes de Informática

Crimes utilizando tecnologia são conhecidos popularmente como crimes de informática, crimes informáticos ou crimes cibernéticos. Carvalho et al. (2008) apresenta a seguinte classificação de crimes de informática:

1. Crimes de informática mediatos ou indiretos: ocorrem quando o delito-meio informático é usado para sua consumação. Ex.: acesso não autorizado a sistema bancário (inviolabilidade de dados) e a conseqüente transferência de dinheiro para sua conta (furto). Esta categoria difere dos delitos impróprios, pois aqui há um delito-meio, e não somente a informática como meio, o que aconteceria no caso da difamação, por exemplo.
2. Crimes de informática próprios ou puros: só podem ser praticados através da informática, sem a qual a execução e consumação da infração não poderiam ocorrer. Ex.: violação de *e-mail*, vandalismo na *web*, difusão de vírus etc.
3. Crimes de informática impróprios ou comuns: podem ser praticados de qualquer forma, inclusive através da informática (como meio). Ex.: estelionato, calúnia, pedofilia etc.
4. Crimes de informática mistos: abrangem o bem juridicamente protegido, mas sua prática só é possível com a informática. Ex.: obter acesso a sistema de tratamento automático de dados usado pelo serviço eleitoral, para tentar alterar a contagem dos votos. (Carvalho et al., 2008).

No Brasil, ainda não existe legislação específica que tipifica os crimes de informática próprios ou puros, sendo os demais crimes penalizados de acordo com os tipos penais previsto em lei, uma vez que o computador é apenas uma ferramenta meio.

Embora exista o preconceito de que a ICI é útil apenas para investigações de crimes de informática próprios, como pode ser observado nos exemplos apresentados por Carvalho et al. (2008), ela pode ser utilizada em investigações de qualquer tipo de crime onde se utiliza a Internet como meio de comunicação.

2.4 PADRONIZAÇÃO

A padronização da interceptação legal por institutos internacionais é essencial, já que pode ser uma potencial ameaça à segurança e a privacidade dos usuários como também alvo de controles rígidos de governos que podem afetar significativamente no desenvolvimento de novos serviços baseados na Internet (Branch, 2003). O ETSI (*European Telecommunication Standards Institute*) possui o grupo de trabalho específico para padronização da interceptação legal ETSI TC LI (*ETSI Technical Committee on Lawful Interception* – ETSI, 2008), tendo emitido desde sua criação diversos documentos técnicos

sobre o tema. Já o *Internet Engineering Task Force* (IETF) decidiu não suportar uma trilha de trabalho de padrões para essa área conforme publicado na RFC (*Request for Comments*) 2804 (IAB e IESG, 2000).

O Brasil não adota um padrão para a interceptação legal, tornando complexo o uso de ICIs nas investigações principalmente devido as variadas formas de disponibilização do tráfego pelas operadoras. As operadoras provêem acesso à Internet utilizando diversas tecnologias (ex: ADSL - *Asymmetric Digital Subscriber Line*, VDSL - *Very-high-bit-rate Digital Subscriber Line*, *Cable Modem*, GPRS - *General Packet Radio Service*, WCDMA - *Wideband Code Division Multiple Access*, HDSPA - *High-Speed Downlink Packet Access*, EDGE - *Enhanced Data Rates for GSM Evolution*, WiMAX - *Worldwide Interoperability for Microwave Access*, etc.) através de complexa estrutura de rede, o que na prática inviabiliza a captura do tráfego diretamente pelo OICB. Os OICBs então requerem às operadoras que capturem o tráfego da conexão investigada e disponibilizem-no de forma automatizada. As operadoras têm atendido da forma mais simples e menos oneroso, considerando sua tecnologia e seu corpo técnico, utilizando-se da ausência de padrão. Cabe aos OICBs se adequarem as especificidades de cada operadora.

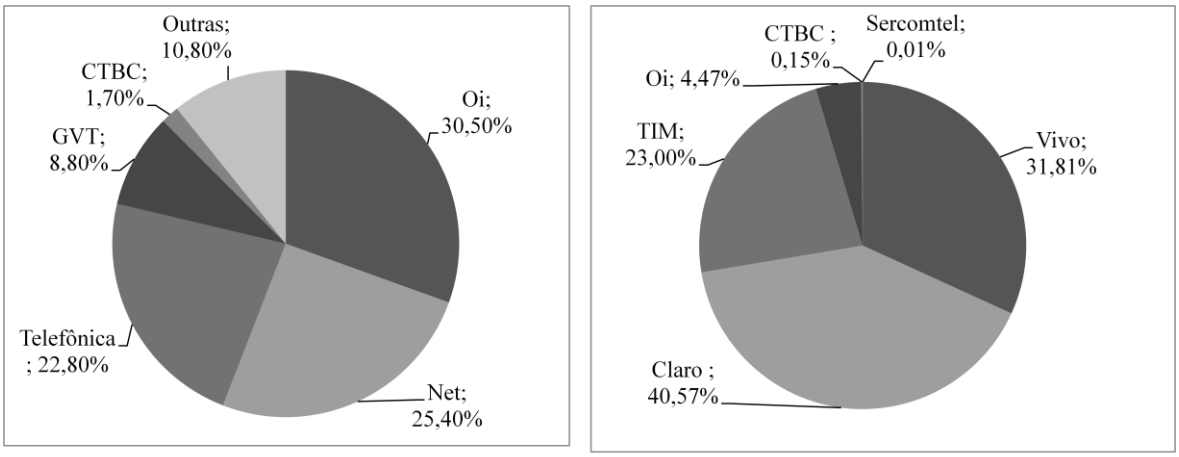
2.5 CAPTURA E DISPONIBILIZAÇÃO DO TRÁFEGO

Com o uso massivo da Internet como meio de comunicação, as Interceptações de Conexões à Internet (ICIs) tem sido a forma mais comum de uso da interceptação telemática nas investigações criminais. O processo é conduzido de forma similar à interceptação telefônica (Broadway et al., 2008): é expedido mandado judicial para o provedor do serviço do investigado para que seja instalada ferramenta para escuta (captura dos pacotes) do tráfego que flui de e para a conexão investigada. A captura pode ser conduzida pela operadora ou pelo órgão de investigação.

O tráfego capturado deve ser disponibilizado para os Investigadores em tempo real, pois vidas podem estar em risco em algumas investigações. O tráfego pode ser entregue em forma de arquivos ou encaminhamento dos pacotes capturados de forma *on-line* (encapsulamento de tráfego). Além da existência de grande variedade de formatos de arquivos de captura (ex: pcap, Snoop, netmon, ncf, etc.), a forma de entrega destes arquivos pode assumir uma infinidade de variações, sendo o uso de servidores de

transferência segura de arquivos o mais coerente. Na forma de encapsulamento de tráfego, os pacotes capturados são encaminhados encapsulados em pacotes IP endereçados para o servidor do órgão de investigação. Dependendo do ponto de captura do tráfego, os pacotes podem ter ainda apresentar uma variedade de camadas de enlace.

Como no Brasil existe uma concentração de clientes de acesso à Internet em grandes operadoras (acima de 87% em banda larga segundo a Teleco (2011a) e acima de 99% em 3G segundo a Teleco (2011b) - Figura 2-1), que já possuem pessoal especializado e método definido de disponibilização do tráfego interceptado, e a maioria das ICIs acabam recaindo sobre investigados clientes destas operadoras, foram estudados em detalhes as formas de entrega de tráfego e formatos de arquivos de captura e pacotes adotados por elas.



a) Banda Larga (Teleco, 2011a) b) 3G (Teleco, 2011b)

Figura 2-1 – Concentração de clientes de acesso à Internet em grandes operadoras

2.5.1 Formas de Entrega do Tráfego

As principais operadoras brasileiras entregam o tráfego capturado em três formas, aqui categorizados como SFTPServer, SFTPClient e encapsulamento de tráfego.

2.5.1.1 SFTPServer

Na forma SFTPServer (Figura 2-2), a operadora fornece em seu servidor de interceptação uma conta de um servidor SFTP (*Secure File Transfer Protocol*) ou SCP (*Secure CoPy*) para cada conexão interceptada, onde ficam disponíveis para *download* arquivos de captura

(arquivos contendo o pacotes de rede coletados) referentes ao tráfego de rede da conexão alvo. Os arquivos de captura são limitados a determinado tamanho (ex: 20MB) e possuem nomenclatura sequencial.

O servidor SFTP é configurado para que o arquivo ativo (arquivo que ainda encontra-se aberto para gravação de novos pacotes coletados e que não alcançou o tamanho limite) possa ser baixado e com função de continuação de *download* ativo, o que permite que a qualquer tempo o Investigador possa acompanhar os últimos passos do investigado.

Como os pacotes são capturados e armazenados dentro da rede da operadora e os arquivos de captura são apagados pelo Investigador apenas depois de copiados, essa é forma mais confiável de entrega (todo o pacote capturado é entregue ao OICB).

A segurança do processo é feita pela operadora permitindo que apenas IPs de um OICB possam se conectar ao servidor e pelo OICB verificando se o certificado apresentado pelo servidor SFTP é da operadora, além, é claro, da autenticação e da transferência de dados criptografados inerentes ao protocolo SFTP.

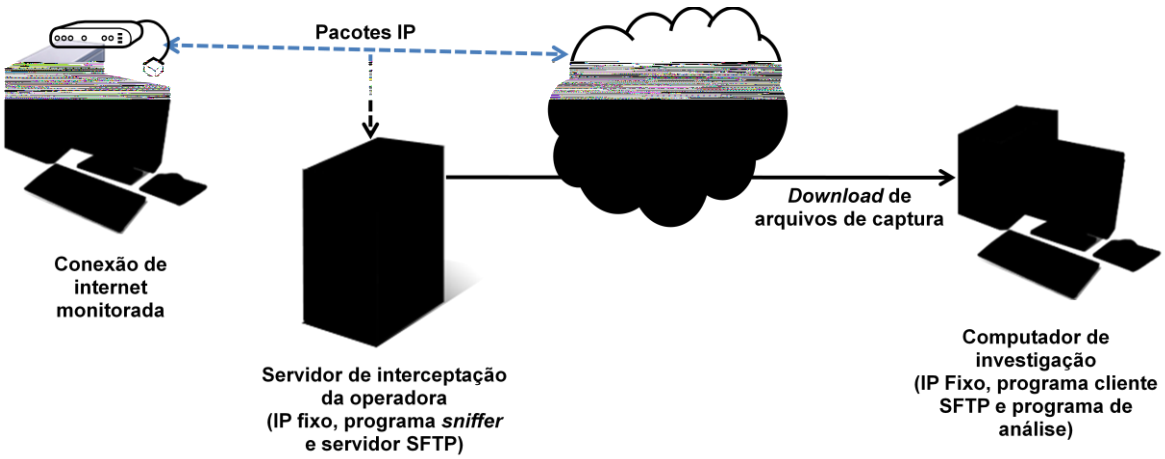


Figura 2-2 – Forma de entrega SFTPServer

2.5.1.2 SFTPClient

Na forma SFTPClient (Figura 2-3), a operadora coleta em seu servidor de interceptação os pacotes de rede da conexão alvo em pequenos arquivos de captura (normalmente 50KB ou

500KB e ainda por tempo limite de 30 segundos) e os envia (*upload*) para uma conta em servidor SFTP disponibilizado pelo OICB.

Arquivos de captura que não puderem ser entregues após determinado número de tentativas são desprezados pela operadora, devendo o OICB utilizar equipamentos e conexões de Internet confiáveis e com banda suficiente sob risco de perda de informações.

A segurança do processo é feita pelo OICB permitindo que apenas IPs de operadoras possam se conectar ao servidor e pela operadora verificando se o certificado apresentado pelo servidor SFTP é de um OICB, além, é claro, da autenticação e da transferência de dados criptografados inerentes ao protocolo SFTP.

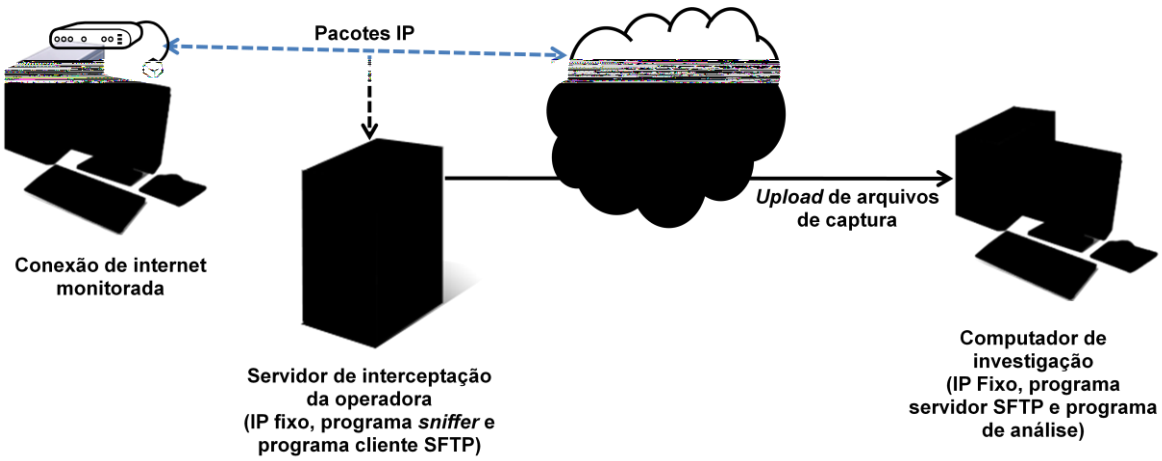


Figura 2-3 – Forma de entrega SFTPClient

2.5.1.3 Encapsulamento de Tráfego

Na forma encapsulamento de tráfego (Figura 2-4), a operadora configura seu servidor de interceptação para gerar cópia de todos os pacotes de rede que trafegam pela conexão alvo e enviá-los para o OICB. O pacote de rede original é encapsulado em um novo pacote de rede que tem como endereço de origem o IP do equipamento da operadora e como endereço de destino o IP do servidor do OICB.

Nesta forma de entrega não há buferização no envio nem confirmação de entrega dos pacotes. Então a premissa básica é que a conexão de Internet do OICB tenha banda

superior à soma das bandas de subida e descida da conexão interceptada, o que mesmo assim não garante o recebimento de todos os pacotes, já que na rota entre o servidor da operadora e o do OICB podem haver congestionamentos.

A segurança é baixa neste processo. Os pacotes são transferidos em claro e por protocolos que não garantem a entrega dos dados. Não há autenticação entre as partes, apenas verificação dos IPs de origem (operadora) e destino (OICB).

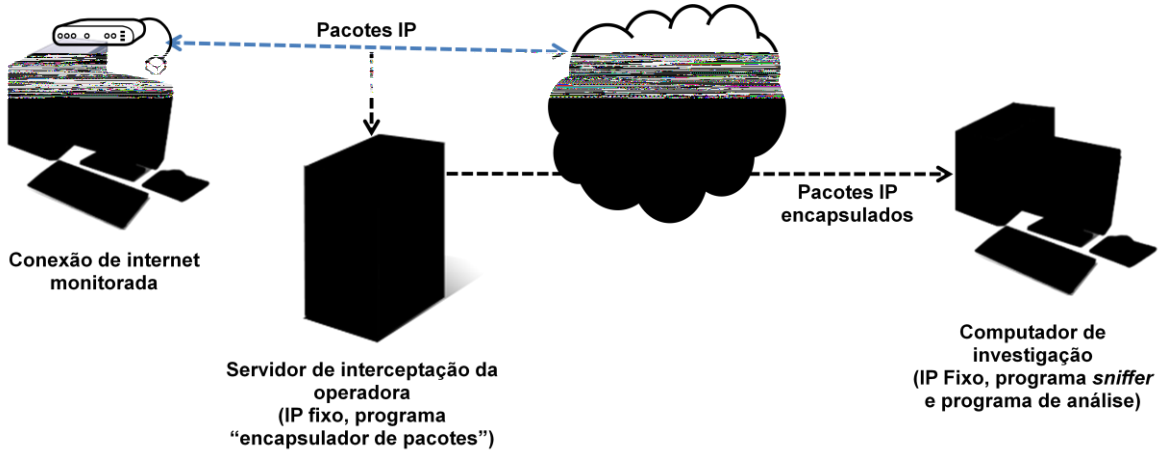


Figura 2-4 – Forma de entrega encapsulamento de tráfego

2.5.2 Formatos de Arquivos de Captura

As operadoras escolhidas para análise neste trabalho, que entregam o tráfego interceptado em forma de arquivo, utilizam os formatos de arquivo de captura pcap, Snoop e ETSI.

Pcap é um formato de arquivo para salvar pacotes de rede capturados que se tornou padrão de fato (Ficara et al., 2008). O arquivo possui um cabeçalho global seguindo por zero ou mais registros de pacotes. Cada registro de pacote possui um cabeçalho e o pacote de rede capturado. Encontra-se documentado em Wireshark Foundation (2011).

Snoop é um formato de arquivo de captura utilizado pelo programa de mesmo nome da Sun. O formato está documentado na RFC 1761 (Callaghan e Gilligan, 2005). Assim como o formato pcap, o Snoop possui um cabeçalho global seguindo por zero ou mais registros de pacotes. Cada registro de pacote possui um cabeçalho e o pacote de rede capturado.

O formato de arquivo de captura padronizado pelo *European Telecommunication Standards Institute* (ETSI) é uma estrutura de dados ASN.1 (*Abstract Syntax Notation One* – ISO/IEC, (2002)) descrita no documento ETSI TS 102 232-3 (ETSI, 2006).

2.5.3 Formatos dos Pacotes

Os pacotes contidos em arquivos de captura entregues pelas operadoras, ou mesmo capturados pelo OICB quando são entregues na forma Encapsulamento de Tráfego, possuem geralmente um ou mais cabeçalhos de enlace. Os cabeçalhos de enlace Ethernet (Postel e Reynolds, 1988), LCC (*Linux Cooked Capture* - Wireshark Foundation, 2010), PPPoE (*Point-to-Point Protocol over Ethernet* - Mamakos et al., 1999) e VLAN (*Virtual Local Area Network* - Jeffree, 2005), bem como uma combinação entre eles, são os normalmente encontrados.

Além do cabeçalho de enlace, algumas operadoras entregam os pacotes IPs interceptados encapsulados dentro de outro pacote IP. Os encapsulamentos IP/GRE (*Generic Routing Encapsulation* - Li et al., 2000), IP/UDP/IP, PCLI (*Packet Cable Lawful Interception* - CableLabs, 2004), Juniper (Juniper Networks, 2010) e TZSP (*TaZmen Sniffer Protocol* - Wikimedia Foundation, 2011) são os utilizados pelas operadoras brasileiras.

Os cabeçalhos de enlace e encapsulamento aqui detalhados não possuem relevância para a investigação, pois são referentes à infraestrutura de captura e disponibilização das operadoras e não à conexão interceptada.

2.6 USUÁRIOS

Com o tráfego coletado, é necessário analisá-lo para extrair informações úteis, o que é realizado por policiais com treinamento limitado em Informática Forense e, nos casos mais complexos, por especialistas experientes em protocolos de comunicações de Internet (Broadway et al., 2008).

Nos OICBs se observam dois principais usuários das ICIs: o Investigador e o Perito.

O Investigador é o policial especializado no crime sob investigação (tráfico de drogas e armas, corrupção, pedofilia, contrabando e descaminho, etc.). Possui conhecimentos básicos de informática. Necessita acompanhar em tempo real as comunicações do investigado (*e-mails*, conversas instantâneas, redes sociais, etc.) a fim de providenciar outras diligências (vigilância, consulta a bancos de dados abertos ou oficiais, etc.) para comprovar o cometimento dos crimes sob investigação ou mesmo evitar o cometimento de outros crimes (homicídios, abusos sexuais, lesões corporais, etc.). Precisa de ferramenta única, simples e intuitiva, de forma semelhante às ferramentas atualmente utilizadas nas interceptações telefônicas.

O Perito é o policial especialista em informática. Sua atuação principal nas ICIs é como consultor dos Investigadores, indicando formas de investigação, ferramentas a serem utilizadas e suas limitações, podendo substituí-los em investigações mais complexas que exijam conhecimento técnico avançado. Também é acionado para auxiliar o juiz através de documento oficial (Laudo) para esclarecer a materialidade do crime e/ou sua autoria com base no tráfego capturado. O Perito possui tempo maior para análise, podendo dispor de diversas ferramentas de análise, a fim de formar sua convicção.

2.7 RECEBIMENTO E ANÁLISE DO TRÁFEGO

A infraestrutura mínima para um OICB realizar ICIs é constituída por computador, programas (sistema operacional e ferramentas para obtenção, tratamento e análise de tráfego) e conexão de Internet com IP fixo (exigência das operadoras para envio do tráfego). As investigações ocorrem, por obrigação legal (art. 4 do Código de Processo Penal – Brasil, 1941), na unidade do OICB responsável pela circunscrição no local do crime.

Nas unidades principais (principais capitais e cidades de interior perto de grandes centros) conseguem-se com certa facilidade computadores confiáveis (servidores) com energia ininterrupta e climatização, programas necessários e conexões de Internet confiáveis com banda suficiente. Já nas unidades remotas (unidades localizadas em cidades de interior longe dos grandes centros e em capitais menores), além da dificuldade na disponibilização de servidores e programas para esse fim, a contratação de conexão de Internet confiável com banda suficiente é inviável ou, às vezes, não disponível. A infraestrutura inadequada

pode causar prejuízos à investigação devido à perda de parte de tráfego interceptado, ou mesmo inviabilizá-la.

2.7.1 Ferramentas de Análise Forense de Rede

Para análise do tráfego, são utilizadas ferramentas de análise forense de rede (NFATs - *Network Forensics Analisys Tools*), que são programas que permitem uma visualização de alto nível de dados coletados (Corey, 2002), focando na análise da camada de aplicação (*e-mails*, páginas *web*, VoIP, comunicadores instantâneos, etc.). São diferentes de ferramentas de segurança e monitoramento de rede (NSM - *Network Security and Monitoring*) tais como Wireshark[8] e Tcpdump[9], que embora possam ajudar na análise forense, são desenvolvidas para serem usadas por especialistas de segurança de rede (Pilli, 2010).

Existe uma variedade de NFATs comerciais (ex: NetworkMiner[10], NetWitness[11], NetResident[12], NetIntercept[13], Iris[14] e OmniPeek[15]) e algumas poucas livres (ex: Xplico[16] e PyFlag[17]) disponíveis. Em análise nas versões adquiridas ou disponíveis para *download* ou de suas documentações quanto a sua aderência as características das ICIs no Brasil, nenhuma delas suporta todos os formatos de arquivos, enlaces e escapsulamentos de pacotes que são disponibilizados pelas operadoras brasileiras. Também não apresentam função de busca de arquivos de captura em servidor SFTP. Cabe observar ainda que a maior parte destas ferramentas é voltada para o uso em segurança de rede e resposta a incidentes, análises estáticas ou para uso de Peritos. Pela facilidade de uso, quantidade de protocolos reconhecidos e baixo custo, a NFAT NetResident tem sido muito utilizada nas OICBs.

As NFATs funcionam bem quando os pacotes possam são capturados diretamente da interface de rede pelas ferramentas ou ainda quando todos os arquivos de captura já estão

[8] Disponível em http://www.wireshark.org/
[9] Disponível em http://www.tcpdump.org/
[10] Disponível em http://www.netresec.com/?page=NetworkMiner
[11] Disponível em http://www.netwitness.com/
[12] Disponível em http://www.tamos.com/products/netresident/
[13] Disponível em http://www.niksun.com/
[14] Disponível em http://www.eeye.com/products/iris-network-traffic-analyzer
[15] Disponível em http://www.wildpackets.com/
[16] Disponível em http://www.xplico.org/
[17] Disponível em http://www.pyflag.net/

disponíveis, sendo que os pacotes IP apresentem somente enlace Ethernet e não possuam encapsulamento IP, além dos investigados utilizarem em suas comunicações protocolos decodificados por elas.

Embora as NFATs anunciem decodificar grande quantidade de protocolos, em uma análise mais detalhada, identifica-se que diversas informações importantes não são identificadas. Entre elas, destacam-se:

- Decodificação parcial de protocolos utilizados por comunicadores instantâneos: geralmente apenas as conversas são extraídas, passando sem análise os arquivos transferidos, lista de contatos e identificação do IP dos interlocutores;
- Não decodificação de protocolos dos comunicadores instantâneos integrados a serviços *web*, tais como as redes sociais Facebook[18] e Orkut[19], os *webmails* Hotmail[20] e Yahoo[21] e os *chats* eBuddy[22] e Meebo[23];
- Decodificação parcial de *e-mails* enviados e recebidos através de *webmails*: apenas os principais serviços internacionais (Hotmail e Yahoo) são decodificados, sendo que os anexos dos *e-mails* não são identificados. Serviços de *webmails* brasileiros não são decodificados.

Outras limitações que dificultam o uso das NFATs disponíveis são:

- Impossibilidade de automatização do processo de importação de arquivos de captura através de *scripts*;
- Ausência de interface para desenvolvimento de módulos externos para decodificação de outros protocolos;
- Ausência ou ineficiência de mecanismos de classificação de relevância, tais como URLs (*Unified Resource Locators*), *e-mails* e usuários com alta ou baixa relevância, auxiliando ao Investigador a reduzir a quantidade itens a serem analisados;
- Ausência ou ineficiência de mecanismos de marcação por parte do Investigador de itens importantes ou não para a investigação ("*bookmarking*"). Esse mecanismo,

[18] Disponível em http://www.facebook.com/
[19] Disponível em http://www.orkut.com/
[20] Disponível em http://www.hotmail.com/
[21] Disponível em http://mail.yahoo.com/
[22] Disponível em http://www.ebuddy.com/
[23] Disponível em http://www.meebo.com/

além de facilitar a análise diferenciando itens analisados dos não analisados, permitiria um mecanismo de exportação em bloco de todos os itens marcados como importantes para relatório;

- Ausência de mecanismos de gerenciamento automático de arquivos de captura e pacotes IP "já importados" / "não importados", ficando a cargo do Investigador essa função através da criação de pastas "não importados" e "já importados" e movimentação de arquivos entre elas. Arquivos de captura importados equivocadamente duas vezes geram informações extraídas duplicadas, o que impede acompanhamento em tempo real de ICIs com arquivos de captura maiores;
- Demora na disponibilização de novas versões quando protocolos proprietários são modificados (principalmente comunicadores instantâneos e *webmails*);
- Dificuldade no gerenciamento de múltiplas ICIs em um mesmo computador.

A não aderência das NFATs às ICIs brasileiras torna muito complexas as investigações utilizando esse artifício. Ferramentas adicionais são necessárias para obtenção e tratamento dos dados. Para obtenção dos dados são utilizados programas servidores SFTP (ex: openSSH[24]), clientes SFTP (ex: WinSCP[25]) e programas de captura (ex: Tcpdump[26], Windump[27] e Wireshark[28]). Para tratamento são utilizados programas de conversão de formatos de arquivos de captura (ex: Editcap[29]) e editores de arquivos de captura (ex: Bittwiste[30]) para remoção de cabeçalhos de enlaces e/ou encapsulamentos.

Mesmo com esse conjunto de ferramentas, não se consegue decodificar tráfego recebido na forma de arquivos de captura ETSI (não foi encontrado ferramenta de conversão para esse formato e operadora diz estar se adaptando para mudar formato de arquivos) e a decodificação parcial de tráfego quando for capturado com encapsulamento IP (ferramentas como a Bittwiste não funcionam adequadamente quando o pacote encapsulador IP encontrar-se fragmentado, problema esse que é minimizado quando a operadora consegue reduzir o MTU – *Max Transfer Unit* - da conexão interceptada).

---

[24] Disponível em http://www.openssh.com/
[25] Disponível em http://WinSCP.net/
[26] Disponível em http://www.tcpdump.org/
[27] Disponível em http://www.winpcap.org/windump/
[28] Disponível em http://www.wireshark.org/
[29] Disponível em http://www.wireshark.org/
[30] Disponível em http://bittwist.sourceforge.net/doc/bittwiste.1.html

2.8 METODOLOGIA DE ANÁLISE

A metodologia de análise de dados digitais sugerida por McKemmish (1999) envolve as fases de Identificação (onde e como os dados estão armazenados), Preservação (dados devem ser tratados de forma menos intrusiva possível), Análise (extração, processamento e interpretação dos dados) e Apresentação (relatório das provas obtidas). Para a análise de tráfego de rede uma possível especialização pode ser a Identificação como a indicação da conexão alvo a ser interceptada; a Preservação como o armazenamento dos pacotes em arquivo de captura em mídia não regravável e cálculo de seus *hashs*; a Análise como o tratamento (conversão de formato de arquivos ou pacotes caso necessário), importação (processamento dos arquivos de captura na NFAT) e interpretação dos vestígios encontrados (páginas *web*, *e-mails*, conversas, arquivos transferidos, etc.); e a Apresentação com o relatório dos indícios/provas encontradas no tráfego.

2.8.1 Metodologia de Investigação

A Metodologia de Investigação utilizando ICI observada em um OICB, que leva em consideração as regras gerais da Forense de Rede e as peculiaridades brasileiras (aspectos jurídicos, aspectos técnicos, infraestrutura, ferramentas e usuários), pode ser dividida basicamente em três fases (Figura 2-5):

1. Procedimentos Iniciais: o processo inicia-se com a emissão por juiz competente de mandado judicial de autorização de afastamento de sigilo telemático com validade de quinze dias. Este mandado deve ser encaminhado à operadora fornecedora da conexão de Internet do investigado juntamente com informações para estabelecimento da comunicação para obtenção do tráfego. Com as informações repassadas pela operadora, são realizadas as configurações dos programas e equipamentos na estação de trabalho de análise;
2. Procedimentos Rotineiros: são iniciados tão logo os dados interceptados comecem a ser entregues pela operadora e são realizados em intervalos de tempos regulares. Conforme análise de risco do investigado, esses procedimentos necessitam serem feitos continuamente, pois o acompanhamento dos passos do investigado em tempo real pode ser crucial para a investigação. Nesta fase são realizadas a obtenção, tratamento, importação e análise do tráfego;

3. Procedimentos Finais: uma vez vencido ou por vencer o mandado judicial, deve-se confeccionar relatório chamado de auto circunstanciado contendo análise das informações obtidas. No relatório deve-se sugerir pela prorrogação ou não do mandado, que será analisada pelo juiz. Juntamente com o relatório, deve-se encaminhar todo o tráfego obtido com o devido tratamento para sua preservação.

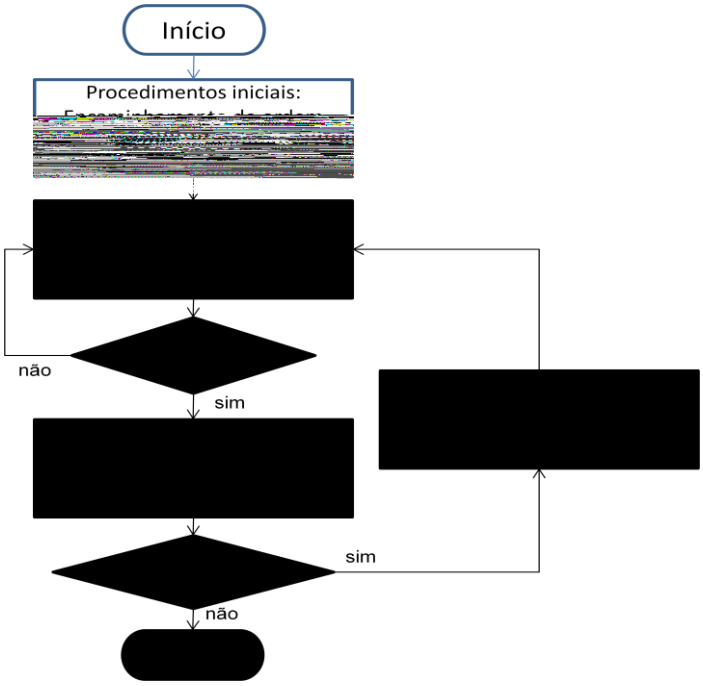


Figura 2-5 – Metodologia de Investigação para Interceptação Telemática

Traçando-se um paralelo com a metodologia sugerida por McKemmish (1999), os Procedimentos Iniciais são a fase de Identificação, os Procedimentos Rotineiros são a fase Análise e os Procedimentos Finais são as fases Apresentação e Preservação.

Como os Procedimentos Iniciais e os Procedimentos Rotineiros variam dependendo da forma de disponibilização do tráfego adotado pela operadora, uma Metodologia de Investigação diferente é adotada para cada situação, conforme resumido na Tabela 2.1. Os Procedimentos Finais não variam.

Tabela 2.1 – Metodologia nas diferentes formas de disponibilização de tráfego

	SFTPServer		**SFTPClient**			**Enc. Tráfego**
Proc. Iniciais	Envio Mandado Config. Remota Config. Local (Cliente SFTP)		Config. Local (Servidor SFTP) Envio Mandado Config. Remota			Envio Mandado Config. Remota Config. Local (sniffer)
Proc. Rotineiros	**Pcap** Obtenção (Cliente SFTP) Importação (NFAT) Análise (NFAT, Word)	**Pcap c/ encap**. Obtenção (Cliente SFTP) Tratamento (bittwiste) Importação (NFAT) Análise (NFAT, Word)	**Pcap** Importação (NFAT) Análise (NFAT, Word)	**Snoop** Tratamento (editcap) Importação (NFAT) Análise (NFAT, Word)	**ETSI** Incompatível	**Pcap c/ encap**. Tratamento (bittwiste) Importação (NFAT) Análise (NFAT, Word)
Proc. Finais	Relatório (NFAT, Word) Preservação					

2.8.1.1 Metodologia de Investigação A: Entrega SFTPServer

Para uma ICI de uma operadora que fornece os arquivos na forma SFTPServer temos os seguintes Procedimentos Iniciais:

1. Envio de mandado: Investigador encaminha à operadora pedido de ICI, acompanhado de mandado judicial;
2. Configuração remota: operadora cria conta SFTP em seu servidor de interceptação; configura programa de captura para armazenar os pacotes que trafegam na conexão solicitada em arquivos de captura de determinado tamanho (20MB normalmente) na pasta *home* da conta SFTP criada; informa ao Investigador dados de acesso aos arquivos (IP/porta/usuário/senha);
3. Configuração local: Investigador cria pastas “não tratados”, “não importados” e “já importados” para armazenamento de arquivos de captura e configura os programas cliente SFTP e NFAT.

Uma vez iniciada a interceptação, os seguintes Procedimentos Rotineiros são realizados:

1. Obtenção: Investigador, utilizando o programa cliente SFTP, faz *download* dos arquivos ainda não baixados do servidor da operadora (a verificação é manual: inspeção visual das pastas “não tratados”, “já importados” e “não importados”), exceto o arquivo de captura aberto (arquivo de captura aberto é o arquivo ainda em uso pelo programa de captura que não alçou o tamanho limite configurado), para a pasta local “não tratados”;

2. Tratamento: fase aplicada apenas se os arquivos de captura ou os pacotes nele contidos sejam incompatíveis com o programa de análise, o Investigador executa programa de conversão de formato ou de extração de cabeçalhos nos arquivos da pasta “não tratados”, gerando os novos arquivos na pasta “não importados”. Caso não seja necessária esta fase, a pasta “não tratados” não é necessária, podendo os arquivos de captura serem baixados diretamente na pasta “não importados”. Os arquivos originais é que devem ser preservados para encaminhamento junto com o relatório de análise;
3. Importação: Investigador, através de comando de importação de arquivos de captura do NFAT, carrega todos os arquivos da pasta “não importados” em ordem cronológica de criação. Depois de carregados, os arquivos são movidos manualmente para a pasta “já importados”;
4. Análise: Investigador visualiza, em ordem cronológica, as informações interpretadas pelo programa de análise (páginas *web* acessadas, *e-mails*, conversas realizadas em comunicadores instantâneos, etc.), e os itens que julgar relevantes para a investigação são copiados para relatório de análise através da área de transferência do sistema operacional; Investigador anota a data/hora do último item analisado para retomar o trabalho quando mais arquivos estiverem disponíveis.

Ao fim do prazo de interceptação, os seguintes Procedimentos Finais são realizados:

1. Relatório: Investigador revisa seu relatório, que é montado durante os dias de validade do mandado, fazendo buscas na ferramenta de análise pelos itens anteriormente analisados a fim de complementar as informações nele contidas. No relatório, o Investigador também conclui pelo pedido de renovação ou interrupção da interceptação;
2. Preservação: Investigador gera mídia não regravável com os arquivos de captura originais e calcula os seus *hashs*, que são listados no relatório. A mídia passa a ser anexo do relatório durante todo o processo legal.

2.8.1.2 Metodologia de Investigação B: Entrega SFTPClient

Para uma ICI de uma operadora que fornece os arquivos na forma SFTPClient temos os seguintes Procedimentos Iniciais:

1. Configuração local: Investigador cria as pastas “não tratados”, “não importados” e “já importados” para armazenamento de arquivos de captura; configura NFAT e cria conta de usuário no programa servidor SFTP instalado no computador de análise, vinculando a pasta *home* dessa conta à pasta “não tratados”;
2. Envio de mandado: Investigador encaminha à operadora pedido de ICI, acompanhado de mandado judicial, informando dados de acesso (IP/porta/usuário/senha) para envio dos arquivos de captura;
3. Configuração remota: operadora, em seu servidor de interceptação, configura programa de captura para armazenar os pacotes que trafegam na conexão solicitada em arquivos de captura de tamanho pequeno (50KB ou 500KB normalmente) em determinada pasta; configura programa que varre esta pasta a cada intervalo de tempo e os envia para a pasta *home* do usuário do servidor SFTP informado pelo Investigador. Depois de enviados, os arquivos são imediatamente apagados do servidor da operadora. O mesmo ocorre com arquivos que não puderam ser enviados após determinadas tentativas sem sucesso por problemas na conexão com o servidor SFTP informado.

Uma vez iniciada a interceptação, os seguintes procedimentos são feitos em intervalos regulares:

1. Obtenção: arquivos de captura já são disponibilizados na pasta “não tratados” ao serem enviados pela operadora;
2. Tratamento: mesmas considerações da Metodologia de Investigação A;
3. Importação: além das considerações da Metodologia de Investigação A, o Investigador deve ter especial atenção, pois no período entre o comando de importação e o processo manual de mover os arquivos, novos dados podem ter sido enviados pela operadora e, portanto, não devem ser movidos para a pasta “já importados”;
4. Análise: mesmas considerações da Metodologia de Investigação A.

Os Procedimentos Finais são os mesmos da Metodologia de Investigação A.

2.8.1.3 Metodologia de Investigação C: Entrega Encapsulamento de Tráfego

Para uma ICI de uma operadora que fornece os arquivos na forma Encapsulamento de Tráfego temos os seguintes Procedimentos Iniciais:

1. Envio de mandado: Investigador encaminha para a operadora pedido de interceptação telemática da conexão de Internet vinculada ao investigado, acompanhado de mandado judicial, informando o IP para envio dos pacotes espelhados;
2. Configuração remota: operadora, em seu servidor de interceptação, configura programa para copiar os pacotes que trafegam na conexão solicitada para envio para o IP informado (o pacote IP copiado é enviado na área de dados de um pacote IP/GRE ou outro encapsulamento onde o endereço de origem é o IP do equipamento da operadora e o endereço de destino é o IP informado pelo Investigador); operadora informa ao Investigador o IP de origem dos pacotes;
3. Configuração local: Investigador cria as pastas “não tratados”, “não importados” e “já importados” para armazenamento de arquivos de captura; configura programa NFAT e configura programa de captura para armazenar os pacotes que tem como IP de origem o IP informado pela operadora em arquivos de captura de determinado tamanho (10MB normalmente) na pasta “não importados”.

Uma vez iniciada a interceptação, os seguintes procedimentos são feitos em intervalos regulares:

1. Obtenção: arquivos de captura já são disponibilizados na pasta “não tratados” ao serem gravados pelo programa de captura;
2. Tratamento: Investigador executa programa de desencapsulamento para retirar os cabeçalhos IP/GRE ou outro encapsulamento. O programa lê os arquivos de captura da pasta “não tratados” (exceto o arquivo de captura ainda não finalizado) e gera novos arquivos na pasta “não importados”;
3. Importação: Investigador, através de comando de importação de arquivos de captura no programa de análise, carrega todos os arquivos da pasta “não importados” em ordem cronológica de criação dos arquivos originais. Os arquivos originais correspondentes aos desencapsulados são movidos manualmente para a pasta “já importados” e os arquivos da pasta “tratados” são removidos;
4. Análise: mesmas considerações da Metodologia de Investigação A.

Os Procedimentos Finais são os mesmos da Metodologia de Investigação A.

2.9 DESAFIOS DO USO DE ICIS NAS INVESTIGAÇÕES

A simplificação da Metodologia de Investigação e o aumento da confiabilidade do processo podem aumentar o uso da ICI nas investigações. Diversas linhas nesse sentido podem ser adotadas, onde se destacam:

- Adoção de normas de padronização unificando a disponibilização do tráfego pelas operadoras: a adoção da entrega SFTPServer, além da simplificação, garantiria a confiabilidade de entrega do tráfego; a adoção de arquivos de captura no formato pcap e pacotes de rede apenas com enlace Ethernet e sem encapsulamento IP além da simplificação, permitiria o uso de uma gama maior de NFATs. Esta linha depende da sensibilização de órgãos externos aos OIBCs, tais como ANATEL, CNJ e Câmara dos Deputados;
- Criação de infraestrutura central nos OICBs com computadores servidores com energia ininterrupta e climatização e conexões de Internet de qualidade com banda suficiente, administrada por equipe de TI, para recebimento do tráfego: simplificaria o processo, já que as funções mais complexas (configuração de *firewall*, programas de captura e servidor SFTP e solução de problemas junto às operadoras) ficariam a cargo de equipe de TI e ocorreria a padronização da entrega do tráfego para os Investigadores; e melhoraria a confiabilidade na entrega do tráfego, pois uma infraestrutura de melhor qualidade diminui o tempo de indisponibilidade dos sistemas, evitando perda de tráfego. Os OICBs já possuem internamente unidades de TI para administração de seus sistemas e redes internas, e uma infraestrutura escalável bem definida não traria grandes impactos no orçamento destas unidades;
- Desenvolvimento de uma NFAT compatível com as formas de entregas, formatos de arquivos e de pacotes disponibilizados pelas operadoras brasileiras e que sane as principais deficiências apontadas nestas ferramentas: simplificaria o processo, pois eliminaria ou automatizaria as etapas de obtenção, tratamento e importação do tráfego evitando o uso de diversas ferramentas e/ou controles manuais; aumentaria a abrangência da análise, pois seriam tratados arquivos de captura no formato ETSI e pacotes com encapsulamento IP fragmentados, atualmente não tratados ou

parcialmente tratados; aumentaria a quantidade de vestígios analisados, pois permitiria o desenvolvimento de rotinas que extraiam do tráfego informações de serviços muito utilizados pelos brasileiros (principalmente *webmails* brasileiros e *chats* via *web*), atualmente não extraídas pelas ferramentas disponíveis. A maioria dos OICBs não tem capacidade técnica para desenvolvimento de ferramenta dessa complexidade ou mesmo de sua especificação para contratação de desenvolvimento junto à iniciativa privada, sendo a parceria com universidades um caminho viável.

3 SOLUÇÃO PROPOSTA: SIT, CLIT E METODOLOGIA

Neste capítulo é apresentada a proposta desta dissertação. A solução envolve dois sistemas: infraestrutura para recebimento dos dados (centralizada) e ferramenta para obtenção e análise dos dados (descentralizada), e uma nova Metodologia de Investigação com o uso dos mesmos. Os sistemas podem funcionar de forma independente, podendo a infraestrutura central ser utilizada em conjunto com outras ferramentas de análise de dados e a ferramenta descentralizada pode funcionar sem uma infraestrutura central de recebimento de dados.

A divisão da solução em duas partes, uma centralizada e outra descentralizada, se deve a própria organização dos OICBs, que possuem órgão central, localizado na capital federal ou capitais estaduais, responsável pelas funções técnicas, administrativas e logísticas da instituição, e unidades descentralizadas, espalhadas pelo país ou estados, responsáveis pelas funções executivas, entre elas, a investigação de crimes na sua área de circunscrição.

Basicamente a infraestrutura de TI de um OICB é composta por um *datacenter* localizado no órgão central, abrigado em uma sala cofre com condições de energia e refrigeração adequadas, além de pessoal qualificado. É interligado com as unidades descentralizadas através de uma rede *Wide Area Network* - WAN (intranet) para acesso a sistemas corporativos e compartilhamento de recursos, permitindo acesso controlado à Internet através de conexões confiáveis (redundância, garantia de banda, alta disponibilidade, etc.), conforme Figura 3-1.

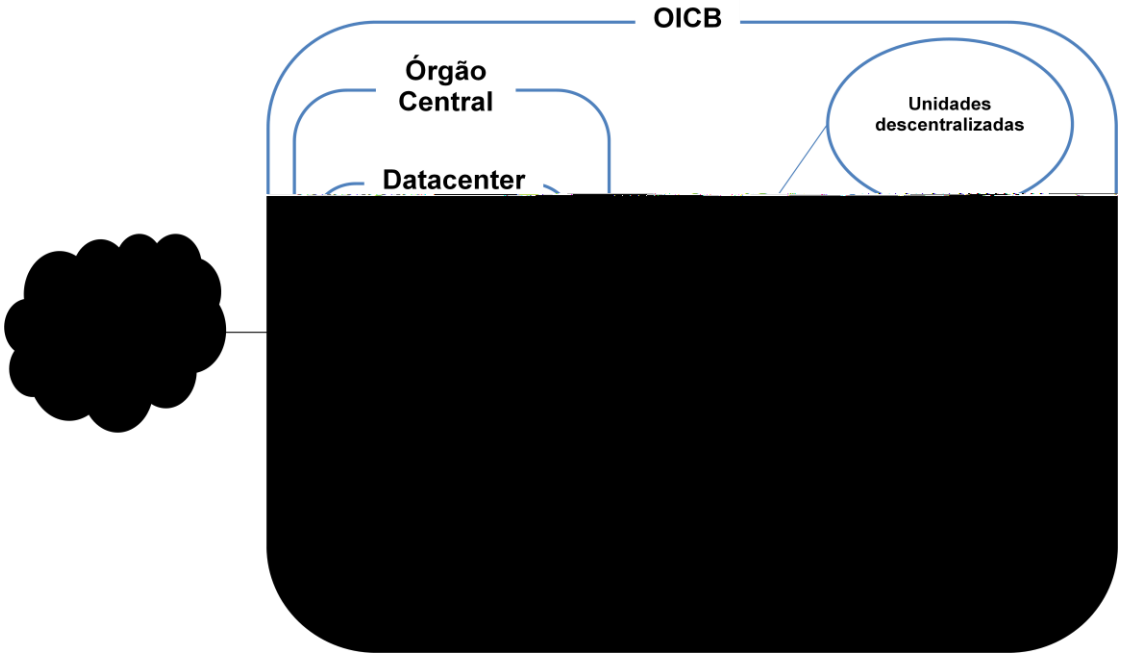


Figura 3-1 – Organização de um OICB e infraestrutura de TI

3.1 SIT - INFRAESTRUTURA PARA RECEBIMENTO DOS DADOS

A infraestrutura para recebimento dos dados, aqui chamada de Servidor de Interceptação Telemática (SIT), é composta basicamente por servidor com armazenamento, conexão à Internet, faixa de IPs fixos, sistema operacional, *software* servidor SFTP e programa de captura de pacotes (Figura 3-2).

O SIT deve ser instalado no *datacenter* do OICB, sendo visível através da Internet para recebimento do tráfego e da intranet do órgão para entrega do tráfego às unidades descentralizadas.

A fim de minimizar a perda de pacotes pelo programa de captura, é indicado o uso de, pelo menos, três interfaces de rede (NICs – *Network Interface Cards*): a primeira (NIC1) configurada com um IP público respondendo a conexões das operadoras através da Internet ao servidor SFTP, a segunda (NIC2) configurada com um IP privado da intranet respondendo a conexões da intranet ao servidor SFTP e a terceira (NIC3) respondendo por uma faixa de IPs púbicos que terão como destino pacotes “espelhados” de uma conexão alvo enviados pelas operadoras.

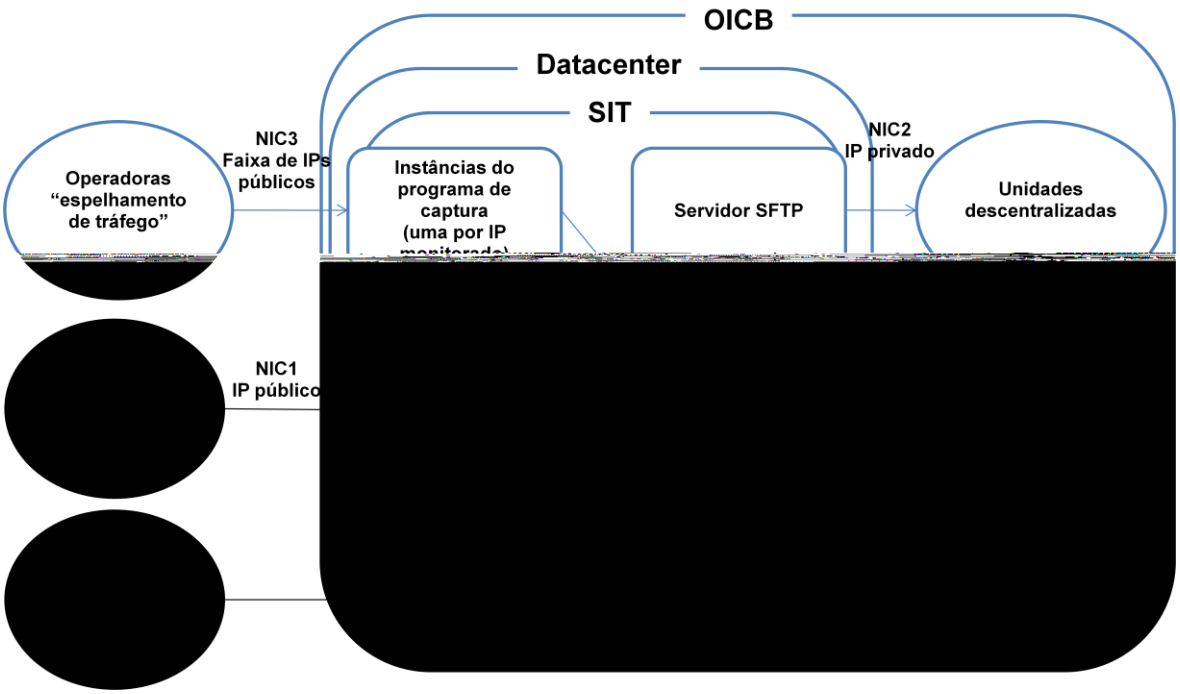


Figura 3-2 – Servidor de Interceptação Telemática (SIT)

Como o SIT envolve o uso de programas de uso geral (Sistema Operacional - SO, servidor SFTP e programa de captura), ele pode ser implementado na plataforma de SO escolhida pelo OICB. Uma possível implementação utilizando apenas *software* livre é o uso do Sistema Operacional Linux[31], Servidor SFTP do pacote OpenSSH[32] e programa de captura Tcpdump[33].

O SIT deve ser administrado por equipe gestora designada e formada por técnicos em TI, que atende pedidos dos responsáveis pelas investigações das unidades descentralizadas.

Caso a operadora indicada entregue os dados na forma SFTPClient, a equipe gestora cria uma conta SFTP no SIT e responde o pedido com as seguintes informações: IP e porta do servidor SFTP, usuário e senha da conta criada. O responsável pela investigação comunica a operadora da autorização judicial e os dados da conta SFTP para envio dos dados. A

[31] Disponível em http://www.linuxfoundation.org/
[32] Disponível em http://www.openssh.com/
[33] Disponível em http://www.tcpdump.org/

mesma conta SFTP será utilizada para obtenção dos arquivos de captura pelos Investigadores.

Caso a operadora indicada entregue os dados na forma Encapsulamento de Tráfego, a equipe gestora cria uma conta SFTP no SIT, reserva um endereço IP da faixa de IPs públicos e ativa um processo *sniffer* que captura todo o tráfego que tem como IP destino o IP reservado gravando-o em arquivos de captura na pasta *home* da conta SFTP criada. A equipe gestora responde o pedido com as seguintes informações: IP destino da interceptação e os dados da conta SFTP criada (IP e porta do servidor SFTP, usuário e senha). O responsável pela investigação comunica a operadora do mandado e o IP para envio dos dados. A conta SFTP será utilizada para obtenção dos arquivos de captura pelos Investigadores.

No caso das operadoras que entregam os dados na forma SFTPServer não há a necessidade do uso do SIT, já que a obtenção dos arquivos de captura pode ser realizada diretamente no servidor da operadora.

Além de cumprir seus objetivos de unificar a entrega de dados interceptados para a equipe de investigação nas unidades descentralizadas (entrega SFTPServer) de forma a padronizar os procedimentos realizados pelo Investigado e aumentar a confiabilidade no recebimento do tráfego (uso de computadores e conexões de Internet confiáveis com infraestrutura de energia e climatização), uma infraestrutura centralizada como esta proposta apresenta como benefícios:

- Aproveitamento da infraestrutura centralizada (*datacenter* com refrigeração, energia, segurança, equipe especializada, etc.) e descentralizada (estações de trabalho) e sua interligação (intranet), necessitando apenas de investimento em servidor e conexão de Internet no *datacenter*, evitando investimentos em diversas unidades descentralizadas;
- Menores custos no investimento com conexão de Internet, já que o preço tende a ser mais baixo nas capitais, além de disponibilidade de largura de banda maiores;
- Parte mais complexa da interceptação (instalação e configuração de programa servidor SFTP e de captura de pacotes, configurações de rede e validação da comunicação com as operadoras) passa a ser executada por técnicos em TI.

3.2 CLIT - FERRAMENTA PARA OBTENÇÃO E ANÁLISE DOS DADOS

A ferramenta para obtenção e análise de dados, aqui chamada de Cliente de Interceptação Telemática (CLIT), diferentemente do SIT, que envolve integração de *hardware* e *software* livre disponíveis, foi quase que totalmente desenvolvida.

O sistema CLIT desenvolvido é composto pelos programas Importador e Analisador e integra a solução proposta conforme Figura 3-3.

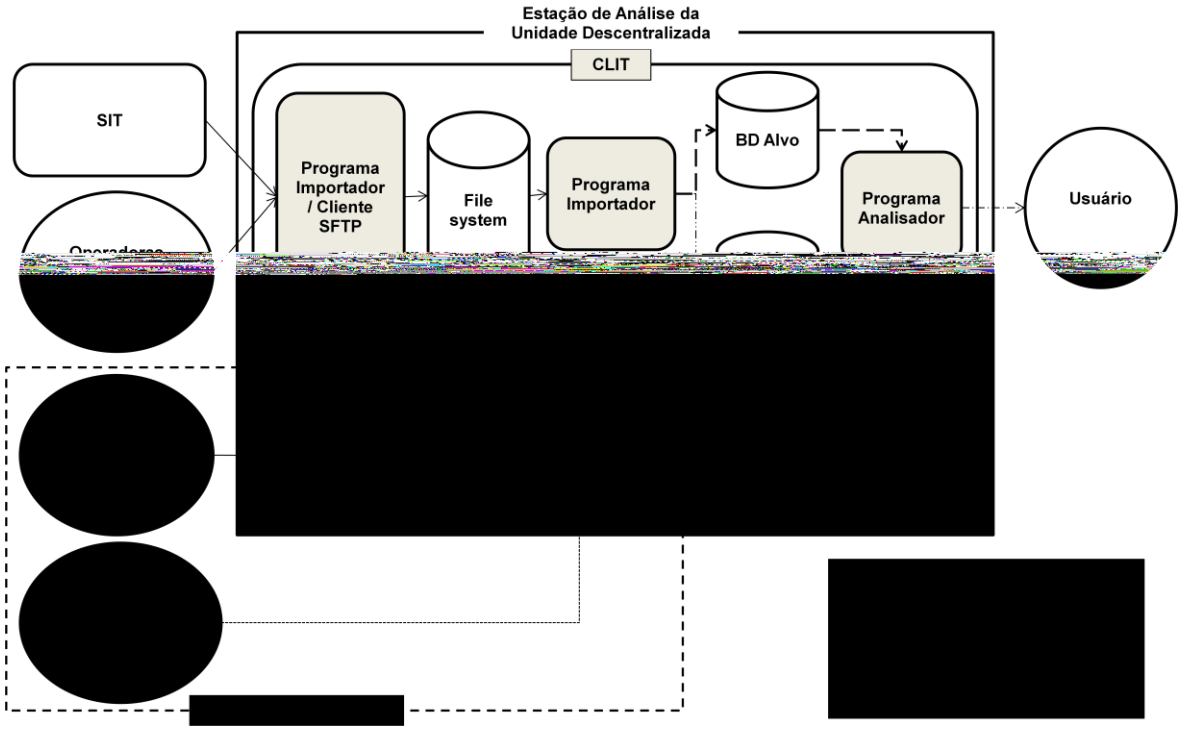


Figura 3-3 – Sistema Cliente de Interceptação Telemática (CLIT)

O programa Importador é o responsável pela obtenção e tratamento dos arquivos de captura e alimentação de um banco de dados da conexão interceptada, que uma vez configurado para determinada interceptação, funciona de forma totalmente automática.

O programa Analisador é o programa que permite ao Investigador abrir o banco de dados gerado pelo programa Importador e a visualização e classificação das informações interpretadas.

Para definição dos requisitos do programa Importador, foram levados em consideração a análise das formas de entrega, formatos e encapsulamentos dos arquivos de captura, a forma de trabalho atual dos analistas nas interceptações telefônicas, as dificuldades levantadas nas atuais interceptações telemáticas e dos objetivos propostos (aumento da abrangência do tráfego processado, automatização de processo manuais, eliminação de controles manuais, redução na quantidade de programas utilizados, acompanhamento em tempo real do tráfego do investigado, padronização de procedimentos realizados pelo Investigador e decodificação de aplicações/protocolos utilizados pelos internautas brasileiros), os quais são os seguintes:

- Permitir a análise separada de várias conexões de Internet interceptadas em um mesmo computador;
- Ter um módulo cliente SFTP integrado, que deve baixar os arquivos de captura do servidor da operadora ou do SIT ainda não baixados, e com função de continuação de *download* de arquivos parcialmente baixados;
- Ter como entrada uma pasta de arquivos de captura, devendo processar todos os arquivos nela existentes de tempos em tempos em ordem cronológica, mantendo registro dos arquivos já processados e arquivos processados parcialmente;
- Ser independente do SIT, permitindo seu uso integrado com um programa de captura gravando os pacotes de rede em arquivos de captura, com um servidor SFTP que recebe os arquivos de captura da operadora ou ainda alimentado manualmente com arquivos de captura na sua pasta de leitura;
- Reconhecer os formatos de arquivo de captura pcap, Snoop e ETSI;
- Reconhecer os pacotes IPs encapsulados com enlaces Ethernet, LCC, PPPoE e VLAN e combinações entre eles;
- Reconhecer os pacotes IPs encapsulados em pacotes de rede IP/UDP, Juniper, PCLI e TZSP e IP/GRE tratando adequadamente a fragmentação gerada;
- Permitir a análise do tráfego interceptado em tempo real;
- Reconhecer fluxos TCP (*Transmission Control Protocol*) e UDP (*User Datagram Protocol*) fragmentados em diversos arquivos de captura;
- Possuir tanto interface gráfica quanto por linha de comando;
- Permitir a integração com módulos externos de reconhecimento de protocolos de aplicação (Filtros) provendo uma interface de fácil leitura dos fluxos TCP e UDP.

Os requisitos definidos para os módulos Filtros são:

- Gerar um ou mais objetos (registros no banco de dados de informação interceptada reconhecida) apontando para arquivos a serem gerados no sistema de arquivos, prevendo sua exibição em navegador *web*, quando reconhecer o seu protocolo a partir de um fluxo;
- Incorporar, quando possível, parametrização para classificação automática de relevância para análise dos objetos.

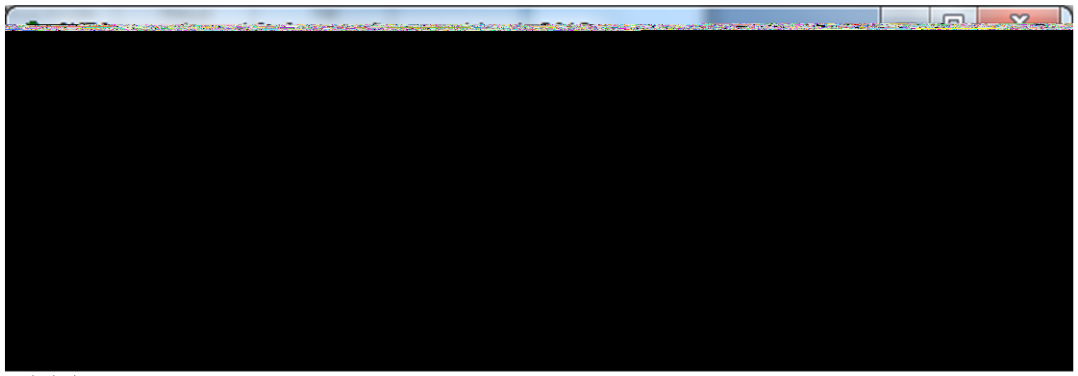
Os requisitos definidos para o programa Analisador são:

- Abrir banco de dados de conexão interceptada sem necessidade de interrupção da atividade do módulo de importação;
- Ter interface gráfica simples, nos moldes das ferramentas de interceptação telefônica, que exiba tabela com os metadados dos objetos reconhecidos e exiba o conteúdo do objeto selecionado em uma interface *web*;
- Possibilitar classificação quanto a relevância para a investigação e de inserção de comentário ao objeto selecionado;
- Possibilitar análise em ordem cronológica com fácil identificação do ponto em que foi interrompida a análise anterior;
- Ter, pelo menos, possibilidade de filtro pelos metadados dos objetos;
- Ter formas de exportação de um ou mais objetos selecionados para elaboração de relatório de análise.

3.2.1 Programa Importador

O programa Importador é o responsável por obter os arquivos de captura e extrair as informações de interesse da investigação (páginas acessadas, *e-mail*, conversas, etc.), aqui chamados de objetos. Tem como entrada a pasta dos arquivos de captura e como saída um banco de dados e arquivos organizados no sistema de arquivos sob a pasta “dados”. Trata-se de programa que pode ser executado com interface gráfica (Figura 3-4-a) ou linha de comando (Figura 3-4-b), facilitando, no último caso, seu uso em *scripts* ou sua execução automática através de um agendador de tarefas.

a)



b)

```
:inicio
C:\intercept\importador.exe c:\intercept\OpX\Alvo1\pcaps  c:\intercept\OpX\Alvo1\Alvo1.mdb -auto -now
C:\intercept\importador.exe c:\intercept\OpX\Alvo2\pcaps  c:\intercept\OpX\Alvo2\Alvo2.mdb -auto -now
C:\intercept\importador.exe c:\intercept\OpX\Alvo2\pcaps  c:\intercept\OpX\Alvo2\Alvo2.mdb -auto -now
Goto inicio
```

Figura 3-4 – Importador: interface gráfica e linha de comando

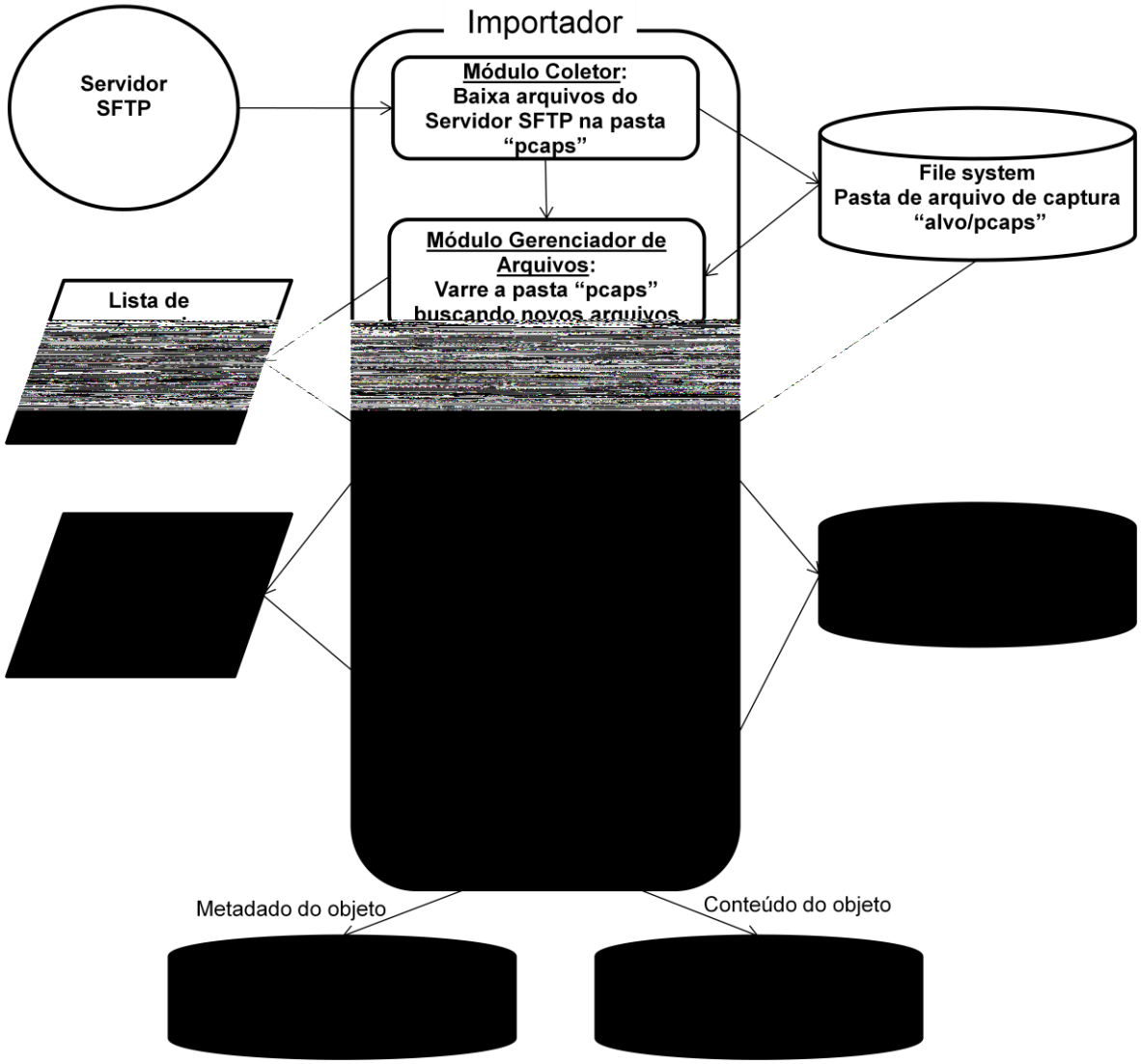


Figura 3-5 – CLIT: Módulos do programa Importador

Possui os seguintes módulos (Figura 3-5):

1. Coletor: obtêm os arquivos de captura de um servidor SFTP;
2. Gerenciador de Arquivos: mantêm informações de quais arquivos de captura já foram processados, parcialmente processados ou não processados;
3. Extrator de Streams: extrai os fluxos TCP e UDP dos arquivos de captura gerando estruturas mais simples de serem processadas, chamadas de "Arquivos de Fluxo" (AFs);
4. Gerenciador de Filtros: entrega os AFs para os módulos filtros configurados;
5. Filtro: módulo interno ou externo que identifica determinado tipo de protocolo de aplicação (ex: HTTP, SMTP, POP, etc.) a partir de um AF, gerando como saída objetos com visualização de alto nível para o usuário.

O fluxo do programa Importador segue o algoritmo apresentado na Figura 3-6.

```
Programa Importador(pastaPcaps, arqMDB);
  // O módulo Coletor, caso configurado, baixa os arquivos de captura
  // de um servidor SFTP para a pasta local "pastaPcaps"
  Coletor(pastaPcaps);

  // O módulo Gerenciador de Arquivos varre a pasta "pastaPcaps"
  // e compara com as informações contidas em tabela no BD,
  // gerando lista de arquivos de captura pendentes de processamento
  listaArqsPcaps := GerenciadorDeArquivos(pastaPcaps, arqMDB);

  Para Cada ArquivoPcap em listaArqsPcaps Faça

    // O módulo Extrator de Streams lê o arquivo captura,
    // agrupa os payloads do TCP/UDP de um mesmo fluxo em
    // estrutura "Arquivo de Fluxo"
    listaArqsStreams := ExtratorDeStreams(ArquivoPcap);

    // O módulo Gerenciador de Filtros possui os módulos Filtros
    // pré configurados
    listaFiltros := GerenciadorDeFiltros.RetornaFiltrosConfigurados;

    // Cada ArquivoDeStream criado ou alterado pelo módulo Extrator de
    // Streams é processado para cada módulo Filtro configurado
    Para Cada ArquivoDeStream em listaArqsStreams Faça
      Para Cada Filtro em listaFiltros Faça

        // Um módulo Filtro lê um Arquivo de Fluxo e, caso reconhecido,
        // gera objetos de análise no BD e no filesystem
        GerenciadorDeFiltro.Processa(Filtro, ArquivoDeStream, arqMDB);

      FimFaça
    FimFaça
  FimFaça
FimPrograma
```

Figura 3-6 – Algoritmo do programa Importador

3.2.1.1 Módulo Coletor

O módulo Coletor é responsável pelo *download* dos arquivos de captura disponíveis no SIT ou no servidor de interceptação da operadora. Trata-se de um cliente SFTP que, ao contrário de implementá-lo, foi desenvolvida rotina para chamar a ferramenta *open source* WinSCP[34] através de sua interface de linha de comando. Sua execução é opcional, já que os arquivos de captura podem estar disponíveis no sistema de arquivos, sendo ativado apenas caso exista o arquivo com o nome ".script.txt" na pasta indicada como fonte dos arquivos de captura para o programa Importador. O arquivo ".script.txt" deve ter como conteúdo *script* compatível com o programa WinSCP conforme o modelo da Figura 3-7-a, sendo que [login], [senha], [ip], [porta] e [pastaRemota] devem ser substituídos pelas informações do servidor SFTP onde se encontram os arquivos de captura, [pastaLocal] pelo nome da pasta onde os arquivos devem ser copiados localmente e [arquivosCaptura] com o padrão de nomenclatura dos arquivos de captura utilizando-se do caractere curinga "*", conforme exemplo apresentado na Figura 3-7-b.

a) Modelo

```
open sftp://[login]:[senha]@[ip]:[porta]/[pastaRemota]
lcd [pastaLocal]
get [arquivosCaptura]
bye
```

b) Exemplo

```
open sftp://alvo1:ju5p$9TR@10.61.76.111:443/upload
lcd c:\intercept\opAguia\Jose\pcaps
get *.cap
bye
```

Figura 3-7 – Modelo e exemplo de arquivo de *script* para *download* automático

3.2.1.2 Módulo Gerenciador de Arquivos

O módulo Gerenciador de Arquivos é responsável por identificar os arquivos de captura a serem processados. Ao ser chamado, varre a pasta de arquivos de captura buscando

[34] Disponível em http://WinSCP.net

identificar novos arquivos ou arquivos modificados desde a sua última execução. Esses arquivos identificados são entregues em ordem cronológica ao módulo Extração de Streams.

Este módulo faz uso da tabela “PCAPSIMPORTADOS” no banco de dados da interceptação para marcar quais arquivos já foram processados e os que tiveram seu tamanho alterado desde o último processamento e, portanto, necessitam ser processados a partir do último pacote lido anteriormente.

A estrutura da tabela “PCAPSIMPORTADOS” é apresentada na Tabela 3.1.

Tabela 3.1 – Campos da tabela “PCAPSIMPORTADOS”

Ordem	**Campo**	**Descrição**
1	ARQUIVO	Nome do arquivo de captura
2	DATAALT	Data/hora da última alteração do arquivo
3	TAMANHO	Tamanho em *bytes* do arquivo
4	PENDENTE	*Flag* indicando se arquivo está pendente de processamento (valor “S) ou já foi processado (valor “N”)
5	PACOTESLIDOS	Número do último pacote processado

O módulo Gerenciador de Arquivos obedece ao algoritmo descrito na Figura 3-8.

3.2.1.3 Módulo Extrator de Streams

O módulo Extrator de Streams é responsável por ler os arquivos de captura e agrupar os pacotes dos fluxos TCP e UDP em estruturas do tipo Arquivos de Fluxo (AFs), sendo que um mesmo fluxo pode estar espalhado por diversos arquivos de captura. Todos os AFs criados e/ou alterados durante o processamento de um arquivo de captura são entregues ao módulo Gerenciador de Filtros.

```
Rotina GerenciadorDeArquivos(Pasta, BD)
  Para cada Arquivo em Pasta Faça
    Tabela := BD.Open(SELECT * FROM PCAPIMPORTADOS
                      WHERE ARQUIVO = Arquivo.nome)
    Se Tabela vazia então
      // encontrado novo arquivo na pasta
      // insere no BD como pendente de processamento
      BD.Exec(INSERT INTO PCAPSIMPORTADOS
           (ARQUIVO, DATAALT e TAMANHO, PENDENTE e PACOTESLIDOS)
            VALUES (Arquivo.nome, Arquivo.dataUltAlt, Arquivo.tamanho,
                    "S", "0") )
    Senão
      Se Tabela.TAMANHO <> Arquivo.tamanho então
        // arquivo modificado encontrado na pasta
        // atualiza registro no BD e marca como pendente de processamento
        BD.Exec(UPDATE PCAPIMPORTADOS
                SET DATAALT = Arquivo.dataUltAlt,
                    TAMANHO = Arquivo.tamanho,
                    PENDENTE = "S")
      FimSe
    FimSe
  FimFaça

  // gera lista de arquivos pendentes de processamento
  // em ordem cronológica de data/hora de alteração
  Tabela := BD.Open(SELECT ARQUIVO, PACOTESLIDOS
                    FROM PCAPIMPORTADOS
                    WHERE PENDENTE = "S"
                    ORDER BY DATAALT)
  Para cada Registro em Tabela Faça
    // chama módulo Extrator de Streams passando o nome do arquivo
    // o número do último pacote processado
    pacProcessados := ExtratorDeStreams(Registro.ARQUIVO,
                                        Registro.PACOTESLIDOS)

    // marca arquivo não pendente e
    // atualiza o campo referente ao número do último pacote processado
    // retornado pelo módulo Extrator de Streams
    BD.Exec(UPDATE PCAPIMPORTADOS
            SET PENDENTE = "N", PACOTESLIDOS = pacProcessados)
  FimFaça
FimRotina
```

Figura 3-8 – Algoritmo do módulo Gerenciador de Arquivos

Trata-se do módulo principal da ferramenta, já que tem como função a leitura de diversos formatos de arquivos, a extração de diversos tipos de enlace de um pacote IP, a extração de cabeçalhos de diversos encapsulamentos de pacotes IP, a remontagem de fragmentos IP e o agrupamento de fluxos TCP e UDP. O Extrator de Streams entrega ao módulo Gerenciador de Filtros uma visão apenas da camada de aplicação, escondendo toda a complexidade das demais camadas.

Formatos de arquivos de captura

Foi implementado neste módulo suporte aos formatos de arquivos de captura pcap (Wireshark Foundation, 2011), Snoop (Callaghan e Gilligan, 2005) e ETSI (ETSI, 2006). Os arquivos são abertos em modo leitura/compartilhado, assim é possível processar arquivos que estejam abertos e sendo gravados pelo programa de captura.

Enlace, Encapsulamento e Fragmentação IP

Os pacotes contidos em arquivo de captura possuem geralmente um ou mais cabeçalhos de enlace, que necessitam ser extraídos para se obter o pacote IP (Information Sciences Institute, 1981a). São reconhecidos pelo módulo os cabeçalhos de enlace Ethernet (Postel e Reynolds, 1988), LCC (Wireshark Foundation, 2010), PPPoE (Mamakos et al., 1999) e VLAN (Jeffree, 2005).

Uma vez obtido o pacote IP, deve-se avaliar se ele se trata de pacote encapsulador e, neste caso, retirar todo o seu cabeçalho. Foram implementados o reconhecimento dos encapsulamentos IP/GRE (Li et al., 2000), IP/UDP/IP, PCLI (CableLabs, 2004), Juniper (Juniper Networks, 2010) e TZSP (Wikimedia Foundation, 2011).

Pacotes IP fragmentados são tratados quando o último fragmento for processado. Como fragmentos podem residir em arquivos de captura diferentes, os fragmentos ainda não remontados são armazenados em arquivo, que é lido no início do processamento do arquivo de captura e salvo ao final. Ao carregar arquivo, fragmentos com tempo de captura muito inferior ao tempo de captura do primeiro pacote processado são desprezados. Esse tempo foi definido como 60 segundos, ou seja, um valor de segurança comparado com a recomendação de 15 segundos da RFC791 (Information Sciences Institute, 1981a).

Cabe observar que o processo de encapsulamento de um pacote IP dentro de outro pacote IP comumente gera fragmentação do pacote encapsulador, uma vez que o pacote encapsulado tende a ser do tamanho do MTU (*Maximum Transmission Unit*) do enlace. Mais ainda, o pacote IP encapsulado pode ser um fragmento, conforme exemplo apresentado na Figura 3-9:

- Pacote IP Original com 2100 *bytes* (20 *bytes* de *header* + 2180 *bytes* de *payload*) em a);
- Pacote IP Original transmitido em enlace com MTU de 1500 *bytes* gera 2 fragmentos IP em b);
- Pacotes capturados, encapsulados com IP/GRE e transmitidos em enlace com MTU de 1500 *bytes* geram 3 pacotes (2 fragmentos e 1 pacote inteiro) em c).

a) Pacote IP Original com 2100 *bytes*:

Header IP (20 *bytes*) Ident = Original FragOffset = 0 e FlagMoreFrags = 1	2080 *bytes* do *payload*

b) Pacote IP Original transmitido (2 fragmentos IPs):

Header IP (20 *bytes*) Ident = Original FragOffset = 0 e FlagMoreFrags = 1	Primeiros 1480 *bytes* do *payload* do pacote Orginal

Header IP (20 *bytes*) Ident = Original FragOffset = 1500 e FlagMoreFrags = 0	Demais 600 *bytes* do *payload* do pacote IP Original

c) 3 Pacotes capturados:

Header IP (20 *bytes*) Ident = Encapsulador1 FragOffset = 0 e FlagMoreFrags = 0	Header GRE (4 *bytes*)	Header IP (20 *bytes*) Ident = Original FragOffset = 0 e FlagMoreFrags = 1	Primeiros 1456 *bytes* do *payload* do pacote IP Original

Header IP (20 *bytes*) Ident = Encapsulador1 FragOffset = 1500 e FlagMoreFrags = 0	Próximos 24 *bytes* do *payload* do pacote IP Original.

Header IP (20 *bytes*) Ident = Encapsulador2 FragOffset = 0 e FlagMoreFrags = 0	Header GRE (4 *bytes*)	Header IP (20 *bytes*) Ident = Original FragOffset = 1500 e FlagMoreFrags = 0	Demais 600 *bytes* do *payload* do pacote IP Original

Figura 3-9 – Exemplo de pacote IP fragmentado, encapsulado e novamente fragmentado

A fim de remontar esses pacotes, uma rotina dupla de remontagem de fragmentos IP foi implementada, sendo que a primeira remonta os pacotes encapsuladores, para então retirar seus cabeçalhos e a segunda para remontar os pacotes encapsulados, caso necessário.

As rotinas de extração da camada de enlace, remontagem de fragmentos IP e desencapsulamento foram implementados segundo o algoritmo descrito na Figura 3-10.

```
Rotina ExtraiRemontaEDesencapsulaPacoteIP(pacotesFragmentados, pacote);

  // rotina ExtraiEnlace remove cabeçalhos da camada de enlace
  // e retorna o pacote IP ou nulo se não for IP
  pacoteIP := ExtraiEnlace(pacote);

  Se pacoteIP <> null então
    // caso pacoteIP seja um fragmento
    Se pacoteIP.FragOffSet <> 0 ou pacoteIP.FlagMoreFrags == 1 então
      // rotina JuntaFragmentos verifica se todos os demais
      // fragmentos estão na lista pacotesFragmentados.
      // Em caso positivo retorna pacoteIP remontado.
      // Em caso negativo guarda fragmento na lista e retorna null
      pacoteIP := JuntaFragmentos(pacotesFragmentados, pacoteIP);
    FimSe

    Se pacoteIP <> null então
      // Caso pacote seja encapsulado, retira encapsulamento
      Se ehPacoteEncapsulado(pacoteIP) então
        pacoteIP := DesencapsulaPacoteIP(pacoteIP);
        // o pacoteIP desencapsulado pode ser um fragmento IP
        // então é chamado novamente a rotina JuntaFramentos
        Se pacoteIP.FragOffSet <> 0 ou pacoteIP.FlagMoreFrags == 1 então
          pacoteIP := JuntaFragmentos(pacotesFragmentados, pacoteIP);
        FimSe
      FimSe
    FimSe
  FimSe

  Retorna pacoteIP;

FimRotina
```

Figura 3-10 – Algoritmo de extração da camada de enlace, remontagem de fragmentos IP e desencapsulamento

Agrupamento de fluxos

Agrupamento de dados de um mesmo fluxo trata-se da lógica utilizada para juntar os *payloads* da camada de transporte de um mesmo fluxo TCP ou UDP em ordem correta. Pacotes de um mesmo fluxo possuem mesmos endereços de IP de origem e destino e

portas de origem e destino (HostA:portaX <-> HostB:portaY), além de mesmo protocolo de transporte (Wagener et al., 2008), observando-se que os pacotes seguem nos dois sentidos (pacotes HostA:portaX -> HostB:portaY e HostB:portaY -> HostA:portaX fazem parte do mesmo fluxo). Como o número de porta é um valor limitado, é possível que exista um novo fluxo em outro momento com exatamente as mesmas informações de endereços de IP de origem e destino e portas de origem e destino.

Agrupamento de fluxos TCP

O protocolo TCP (Information Sciences Institute, 1981b) é um protocolo de camada de transporte do TCP/IP para envio de fluxo de dados com estabelecimento de conexão, provendo serviços de entrega confiável, ordenação e integridade de dados. Diversos protocolos de aplicação usam o TCP, entre eles HTTP (*HyperText Transfer Protocol*), SMTP (*Simple Mail Transfer Protocol*), POP (*Post Office Protocol*), e FTP (*File Transfer Protocol*).

Fluxos TCP possuem informações que permitem identificar seu início e fim, seu ordenamento, detecção de pacotes perdidos e pacotes repetidos, através da análise dos campos SEQ e ACK e as *flags* SYN e ACK do cabeçalho TCP dos pacotes que o compõem.

Na remontagem de fluxos TCP, utilizou-se como identificador a data de captura do pacote inicial, o endereço IP e porta de origem, endereço IP e porta de destino, sendo que os valores dos campos SEQ e ACK do início da conexão para pacotes TCP são armazenados para fins de referência da posição dos pacotes no fluxo.

Um novo fluxo TCP é identificado ao processar um pacote com as *flags* SYN e ACK setadas (preferiu-se essa abordagem, que alternativamente poderia ser substituída pela verificação apenas da *flag* SYN, que geraria mais fluxos sem dados nos casos de conexões rejeitadas ou não respondidas pelo *host* destino). Caso já exista um fluxo com o mesmo identificador, o anterior é marcado como fechado, não podendo mais receber pacotes, e um novo é iniciado.

Ao processar um pacote TCP com *payload* é verificado se já existe um fluxo identificado, considerando ambos os sentidos (pacotes no sentido HostA:portaX -> HostB:portaY e no sentido HostB:portaY -> HostA:portaX) na data ou na data anterior do pacote processado. Em caso positivo, o *payload* é adicionado no fluxo considerando o sentido e sua posição em relação ao pacote inicial do fluxo, sendo que o pacote será desprezado se os valores de SEQ e ACK forem incompatíveis com o fluxo identificado. Em caso negativo, o pacote também é desprezado.

O algoritmo de agrupamento de fluxos TCP é listado na Figura 3-11.

Agrupamento de fluxos UDP

O protocolo UDP (Postel, 1980) é um protocolo de camada de transporte do TCP/IP para envio de datagramas sem estabelecimento de conexão, não provendo serviços de entrega confiável, ordenação e integridade de dados. Diversos protocolos de aplicação usam o UDP, entre eles SIP (*Session Initiation Protocol*), RTP (*Real-time Transport Protocol*), DNS (*Domain Name System*), SNMP (*Simple Network Management Protocol*), RIP (*Routing Information Protocol*) e DHCP (*Dynamic Host Configuration Protocol*) (Kurose e Ross, 2010).

Fluxos UDP, diferentemente dos fluxos TCP, não possuem informações que permitam identificar seu início e fim, seu ordenamento, detecção de pacotes perdidos e pacotes repetidos.

Na remontagem de fluxos UDP, utilizou-se como identificador a data de captura do pacote inicial, o endereço IP e porta de origem, endereço IP e porta de destino, sendo que a data/hora do último pacote UDP é armazenada para diferenciação de novos fluxos.

```
Rotina AgrupaPacoteTCPnoFluxo(datahoraCaptura, pacoteIP,
                              listaFluxosAlterados)
  // funcão ArquivoDeFluxoDoPacote verifica se fluxo já existe
  // observando os dois sentidos da conexão e a datahoraCaptura e
  // (datahoraCaptura - 1)
  arqFluxo := ArquivoDeFluxoDoPacote(TCP,
                          pacoteIP.ipScr, pacoteIP.ipDst,
                          pacoteIP.TCP.prtSrc, pacoteIP.TCP.prtDst,
                          datahoraCaptura);

  Se pacoteIP.TCP.FlagSyn = 1 e
     pacoteIP.TCP.FlagAck = 1 então

    Se arqFluxo <> null então
       // caso exista um fluxo anterior com a mesma identificação
       // marca-o como fechado
       MarcaFluxoFechado(arqFluxo);
    FimSe

    arqFluxo := CriaNovoArquivoDeFluxo(TCP, pacoteIP.ipScr,
                         pacoteIP.ipDst,
                         pacoteIP.TCP.prtSrc, pacoteIP.TCP.prtDst,
                         datahoraCaptura,
                         pacoteIP.TCP.seq, pacoteIP.TCP.ack);
  Senão
    Se pacoteIP.TCP.Payload <> null então

      // Verifica o sentido do pacote em relação ao fluxo (AparaB ou
      // BparaA) para obter os valores SEQ de referência
      Se sentidoPacoteNoFluxo(pacoteIP, arqFluxo) == AparaB então
        SeqIni := arqFluxo.SeqABIni;
        SeqFim := arqFluxo.SeqABFim;
      Senão
        SeqIni := arqFluxo.SeqBAIni;
        SeqFim := arqFluxo.SeqBAFim;
      FimSe
      Se arqFluxo <> null então
        // Se valor de SEQ é fora dos padrões despreza pacote
        Se pacoteIP.TCP.seq < SeqIni ou
           pacoteIP.TCP.seq - SeqFim > LIMITEDIFSEQ então
          arqFluxo := null;
        FimSe
      FimSe

      Se arqFluxo <> null então
        // Adiciona o payload no fluxo, considerando o sentido e
        // posição em relação ao início da conexão
        EscrevePacotenoArqFluxo(arqFluxo, pacoteIP.TCP.Payload,
                                pacoteIP.TCP.Seq, SeqIni,
                                datahoraCaptura);

        // Fluxo é marcado como alterado para processamento pelo
        // módulo Gerenciador de Filtros
        listaFluxosAlterados.Adiciona(arqFluxo);
      FimSe
  FimSe
FimRotina
```

Figura 3-11 – Algoritmo de agrupamento de fluxos TCP

Pacotes UDP sem *payload* são desprezados. Ao processar um pacote UDP com *payload*, é verificado se já existe um fluxo identificado, considerando ambos os sentidos (pacotes no sentido HostA:portaX -> HostB:portaY e no sentido HostB:portaY -> HostA:portaX) na data ou na data anterior do pacote processado. Em caso positivo, adiciona-se o *payload* na posição final do fluxo considerando seu sentido, exceto se a data/hora do último pacote do fluxo é muito inferior a data/hora do pacote atual (foi adotado o tempo de uma hora[35]), e neste caso, o fluxo é marcado como fechado, iniciando-se um novo adicionando o *payload* na posição inicial deste fluxo considerando seu sentido. Em caso negativo, um novo fluxo é identificado e o seu *payload* é adicionado considerando seu sentido e na posição inicial dele.

O algoritmo de agrupamento de fluxos UDP é listado na Figura 3-12.

Algoritmo do Módulo Extrator de Streams

O módulo Extrator de Streams segue o algoritmo apresentado na Figura 3-13.

Arquivo de fluxo (AF)

Os fluxos TCP/UDP identificados são armazenados em estrutura de dados AF, que tem por objetivo simplificar o processo da construção de filtros (módulos que identificam e interpretam os protocolos/dados da camada de aplicação), já que esconde toda a complexidade inerente ao tráfego de rede bruto, tais como encapsulamento, fragmentação, ordenação, retransmissão, dentre outras.

[35] Não foi encontrado na literatura pesquisada referências sobre um tempo adequado, tendo sido arbitrado pelo autor o valor de uma hora com base de que é pouco provável que uma comunicação UDP permaneça aberta por tempo maior que esse sem comunicação, bem como sejam encontradas novas comunicações com os mesmos valores de HostA:portaX <-> HostB:portaY durante esse intervalo de tempo.

```
Rotina AgrupaPacoteUDPnoFluxo(datahoraCaptura, pacoteIP,
                              listaFluxosAlterados)
  Se pacoteIP.TCP.Payload <> null então
    // funcão ArquivoDeFluxoDoPacote verifica se fluxo já existe
    // observando os dois sentidos da conexão e a datahoraCaptura e
    // (datahoraCaptura - 1)
    arqFluxo := ArquivoDeFluxoDoPacote(UDP, pacoteIP.ipScr,
                          pacoteIP.ipDst,
                          pacoteIP.UDP.prtSrc, pacoteIP.UDP.prtDst,
                          datahoraCaptura);

    Se arqFluxo <> null então
      Se datahoraCaptura - arqFluxo.DthrFIM > LIMITETEMPOUDP então
        MarcaFluxoFechado(arqFluxo);
        arqFluxo := CriaNovoArquivoDeFluxo(UDP, pacoteIP.ipScr,
                         pacoteIP.ipDst,
                         pacoteIP.UDP.prtSrc, pacoteIP.UDP.prtDst,
                         datahora, 0, 0);
      FimSe
    Senão
      arqFluxo := CriaNovoArquivoDeFluxo(UDP, pacoteIP.ipScr,
                       pacoteIP.ipDst,
                       pacoteIP.UDP.prtSrc, pacoteIP.UDP.prtDst,
                       datahora, 0, 0);

    FimSe

    // Verifica o sentido do pacote em relação ao fluxo (AparaB ou
    // BparaA) para obter os valores SEQ de referência, que no UDP é a
    // posição final do fluxo naquele sentido.
    SeqIni := 0;
    Se sentidoPacoteNoFluxo(pacoteIP, arqFluxo) == AparaB então
      SeqFim := arqFluxo.SeqABFim;
    Senão
      SeqFim := arqFluxo.SeqBAFim;
    FimSe

    // Adiciona o payload no final fluxo, considerando o sentido.
    // Como UDP não tem SEQ, é usado o tamanho do Payload como
    // referência.
    EscrevePacotenoArqFluxo(arqFluxo, pacoteIP.UDP.Payload,
                            SeqFim + pacoteIP.UDP.TamPayload, SeqIni,
                            datahoraCaptura);

    // Fluxo é marcado como alterado para processamento pelo
    // módulo Gerenciador de Filtros
    listaFluxosAlterados.Adiciona(arqFluxo);
  FimSe
FimRotina
```

Figura 3-12 – Algoritmo de agrupamento de fluxos UDP

```
Rotina ExtraiStreams(arqCaptura, QtdePctsJaLidos)
  listaFluxosAlterados := vazio;

  // a rotina CarregaPacotesIPFrag carrega de arquivo pacotes IP
  // fragmentados ainda não remontados
  CarregaPacotesIPFrag(pacotesFragmentados);

  Para cada pacote em arqCaptura a partir de QtdePctsJaLidos faça

    // a rotina ExtraiRemontaEDesencapsulaPacoteIP recebe
    // um pacote lido
    // do arquivo de captura, extrai cabeçalhos de enlace e
    //encapsulamentos e remonta pacotes IP fragmentados.
    pacoteIP := ExtraiRemontaEDesencapsulaPacoteIP
                            (pacotesFragmentados, pacote);
    Se pacoteIP == null então Próximo

    // se camada de transporte for TCP, chama rotina de
    // agrupamento de fluxos TCP
    Se pacoteIP.prot == TCP então
       AgrupaPacoteTCPnoFluxo (pacote.datahora, pacoteIP,
                               listaFluxosAlterados);
    FimSe
    // se camada de transporte for UDP, chama rotina de
    // agrupamento de fluxos UDP
    Se pacoteIP.prot == UDP e pacoteIP.UDP.length > 0 então
       AgrupaPacoteUDPnoFluxo (pacote.datahora, pacoteIP,
                               listaFluxosAlterados);
    FimSe
  Fim Faça

  // a rotina SalvaPacotesIPFrag salva em arquivo pacotes IP
  // fragmentados ainda não remontados.
  SalvaPacotesIPFrag(pacotesFragmentados);

  // retorna o número do último pacote processado
  Retorna arqCaptura.QtdePacotes;

Fim Rotina
```

Figura 3-13 – Algoritmo do módulo Extrator de Streams

O AF é uma estrutura de dados formada por três arquivos com os mesmos nomes, mas com diferentes extensões, a saber:

- ArquivoSND: conteúdo enviado (extensão ".txt.snd");
- ArquivoRCV: conteúdo recebido (extensão ".txt.rcv");
- ArquivoMTD: metadados do fluxo bem como informações referentes a cada pacote que o compôs (extensão ".txt").

A localização (pasta) e o nome do arquivo de fluxo também contêm metadados do fluxo e possui nomenclatura no formato

"*pastaBase\str\dataIníciofluxo[formato:aaaammdd]\ipCliente-portaCliente-ipServidor-portaServidor.protTransp.extensão*" (ex.: "c:\alvox\str\20110226\10.27.5.73-138-10.27.7.200-80.tcp.txt").

Os arquivos de conteúdo (ArquivoSND e ArquivoRCV) possuem apenas os dados (*payload*) dos pacotes TCP ou UDP. No caso do protocolo TCP, os dados estão dispostos na posição física do arquivo referente à sua posição relativa no fluxo tendo como base o campo SEQ do cabeçalho do pacote TCP em relação ao seu valor no estabelecimento da conexão. Já os dados UDP são gravados na ordem de processamento dos pacotes.

O arquivo de metadados do fluxo é composto por um cabeçalho e zero ou mais registros de pacotes. Os campos do cabeçalho são apresentados na Tabela 3.2 e os campos do registro de pacote na Tabela 3.3. Um arquivo exemplo é apresentado na Figura 3-14-c.

Tabela 3.2 – Campos do cabeçalho do arquivo de metadados do fluxo

Ordem	**Campo**	**Descrição**
1	SEQABINI	Número de sequência (valor do campo SEQ do cabeçalho TCP) do primeiro pacote TCP do fluxo do *host* A
2	SEQABFIM	Número de sequência do último pacote TCP do fluxo do *host* A
3	SEQBAINI	Número de sequência do primeiro pacote TCP do fluxo do *host* B
4	SEQBAFIM	Número de sequência do último pacote TCP do fluxo do *host* B
5	DTHRINI	Data/hora de captura do primeiro pacote do fluxo
6	DTHRFIM	Data/hora de captura do último pacote do fluxo

Como pacotes IP/UDP não possuem valores de SEQ e ACK, são usados os valores relativos à quantidade de dados (*payload*) transferidos em relação ao início do fluxo.

Tabela 3.3 – Campos do registro de pacote do arquivo de metadados do fluxo

Ordem	**Campo**	**Descrição**
1	SEQ	Valor do campo SEQ do cabeçalho do pacote TCP relativo ao início da conexão
2	ACK	Valor do campo ACK do cabeçalho do pacote TCP relativo ao início da conexão
3	DTH	Data/hora de captura do pacote
4	STD	Sentido do pacote, sendo o valor “SND” para pacote endereçado no sentido *host* A para *host* B e valor “RCV” para pacote endereçado no sentido *host* B para *host* A. No caso de pacotes do tipo “RCV” os valores dos campos SEQ e ACK são invertidos.

Na Figura 3-14 temos uma conexão exemplo, sua visualização no programa Wireshark em a) e b) e a estrutura de dados de AF gerada pelo módulo Extrator de Streams em c), d) e e). Trata-se de uma conexão TCP/HTTP entre o *host* A (endereço IP 192.168.1.64 e porta 3695) e o *host* B (servidor *web* com endereço IP 208.37.10.13 e porta 80) composta por 10 pacotes IP, sendo que:

- Nos três primeiros pacotes ocorre o estabelecimento da conexão iniciada pelo *host* A;
- No quarto pacote o *host* A solicita um objeto com o comando GET;
- No quinto pacote o *host* B responde com o objeto solicitado;
- No sexto pacote o *host* A apenas envia confirmação de pacote recebido;
- No sétimo pacote o *host* A solicita um segundo objeto com o comando GET;
- No oitavo pacote *host* B responde com o segundo objeto solicitado;
- No nono pacote o *host* A apenas envia confirmação de pacote recebido;
- No décimo pacote o *host* B interrompe a conexão enviando RST.

Na Figura 3-14, detalhadamente, temos:

- Em a) a visualização dos pacotes IP no programa Wireshark incluindo os valores dos campos SEQ e ACK do cabeçalho TCP absolutos e relativos em relação ao início da conexão;

- Em b) a visualização do conteúdo do fluxo TCP/HTTP completo da conexão exemplo, sendo em destaque o conteúdo enviado pelo *host* A e sem destaque o conteúdo enviado pelo *host* B;
- Em c) o conteúdo do arquivo de metadados da conexão, onde observa-se nas duas primeiras linhas o seu cabeçalho e nas demais linhas quatro registros de pacotes, pois dos 10 pacotes da conexão exemplo apenas 4 possuem dados TCP (*payload*). Também se observa a nomenclatura do arquivo, que também contêm informações de metadados da conexão exemplo;
- Em d) e e) o conteúdo dos arquivos enviado e recebido da estrutura arquivo de fluxo, referenciado no lado esquerdo pelo seu deslocamento em relação ao início do arquivo e obedecendo a mesma nomenclatura do arquivo de metadados.

3.2.1.4 Módulo Gerenciador de Filtros e Módulos Filtros

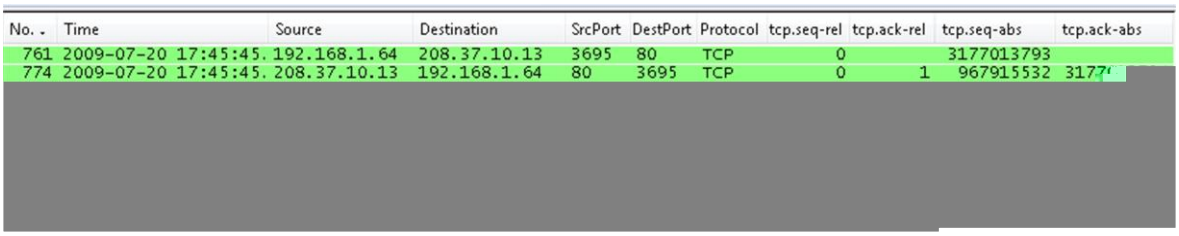
O módulo Gerenciador de Filtros é o responsável por receber uma lista de AFs e chamar individualmente cada Filtro registrado para cada AF dessa lista. Um Filtro é um módulo que recebe um AF e caso identifique o protocolo de camada de aplicação através de análise do seu conteúdo e/ou dos seus metadados, decodifica-o gerando objetos para serem exibidos para o usuário com uma visualização de alto nível (Figura 3-15). Cada objeto é composto por um conjunto de metadados, um arquivo de conteúdo e, opcionalmente, um ou mais arquivos secundários. Os metadados de um objeto decodificado, que devem ser informados pelo módulo Filtro para o módulo Gerenciador de Filtros, são:

- Tipo do objeto (TipoObj): breve texto que identifica o filtro que o gerou (ex: “WEB”, “WEBMAIL”, “EMAIL”, “MSNCHAT”, “RTP”, etc.).
- Informação adicional (Info): breve texto indicando mais detalhes do objeto (ex: “*E-mail* Enviado”, “*E-mail* Recebido”, “*Webmail* Yahoo Enviado”, etc.).
- Observação (Coment): campo textual longo para descrição livre do objeto pelo filtro. É pesquisável e pode ser editado pelo usuário.
- Classificação (Status): o filtro deve indicar a relevância do objeto como “Normal”, “Alerta” ou “Sem Relevância” através de política interna e/ou configurações.
- Carga (Web): objeto pode ser do tipo ObjetoWeb ou ObjetoFile. Objetos do tipo ObjetosWeb são carregados pelo programa Analisador através de requisição HTTP da URL do objeto, que é respondida pelo mesmo programa através de carga do

sistema de arquivos. Já os objetos do tipo ObjetoFile são carregados pelo programa Analisador diretamente do sistema de arquivos.

- URL (Url): URL do objeto identificado caso ele seja do tipo ObjetoWeb ou campo livre para demais objetos.
- Arquivo (Arquivo): nome do arquivo gerado no sistema de arquivos contendo o objeto decodificado.
- Tipo do conteúdo (ContentType): tipo do conteúdo do arquivo (ex: “text/html", “text/plain”, “image/jpeg”, etc.).
- Dados de identificação do fluxo (IpOrig, PtOrig, IpDest, PtDest, Prot, DataHoraIni e DataHorafim): IP e porta do *host* cliente, IP e porta do *host* servidor, protocolo da camada de transporte e data/hora de início e fim da transferência do objeto. Estes dados são atribuídos pelo Gerenciador de Filtros e não pelo Filtro, e permitem que o Investigador chegue rapidamente aos pacotes que geraram determinado objeto em uma ferramenta de análise de tráfego a fim de validar os resultados apresentados pelo CLIT.

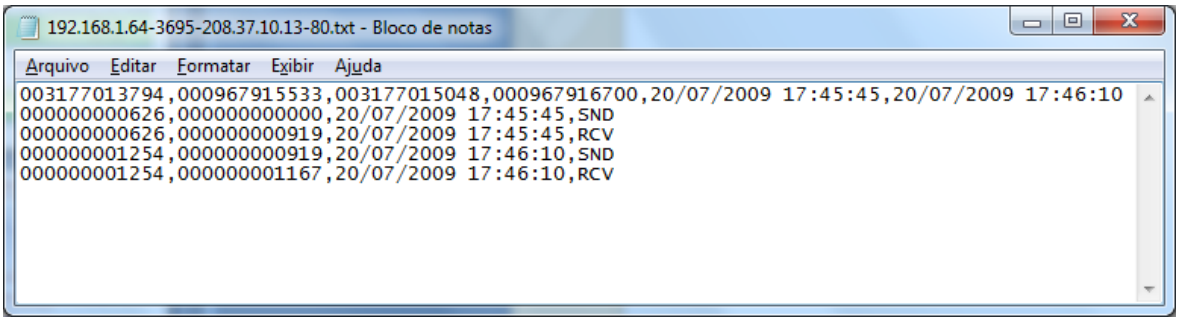
O Gerenciador de filtros armazena, então, os metadados dos objetos nos campos TipoObj, Info, Coment, Status, Web, Url, Arquivo, ContentType, IpOrig, ptOrig, IpDest, ptDest, Prot, DataHoraIni e DataHoraFim, respectivamente, da tabela INDEXDAT do banco de dados da conexão monitorada.



No.	Time	Source	Destination	SrcPort	DestPort	Protocol	tcp.seq-rel	tcp.ack-rel	tcp.seq-abs	tcp.ack-abs
761	2009-07-20 17:45:45.	192.168.1.64	208.37.10.13	3695	80	TCP	0		3177013793	
774	2009-07-20 17:45:45.	208.37.10.13	192.168.1.64	80	3695	TCP	0	1	967915532	3177

a) Visualização de pacotes de uma conexão TCP/HTTP no programa Wireshark

b) Visualização de um fluxo TCP/HTTP no programa Wireshark



c) Conteúdo do arquivo de metadados (20090720\192.168.1.64-3695-208.37.10.13-80.txt)

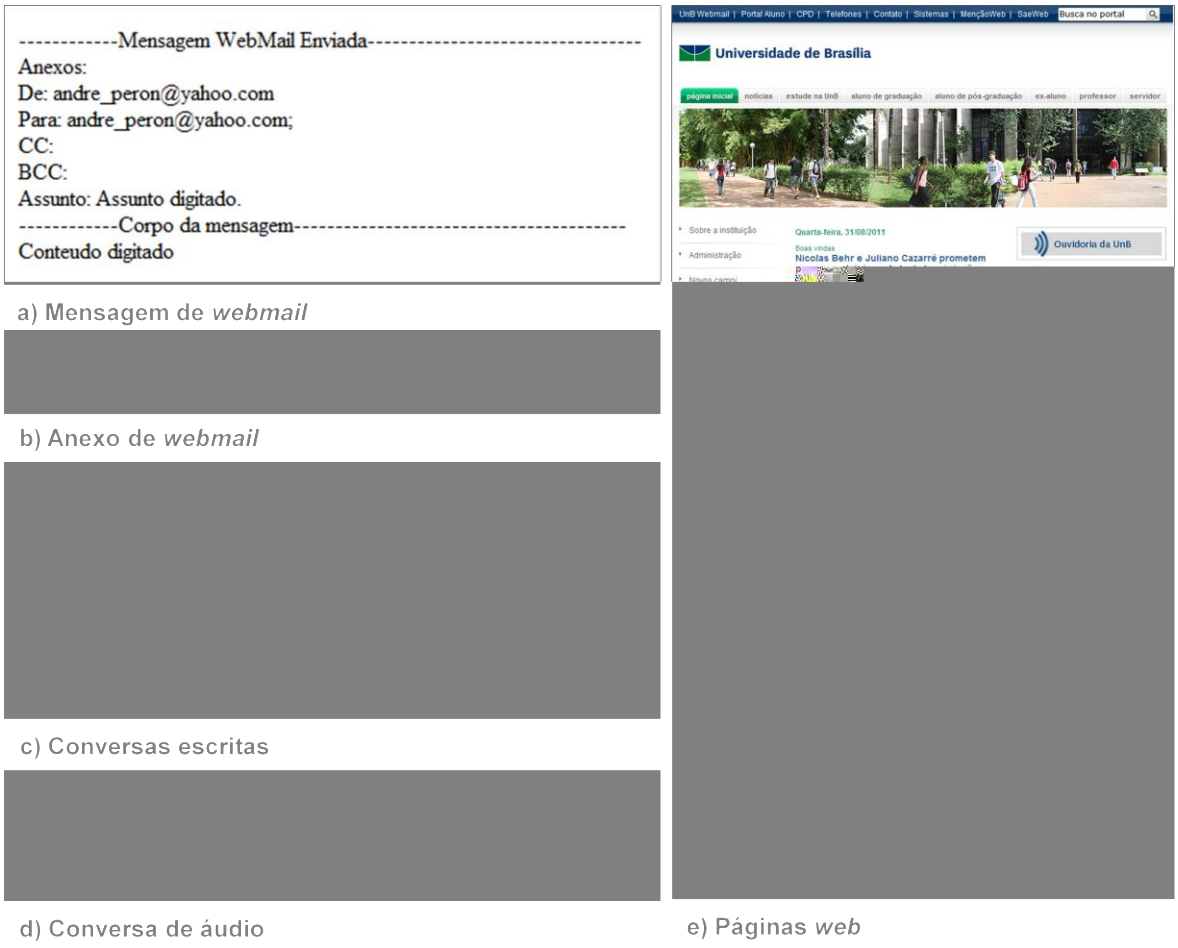
Offset	
00000000	GET //org-img/calif/estr_1.gif HTTP/1.1..Accept: */*..Referer: h
00000064	ttp://br.mc583.mail.yahoo.com/mc/showMessage?sMid=0&fid=Inbox&so
00000128	rt=date&order=down&startMid=0&filterBy=&.rand=1415532356&midInde
00000192	x=0&mid=1_88525_AJzsjkQAAFFAS10QAgezblsmTOI&f=1..Accept-Language
00000256	: pt-br..UA-CPU: x86..Accept-Encoding: gzip, deflate..User-Agent
00000320	: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Mozilla/4.0
00000384	(compatible; MSIE 6.0; Windows NT 5.1; SV1) ; Embedded Web Bro
00000448	wser from: http://bsalsa.com/; .NET CLR 2.0.50727; InfoPath.2)..
00000512	Host: www.mercadolivre.com.br..Connection: Keep-Alive..Cookie: p
00000576	msctx=*****SASUS+FERRARI%7CSPS3%7CSATA+VOIP%7C....GET /jm/ml.dms
00000640	.hit?p=-4:80589918 HTTP/1.1..Accept: */*..Referer: http://br.mc5
00000704	83.mail.yahoo.com/mc/showMessage?sMid=0&fid=Inbox&sort=date&orde
00000768	r=down&startMid=0&filterBy=&.rand=1415532356&midIndex=0&mid=1_88
00000832	525_AJzsjkQAAFFAS10QAgezblsmTOI&f=1..Accept-Language: pt-br..UA-
00000896	CPU: x86..Accept-Encoding: gzip, deflate..User-Agent: Mozilla/4.
00000960	0 (compatible; MSIE 7.0; Windows NT 5.1; Mozilla/4.0 (compatible
00001024	; MSIE 6.0; Windows NT 5.1; SV1) ; Embedded Web Browser from: h
00001088	ttp://bsalsa.com/; .NET CLR 2.0.50727; InfoPath.2)..Host: www.me
00001152	rcadolivre.com.br..Connection: Keep-Alive..Cookie: pmsctx=*****S
00001216	ASUS+FERRARI%7CSPS3%7CSATA+VOIP%7C....

d) ArquivoSND (20090720\192.168.1.64-3695-208.37.10.13-80.txt.snd)

Offset	
00000000	HTTP/1.1 200 OK..Age: 30308..Accept-Ranges: bytes..Date: Mon, 20
00000064	Jul 2009 12:20:34 GMT..Content-Length: 592..Content-Type: image
00000128	/gif..Expires: Wed, 19 Aug 2009 12:20:34 GMT..Cache-Control: max
00000192	-age=2592000..Server: Apache/2.0.54 (Unix)..Last-Modified: Fri,
00000256	30 Nov 2007 13:45:32 GMT..Via: 1.1 CACHE-06 (NetCache NetApp/6.0
00000320	.7)....GIF89a....æb.þ÷ñ÷▮Xø®tù¶▮ýçÕþïãô▮ üÖ¹üÞÇûÎ«ù¾.úè<úê<÷▮$õ.
00000384	<÷¦fú¹-þÃ.ú·,ûÎ4ûæ6úé:úà:ù·&ûº.üÜ,úÆ$õ▮.ù®'ûÇ1÷▮Jø▮%ú²+ö▮JüÑ ù«(
00000448	úê;ù$'øż.ê. ö´0úÑ5ýÏ.úÔ6ö´1÷▮$úæ<õ▮ ù▮.ûà/ö▮/÷¦*ûå5õ▮!ô▮"ö.!ûË3ø
00000512	¥(ûż.ö.#ø¢&ùµ#úç8úÜ9ø▮.ö´.ü¯.ûã3þÈ.õ▮.ø¤&þÉ.úÈ'÷▮#÷▮'ü°.õ▮'õ▮#ú×
00000576	7ú´,ûÅ0ø¯,÷▮#úã;úÆ.úÙ8õ▮!÷..ø¡%ö."ö▮"ùÄ2ûÅ/ö▮!ö▮.ù-)÷¥*üÚ)ÿÿÿ...
00000640	.....
00000704	.....!ù....b.,.....-▮b▮▮▮.▮▮▮▮..▮.▮.'.▮▮.]V▮
00000768	▮▮..I-´▮▮...Y.%;....../5ZX. .`3L,..▮..7R.<#O:8U..H^.▮.2.F_.\.?.
00000832	C..Í▮..9.P).$1=!▮▮.J.+S.."Aé▮TQN.>a*G@Ü▮.¶, q..´.0.d.ÀÅ....®,.¢
00000896	▮.&.Ò..P$À$... BÀ.▮É@.;HTTP/1.1 200 OK..Date: Mon, 20 Jul 2009 2
00000960	0:46:06 GMT..Content-Length: 43..Content-Type: image/gif..Cache-
00001024	Control: max-age=86400, private..Server: Resin/3.0.18..Via: 1.1
00001088	CACHE-06 (NetCache NetApp/6.0.7)....GIF89a....▮..ÿÿÿ...!ù.....,
00001152	.....D..;

e) ArquivoRCV (20090720\192.168.1.64-3695-208.37.10.13-80.txt.rcv)

Figura 3-14 – Exemplo de um Arquivo de Fluxo



a) Mensagem de *webmail*

b) Anexo de *webmail*

c) Conversas escritas

d) Conversa de áudio

e) Páginas *web*

Figura 3-15 – Objetos decodificados pelos módulos Filtros para serem visualizadas no programa Analisador

O Gerenciador de Filtros permite a execução de módulos Filtros externos, ou seja, programas externos de decodificação de protocolos, que podem ser implementados em qualquer linguagem de programação.

Foram implementados no programa Importador os módulos Filtros WEB, POP, SMTP, RTP, MSN, YMSG e ICQ.

3.2.1.5 Filtro WEB

O Filtro WEB foi desenvolvido para decodificar objetos transferidos através do protocolo HTTP (versões 1.0 e 1.1). Embora originalmente fosse destinado a transferência de páginas *web*, hoje diversas aplicações utilizam o protocolo HTTP para serviços de *e-mail*, *chats*, *streaming* de áudio e vídeo, etc., e tem se tornado bastante populares. Por isso, o Filtro

WEB, além de decodificar páginas *web*, foi implementado para reconstruir objetos de *e-mails* e conversas de alguns serviços comumente utilizados no Brasil através de rotinas específicas.

O protocolo HTTP é basicamente composto por requisição de recurso (caminho e arquivo) contendo cabeçalho e dados (opcional) e resposta contendo cabeçalho e dados (opcional). Na sua versão 1.0 (Berners-Lee et al., 1996) uma conexão só pode conter uma requisição e sua resposta. Já na versão 1.1 (Fielding et al., 1999), são suportadas diversas requisições em uma mesma conexão (conexão persistente). São comumente utilizados pelos clientes HTTP a requisição HTTP/GET para obter objetos *web* e a requisição HTTP/POST para envio de dados, incluindo arquivos. No cabeçalho da requisição/resposta destacam-se os parâmetros "Content-Type" para indicar o tipo de conteúdo (texto, *HyperText Markup Language* - HTML, imagem, etc.), "Content-Length" para indicar o tamanho dos dados transferidos, "Content-Encoding" para indicar se o conteúdo está compactado, "Transfer-Encoding" para indicar transferência de dados em blocos (*chunked*) e "*Host*" para o nome do servidor onde se localiza o recurso (obtém-se a URL concatenando o valor do campo "*Host*" com o recurso requisitado).

O Filtro WEB identifica um arquivo de fluxo como sendo HTTP através da busca no ArquivoSND pelas *strings* "GET", "POST", "HEAD" ou "PUT" na posição zero e pela *string* "HTTP/1." antes da primeira ocorrência dos caracteres <CR><LF>. Uma vez positivado, o Filtro WEB lê uma requisição HTTP do ArquivoSND e sua resposta do ArquivoRCV e assim sucessivamente até o final de um deles. Requisições sem resposta ou respostas sem requisição são registradas em *log* para fim de depuração. São também suportadas pelo filtro requisições HTTP para *proxys*, transferências *chunked*, conteúdo compactado e conexões persistentes. Conforme o tipo de requisição, as seguintes ações são executadas:

- Requisição HTTP/GET: gera um arquivo com o conteúdo da resposta e conjunto de metadados com Tipo de objeto = "WEB", Carga = "ObjetoWeb", Arquivo = nome do arquivo gerado, URL = URL requisitada e Tipo de conteúdo = conforme parâmetro "Content-Type" cabeçalho da resposta HTTP;
- Requisição HTTP/POST: gera um arquivo com o conteúdo postado e conjunto de metadados com Tipo de objeto = "WEBPOST", Carga = "ObjetoFile", Arquivo = nome do arquivo do conteúdo postado, URL = URL requisitada e Tipo de conteúdo

= “text/html"; um arquivo com o conteúdo da resposta e conjunto de metadados com Tipo de objeto = “WEB”, Carga = “ObjetoWeb”, Arquivo = nome do arquivo do conteúdo da resposta, URL = URL requisitada e Tipo de conteúdo = conforme parâmetro “Content-Type” cabeçalho da resposta HTTP, Caso tenham sido postados arquivos, além de serem gerados no sistema de arquivo, são exibidos em forma de *link* no arquivo de conteúdo postado.

O Filtro WEB também é responsável por decodificar os serviços de envio e recebimento de *e-mails* através do protocolo HTTP (*webmails*). Para cada serviço de *webmail*, uma rotina de tratamento foi desenvolvida tratando os seguintes objetos:

- *E-mails* enviados: dados do *e-mail* (campos de, para, *Carbon Copy* - CC, *Blind Carbon Copy* - BCC, assunto, conteúdo e nome de arquivos anexos) são enviados para o servidor através de requisição HTTP/POST, sendo que o nome dos campos varia de serviço para serviço. O Filtro WEB identifica um *e-mail* enviado através da URL e de uma tabela de nomes de campos interna para cada serviço tratado. É gerado um arquivo HTML de exibição do *e-mail* e conjunto de metadados com Tipo de objeto = “WEBMAIL”, Informação Adicional = “*Webmail* $idserviço – mensagem enviada”, Carga = “ObjetoFile”, Arquivo = nome do arquivo HTML, URL = URL requisitada, Observação = campos do cabeçalho do *e-mail* e Tipo de conteúdo = “text/html";
- Anexos enviados: os arquivos anexados a um *e-mail* são enviados normalmente através de requisições HTTP/POST com conteúdo formatado como “multipart/form-data”. O filtro WEB identifica-os através da URL. É gerado o arquivo postado no sistema de arquivos e um arquivo HTML com *link* para ele e conjunto de metadados com Tipo de objeto = “WEBMAIL”, Informação Adicional = “*Webmail* $idserviço – anexo enviado”, Carga = “ObjetoFile”, Arquivo = nome do arquivo HTML, URL = URL requisitada, Observação = nome dos arquivos enviados e Tipo de conteúdo = “text/html";
- *E-mails* lidos: cada serviço de *webmail* adota um formato para obter os dados de um *e-mail*. Nos mais simples, os *e-mails* são arquivos HTML, apresentado os dados de cabeçalho e conteúdo e *links* para os arquivos anexos, que são exibidos diretamente em um *frame* do navegador. Nos mais avançados, *os e-mails* estão em formato XML (e*Xtensible Markup Language* – Bray et al. (2008)), *arrays* JavaScript (ECMA International, 1999), objetos JavaScript (JSON – *JavaScript*

Object Notation – Crockford (2006a) e Crockford (2006b)), dentre outros, que são processados por *scripts* no navegador e apresentados em forma legível para o usuário. O Filtro WEB identifica-os através da URL e, para cada serviço, uma rotina foi desenvolvida para convertê-los em formato HTML, também de forma legível, mas sem vínculo com a exibição original do serviço. É gerado um arquivo HTML de exibição do *e-mail* e conjunto de metadados com Tipo de objeto = "WEBMAIL", Informação Adicional = "*Webmail* $idserviço – mensagem lida", Carga = "ObjetoFile", Arquivo = nome do arquivo HTML, URL = URL requisitada e Tipo de conteúdo = "text/html". Adotou-se a nomenclatura "*E-mails* lidos" ao contrário de "*E-mails* recebidos", pois o *e-mail* visualizado pelo usuário interceptado pode estar, por exemplo, na pasta "Enviados", não sendo possível fazer essa diferenciação de forma automática;

- Anexos baixados: os anexos baixados são obtidos como qualquer outro objeto HTTP, normalmente através requisições HTTP/GET. O Filtro WEB identifica-os através da URL. É gerado o arquivo baixado no sistema de arquivos e um arquivo HTML com *link* para ele e conjunto de metadados com Tipo de objeto = "WEBMAIL", Informação Adicional = "*Webmail* $idserviço – anexo baixado", Carga = "ObjetoFile", Arquivo = nome do arquivo HTML, URL = URL requisitada e Tipo de conteúdo = "text/html".

Os anexos não são vinculados pelo filtro aos *e-mails* que os referenciam, pois são enviados em requisição separada, cabendo ao Investigador fazer a sua associação em seu relatório através da ordem cronológica dos eventos ou mesmo informações contidas no cabeçalho ou corpo do *e-mail*.

Normalmente não está disponível documentação dos desenvolvedores dos serviços de *webmail*, sendo que a implementação das rotinas baseou-se em experiências de uso e monitoramento do tráfego gerado.

Foram implementadas rotinas para decodificar os serviços de *webmail* Hotmail[36], Yahoo!Mail[37] (versões Classic, AllNew e Neo), IG[38] e UOL/BOL[39].

[36] Disponível em http://hotmail.com
[37] Disponível em http://mail.yahoo.com

Como exemplo, apresenta-se os parâmetros utilizados para a decodificação de objetos do serviço Yahoo!Mail:

- *E-mail* enviado: a rotina para a versão Classic identifica requisições HTTP/POST com URL contendo “mail.yahoo.com/mc/compose?” e campo “Content-Type” do cabeçalho contendo “application/www-form-urlencoded” ou “multipart/form-data”, e utiliza nos dados enviados os valores dos campos “defFromAddress”, “to”, “cc”, “bcc”, “Subj”, “Content” e “attachment” para reconstrução dos campos de, para, CC, BCC, assunto, corpo e nome dos arquivos anexados do *e-mail* enviado (o conteúdo do anexo é enviado em outra requisição). Na Figura 3-16 é apresentada uma requisição HTTP/POST de envio de um *e-mail* de “andre_peron@yahoo.com” para “andre_peron@yahoo.com”, com assunto “Assunto digitado”, corpo “Conteúdo digitado” e arquivo anexo “arquivo.txt”;
- Anexo enviado: a rotina para a versão Classic identifica requisições HTTP/POST com URL contendo “mail.yahoo.com/ya/upload” e campo “Content-Type” do cabeçalho contendo “multipart/form-data”, e utiliza nos dados enviados os valores dos campos de formulário com atributo “filename” para reconstrução dos anexos enviados. Na Figura 3-17 é apresentada uma requisição HTTP/POST de envio de um arquivo anexo de nome “arquivo.txt” com conteúdo “Conteúdo do arquivo.”, sendo que nenhuma outra informação sobre o *e-mail* (de, para, assunto, etc.) é transmitida de forma clara para que o filtro possa fazer vínculo;

[38] Disponível em http://mail.ig.com.br
[39] Disponível em http://bol.com.br

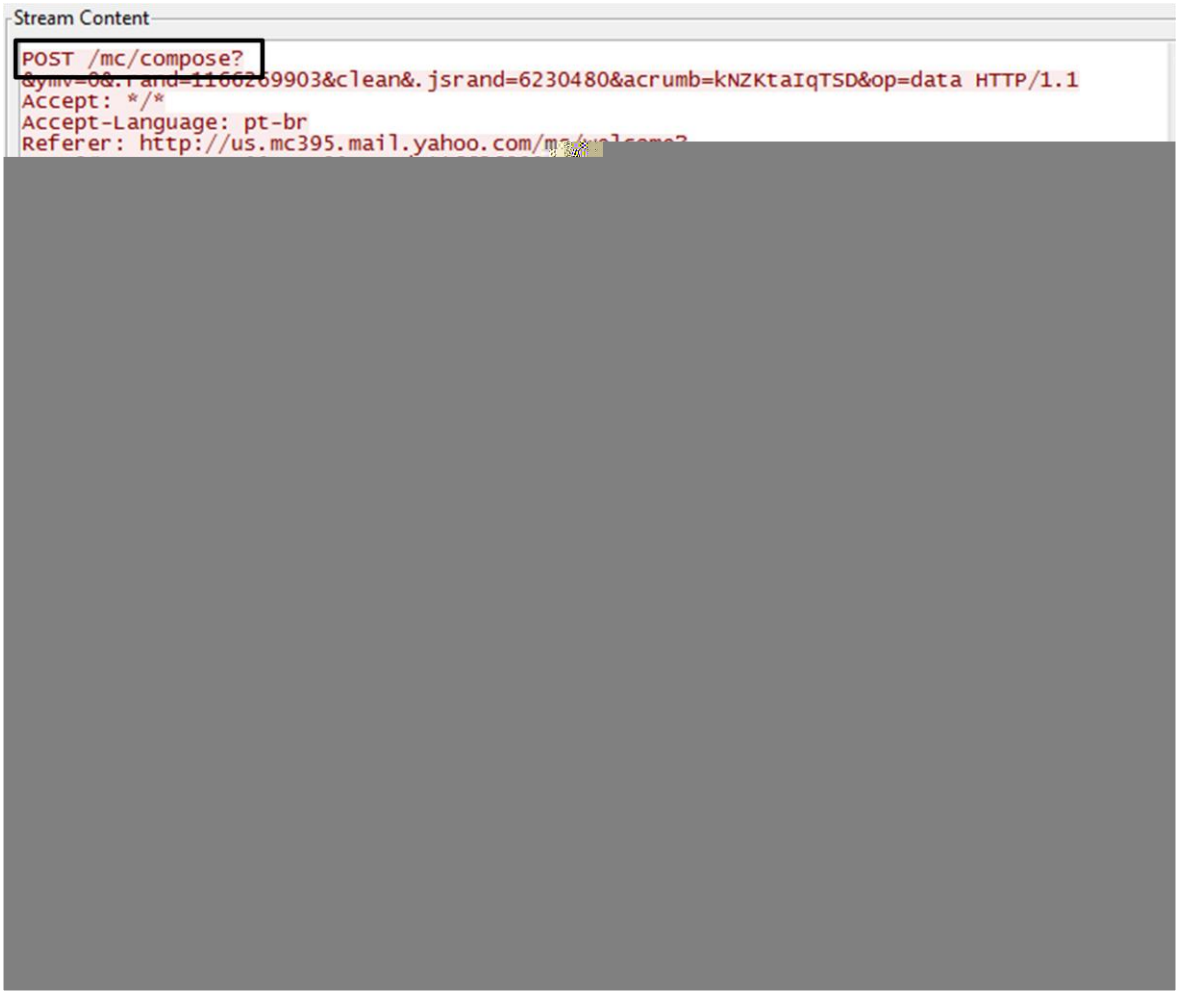


Figura 3-16 – Requisição HTTP para envio de *e-mail* através do Yahoo!Mail Classic

- *E-mail* lido: a rotina para a versão AllNew identifica respostas HTTP de requisições contendo “mail.yahoo.com/” e “m=GetDisplayMessage”, e utiliza nos dados recebidos, que estão codificados como JSON, os valores dos atributos “result.message[].header.from.email”, “result.message[].header.to[].email”, “result.message[].header.cc[].email”, “result.message[].header.bcc[].email”, “result.message[].header.subject”, “result.message[].header.receivedDate”, “result.message[].part[].text”, “result.message[].part[].filename” para reconstrução dos campos de, para, CC, BCC, assunto, data de recebimento, corpo e nome dos arquivos anexados do *e-mail* lido. Na Figura 3-18 é apresentada uma requisição e sua resposta HTTP de leitura de um *e-mail* de “andre.peron@yahoo.com” para “andre.peron@yahoo.com”, com assunto “Assunto digitado”, corpo “Conteúdo digitado.” e arquivo anexo “arquivo.txt” (foram removidas na figura partes da requisição e da resposta HTTP para facilitar entendimento do exemplo);

Figura 3-17 – Anexo de *e-mail* enviado através do serviço de *webmail* Yahoo!Mail Classic

- Anexo baixado: a rotina para a versão Classic identifica respostas HTTP de requisições HTTP com URL contendo "mail.yahoo.com" e campo "Content-Disposition" do cabeçalho contendo "attachment", e utiliza todos os dados recebidos como conteúdo do arquivo anexo baixado. Na Figura 3-19 é apresentada uma requisição HTTP e sua resposta de *download* de um anexo de nome "arquivo.txt" com conteúdo "Conteúdo do arquivo." (o conteúdo na figura está codificado em gzip), sendo que nenhuma outra informação sobre o *e-mail* (de, para, assunto, etc.) é transmitida de forma clara para que o filtro possa fazer vínculo.

Foram implementadas também rotinas para decodificação de conversas via *web* dos serviços eBuddy[40] (permite que usuários dos aplicativos de conversa instantânea MSN, Yahoo!Messenger, AIM, GTalk e ICQ conversem com seus contatos sem instalar os

[40] Disponível em http://ebuddy.com

referidos aplicativos), Facebook[41] (permite que usuário converse com seus amigos da rede social) e Orkut[42] (permite que usuário converse com seus amigos da rede social e contatos do aplicativo GTalk). As rotinas foram desenvolvidas baseadas nas documentações de Agarwal et al. (2010) e Ho et al. (2009), além de experiências de uso dos aplicativos e seu monitoramento.

Figura 3-18 – *E-mail* lido através do serviço de *webmail* Yahoo!Mail AllNew

Para cada conversa identificada é gerado um arquivo HTML com as mensagens digitadas, indicando seu autor e data/hora, e um conjunto de metadados com Tipo de objeto = $idServiço (“EBUDDY”, “FBCHAT” ou “GTALK”), Informação adicional = “Chat $idServiço” , Classificação = “Alerta”, Carga = “ObjetoFile”, Arquivo = nome do arquivo HTML gerado, Tipo de conteúdo = “text/html" e Observação = usuários que participaram da conversa. Em determinadas situações as mensagens digitadas ficam espalhadas em diversas conexões como se fossem conversas diferentes, cabendo ao Investigador, através

[41] Disponível em http://facebook.com
[42] Disponível em http://orkut.com

de análise cronológica das mensagens de mesmos interlocutores, juntá-las para o correto entendimento.

Figura 3-19 – Anexo de *e-mail* baixado através do serviço de *webmail* Yahoo!Mail Classic

A título de exemplo, na Figura 3-20 é mostrada uma conversa realizada através do serviço Facebook em que o usuário “Andre Andre Erdna” envia o texto “msg 1a” para o usuário “Andre Senlegen”. A rotina de decodificação busca por respostas de requisições de URLs contendo “channel.facebook.com” espalhadas por uma mesma conexão HTTP persistente, onde os dados recebidos estejam codificados em JSON e possuam os objetos/atributos “ms[].msg.text”, “ms[].msg.from_name” e “ms[].msg.to_name” que são decodificados como texto digitado, usuário de origem e usuário de destino, respectivamente.

As rotinas de tratamento de *webmail* e *chats* necessitam ser revisadas constantemente (Cohen, 2008), já que sofrem alterações de tempos em tempos de seus desenvolvedores sem prévio aviso. Além de revisar, é necessário que as versões antigas continuem a ser decodificadas, pois a ferramenta pode ser demandada com arquivos de captura que envolvam períodos com diferentes versões do mesmo serviço.

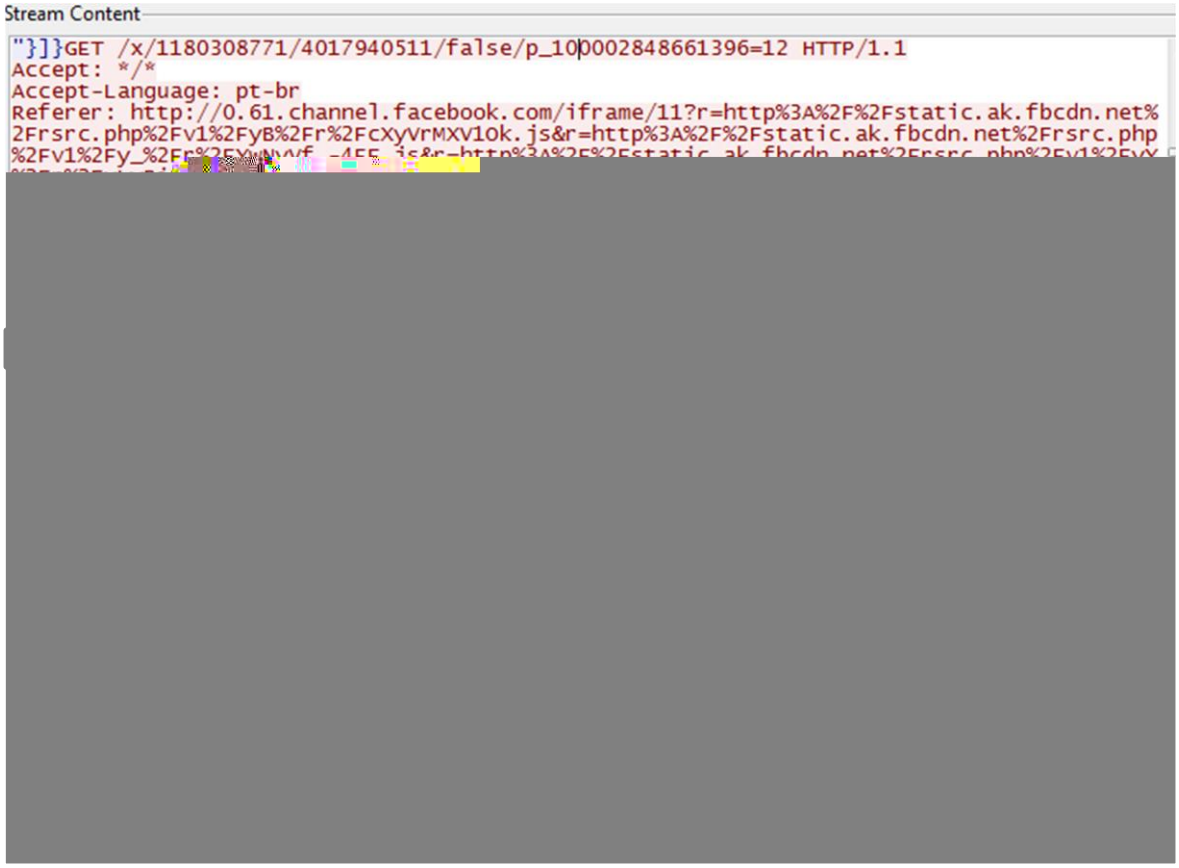


Figura 3-20 – Conversa realizada através do serviço Facebook

Foram adotadas as seguintes políticas internas de classificação de relevância para o Filtro WEB:

- Objetos com tamanho zero e os dados enviados através de requisições HTTP/POST que não contenham arquivo são marcados como “Sem importância”;
- Objetos de *webmail*, de *chats* e referentes a arquivos enviados através de requisições HTTP/POST são marcados como “Alerta”;
- Demais objetos *web* são marcados como “Normal”.

É possível ainda configurar URLs para serem marcadas automaticamente como “Alerta” ou “Sem importância” através dos arquivos “webAlerta.txt” e “webSemRelevancia.txt”, respectivamente, onde se adiciona uma URL por linha. Pode-se utilizar o caractere coringa “*” no início, no fim ou em ambas as posições da URL para designar URLs que terminam, iniciam ou contêm a *string* informada.

3.2.1.6 Filtro POP

O Filtro POP é responsável por decodificar *e-mails* recebidos através do protocolo POP3, comumente utilizado por programas cliente de *e-mail*.

O protocolo POP3 (*Post Office Protocol Version 3* – Myers e Rose, 1996) permite *download* de *e-mails* armazenados em um servidor através de uma conexão TCP (porta padrão 110). Resumidamente, o POP3 implementa comandos de autenticação ("USER" e "PASS" para usuário e senha, respectivamente) e de manipulação de *e-mails* ("LIST" para listar todos os e-mails disponíveis, "RETR" para obter um determinado e-mail, "DELE" para deletar determinado e-mail do servidor, dentre outros). Os comandos só são enviados pelo cliente, devendo o servidor responder com sucesso (*string* "+OK", o conteúdo solicitado e *string* "<CR><LF>.<CR><LF>") ou com erro (*string* "-ERR").

O Filtro POP foi implementado através da leitura exclusiva do ArquivoRCV do fluxo que, caso seja iniciado pela *string* "+OK", é lido linha a linha até ser encontrado um ou mais *e-mails* até o final do arquivo. O início de um *e-mail* é identificado quando for lida uma linha contendo a *string* "@" e o fim quando for encontrada a *string*" <CR><LF>.<CR><LF>".

Para cada *e-mail* identificado são gerados:

- Um arquivo com extensão "EML" com o conteúdo bruto do *e-mail*, que pode ser lido em programas clientes de *e-mail*;
- Um arquivo para cada anexo encontrado no *e-mail*;
- Um arquivo HTML apresentando os principais dados do cabeçalho (de, para, CC, assunto e data), o corpo, lista dos arquivos anexos com *links* para os mesmos (arquivos do tipo imagem são exibidos diretamente no HTML) e *link* para o arquivo "EML";
- Um conjunto de metadados (um registro) com Tipo de objeto = "EMAIL", Informação adicional = "E-mail Recebido(POP)", Classificação = "Alerta", Carga = "ObjetoFile", Arquivo = nome do arquivo HTML gerado, Tipo de conteúdo = "text/html" e Observação = principais dados do cabeçalho do *e-mail*.

3.2.1.7 Filtro SMTP

O Filtro SMTP é responsável por decodificar *e-mails* enviados através do protocolo SMTP, amplamente utilizado para comunicação entre servidores de *e-mail* e por programas clientes de *e-mail* (Sureswaran et al., 2009).

O protocolo SMTP (*Simple Mail Transfer Protocol* – Postel, 1982) foi desenvolvido para envio de *e-mails* somente no sentido cliente para servidor através de conexão TCP (porta padrão 25). É basicamente composto por comando de identificação (“HELO” ou “EHLO”) e comandos de envio de *e-mails* (“MAIL FROM:” para indicar o remetente, “RCPT TO:” para indicar os destinatários, “DATA” para enviar todo o conteúdo do e-mail). Extensões do SMTP permitem outras funções, tais como autenticação, verificação de tamanho limite de *e-mails*, dentre outras.

O Filtro SMTP foi implementado através da leitura exclusiva do ArquivoSND do fluxo, que caso seja iniciado pela *string* “HELO” ou “EHLO”, é lido linha a linha até ser encontrado um ou mais *e-mails* até o final do arquivo. O início de um *e-mail* é identificado quando for lida uma linha contendo exatamente a *string* “DATA” e o fim quando for encontrada a *string*“ <CR><LF>.<CR><LF>”.

Para cada *e-mail* identificado são gerados os mesmo arquivos do Filtro POP e um conjunto de metadados com Tipo de objeto = “EMAIL”, Informação adicional = “E-mail Enviado(SMTP)”, Classificação = “Alerta”, Carga = “ObjetoFile”, Arquivo = nome do arquivo HTML gerado, Tipo de conteúdo = “text/html" e Observação = principais dados do cabeçalho do *e-mail*.

3.2.1.8 Filtro RTP

O Filtro RTP é responsável por decodificar conversas de áudio realizadas através do protocolo RTP, amplamente utilizado por aplicativos de telefonia e comunicadores instantâneos.

O protocolo RTP (*Real-Time Transport Protocol* – Schulzrinne et al., 2003) foi desenvolvido com o objetivo de transportar fluxos de áudio e vídeo, ponto a ponto e de

tempo real através da Internet. É utilizado em conjunto com o protocolo RTCP (*RTP Control Protocol)* para transmissão de informações de controle e estatísticas e com protocolos de estabelecimento de sessão (ex: SIP - *Session Initiation Protocol*). Atua como um complemento da camada de transporte UDP, pois todo o pacote possui novo cabeçalho de transporte contendo os campos *Version* (versão do protocolo, atualmente 2), *Padding* (campo que indica se existe dados adicionais), *Extension* (campo que indica se existe o *Extension Header*), *CSRC Count* (quantidade de campos CSRC), *Marker* (campo que indica que o pacote é "especial" para a aplicação), *Payload Type* (*codec* em que está codificado os dados), *Sequence Number* (número sequencial do pacote), *Timestamp* (horário de envio do pacote), *SSRC* (identificador do remetente dos dados), *CSRC* (identificador dos contribuidores dos dados) e *Extension Header* (cabeçalho adicional, opcional).

Como um fluxo RTP não possui uma assinatura bem definida, sua identificação pelo Filtro RTP é feita através da análise de diversos pacotes buscando-se por padrões nos valores esperados no cabeçalho RTP (*Version*, *Sequence Number*, *Timestamp*, *Payload Type*, etc.) utilizando como base o algoritmo de Costeux et al. (2006). Uma vez identificado como fluxo RTP, o Filtro lê os pacotes (ArquivoSND e ArquivoRCV) na ordem em que foram capturados (ArquivoMTD) gerando um arquivo de áudio em formato batizado de VCA.

Além do arquivo de áudio, é gerado arquivo HTML com nome e *link* do arquivo de áudio e tempo de duração da conversa e um conjunto de metadados com Tipo de objeto = "VOIPRTP", Informação adicional = "Ligação Voip RTP: $Duração", Classificação = "Alerta", Carga = "ObjetoFile", Arquivo = nome do arquivo HTML gerado, Tipo de conteúdo = "text/html" e Observação = "Ligação Voip RTP: $Duração".

O arquivo VCA possui um cabeçalho geral composto de assinatura do arquivo e identificador do *codec* (codificador/decodificador) em que o áudio está codificado e vários registros de áudio compostos por canal (esquerdo ou direito), tamanho do áudio e o áudio propriamente dito.

Para tocar e converter as conversas de áudio foi desenvolvido a partir do projeto de código livre FFMPEG (Bellard, 2011) um módulo de leitura de arquivos VCA e integrado os *codecs* G.711 A-law e μ-law (ITU-T, 1988), MSRTAudio (Microsoft, 2006), G.723.1

(ITU-T, 2006), Siren (ITU-T, 2005b), G.722.1 (ITU-T, 2008a), GSM (ETSI, 2000), G.726 (ITU-T, 2005a) e G.729 (ITU-T, 2007) com base no trabalho de Ravnaas (2008). São fornecidos junto com este módulo o programa “FFPlay.exe” para tocar e o “FFMPEG.exe” para converter os arquivos VCA.

3.2.1.9 Filtro MSN

O Filtro MSN é responsável por decodificar conversas escritas e transferências de arquivos realizados através dos aplicativos MSNM (*MicroSoft Network Messenger*) e WLM (*Windows Live Messenger*). Os aplicativos utilizam o protocolo proprietário genericamente conhecido como MSNP (Zhenyu et al., 2005). Sua decodificação foi implementada baseado nas documentações do protocolo produzidas por Jennings et al. (2006), Khoshbakhtian et al. (2008), Movva e Lai (1999) e Rui et al. (2010) e Zhenyu et al. (2005), além de experiências de uso dos aplicativos e seu monitoramento.

As versões mais antigas os aplicativos MSNM e WLM fazem uma conexão TCP (porta 1863) com o servidor de mensagens para funções carregamento de lista de contatos, alteração de situação de presença dos contatos, etc. (conexão de presença) e uma nova conexão, também com o servidor de mensagens, para cada nova conversa realizada (conexão de conversa). As transferências de arquivos são realizadas através da mesma conexão da conversa ou através de nova conexão direta entre os participantes (conexão de transferência).

O Filtro MSN identifica a conexão de presença quando a primeira *string* transmitida no fluxo for “VER ”. Então varre o fluxo em busca da resposta ao comando “ADL” que é a lista de contatos. Neste caso é gerado um arquivo HTML com a lista de contatos e um conjunto de metadados com Tipo de objeto = “MSNLOGIN”, Informação adicional = “MSNLogin”, Classificação = “Normal”, Carga = “ObjetoFile”, Tipo de conteúdo = “text/html" e Arquivo = nome do arquivo HTML gerado.

A conexão de conversa é identificada quando a primeira *string* transmitida no fluxo for “JOI ”, “CAL ” ou “ANS ”. Então é varrido o fluxo em busca de comandos “MSG” e “SDG” com o parâmetro “Content-Type” igual a “text/plain” para mensagens digitadas e igual a “application/x-msnmsgrp2p” para dados de arquivos transferidos. Neste caso é

gerado um arquivo HTML com as mensagens digitadas, indicando seu autor e data/hora, um arquivo para cada transferência durante a conversa (*links* para estes arquivos estarão no arquivo HTML) e um conjunto de metadados com Tipo de objeto = “MSNCHAT”, Informação adicional = “Chat MSN”, Classificação = “Alerta”, Carga = “ObjetoFile”, Arquivo = nome do arquivo HTML gerado, Tipo de conteúdo = “text/html" e Observação = usuários que participaram da conversa.

A conexão de transferência é identificada quando a primeira *string* transmitida no fluxo for “<EOT><NULL><NULL><NULL>foo<NULL>”. Os dados são transferidos encapsulados em cabeçalho com 52 *bytes* (campos identificados: sessão, tamanho total, tamanho transferido, deslocamento). Os dados podem ser de negociação de transferência (campo sessão igual a zero) ou dados de arquivos (campo sessão diferente de zero). A negociação de transferência ocorre através de protocolo identificado como “MSNSLP” onde é informado o nome do arquivo a ser transferido e um identificador de sessão, além dos usuários envolvidos. Os dados de arquivos são remontados através do agrupamento dos dados referentes à mesma sessão, observando-se os demais campos do cabeçalho. Neste caso é gerado um arquivo para cada transferência identificada, um arquivo HTML com *links* para os arquivos transferidos indicando seu sentido (recebido ou enviado) e um conjunto de metadados com Tipo de objeto = “MSNP2P”, Informação adicional = “MSNP2P”, Classificação = “Alerta”, Carga = “ObjetoFile”, Tipo de conteúdo = “text/html", Arquivo = nome do arquivo HTML gerado e Observação = usuários e arquivos transferidos.

Nas versões mais recentes do MSNM e WLM, não é mais utilizada a conexão de conversa. Os aplicativos passaram a utilizar a conexão de presença para conversas e transferências de arquivos, sendo que a última continua a ocorrer preferencialmente na conexão de transferência.

Nessa situação o Filtro MSN busca na conexão de presença (primeira *string* transmitida no fluxo igual a “VER ”) além de comandos “ADL” (lista de contatos), os comandos “MSG” e “SDG” (mensagens digitadas). Para cada conversa identificada (grupo de mensagens trocadas entre os mesmos usuários) é gerado um arquivo HTML e um conjunto de metadados (mesmos campos/valores das versões anteriores). Transferências de arquivos

ocorridas na conexão de presença não são decodificadas, pois são precedidas por cabeçalho binário de tamanho variável com campos ainda não identificados.

Na conexão de transferência é buscada a mesma assinatura das versões antigas, mas os dados são transferidos encapsulados em cabeçalhos mais simples de tamanho variável (neste novo cabeçalho foram identificados os campos tamanho do pacote, tamanho do cabeçalho e número da sessão). O mesmo protocolo de negociação é utilizado (MSNSLP) e os arquivos podem ser reconstruídos pela simples concatenação de dados com a mesma identificação de sessão. Na Figura 3-21 é apresentada uma transferência do arquivo “arquivo.txt” com conteúdo “Conteúdo do arquivo.” do usuário “senlegen@hotmail.com” para o usuário “senlegen2@hotmail.com”, que é precedida por negociação através do protocolo MSNSLP (comandos INVITE e OK, dados “To”, “From”, “SessionID” e “Context” – nome do arquivo codificado em base64).

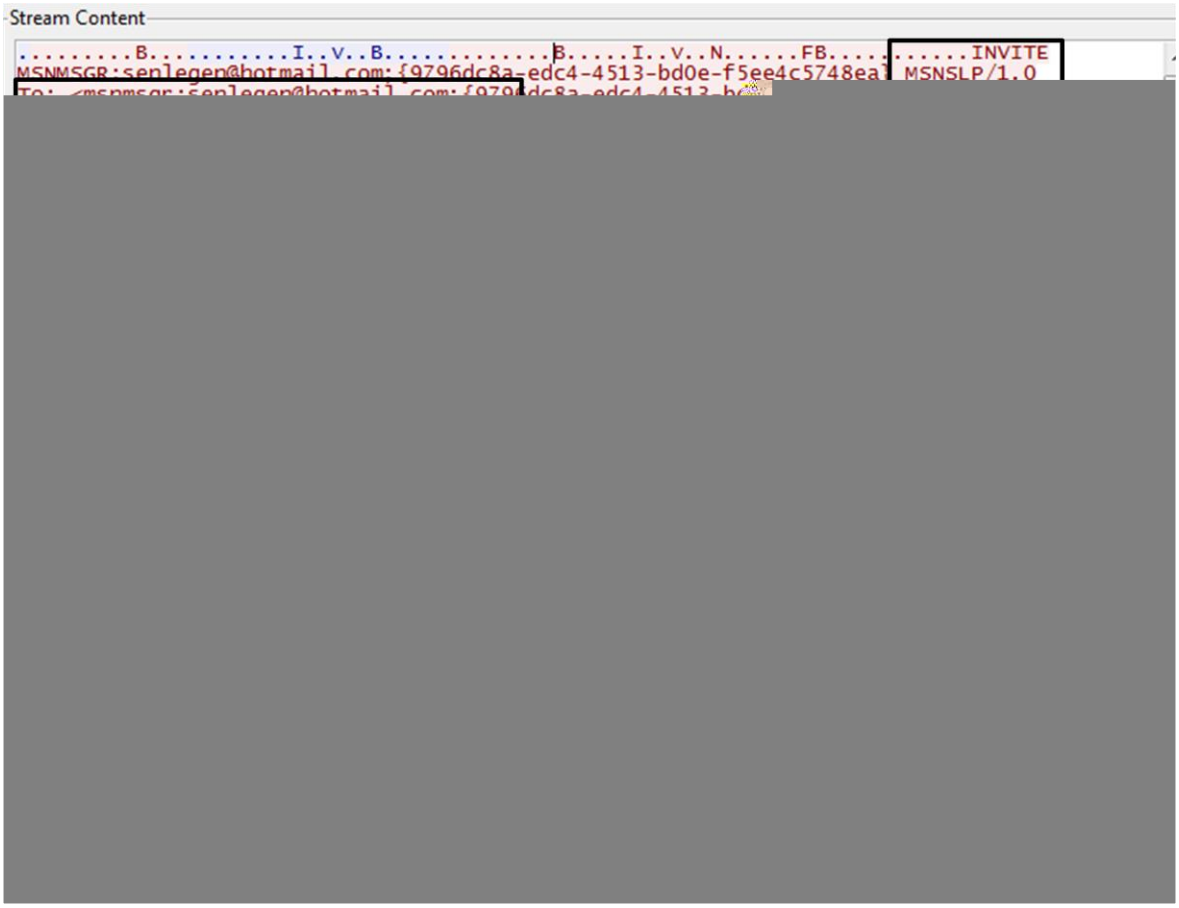


Figura 3-21 – Transferência de arquivo realizada através do WLM

Os aplicativos MSNM e WLM podem utilizar ainda o protocolo HTTP para tunelar o seu protocolo proprietário (Jennings et al., 2006 e Zhenyu et al., 2005), quando não for possível estabelecer conexão com o servidor de mensagens através de sua porta padrão (TCP/1863) por restrições de segurança (regras de *firewall*). A mesma técnica é utilizada pelas versões *web* desses aplicativos. O Filtro MSN decodifica as conversas realizadas através da remoção do cabeçalho HTTP de requisições e respostas de URLs contendo "live.com/gateway/gateway.dll?". Na Figura 3-22 é apresentada uma mensagem digitada "web3" do usuário "senlegen2@hotmail.com" para o usuário "senlegen@hotmail.com" através do comando "SDG" (protocolo MSNP) encapsulado em uma requisição HTTP/POST da URL "baymsg1020326.gateway.messenger.live.com/gateway/gateway.dll?SessionID=780667766.1008558244"

Conversas de áudio dos aplicativos MSNM e WLM são transferidas através do protocolo RTP e são decodificadas pelo Filtro RTP.

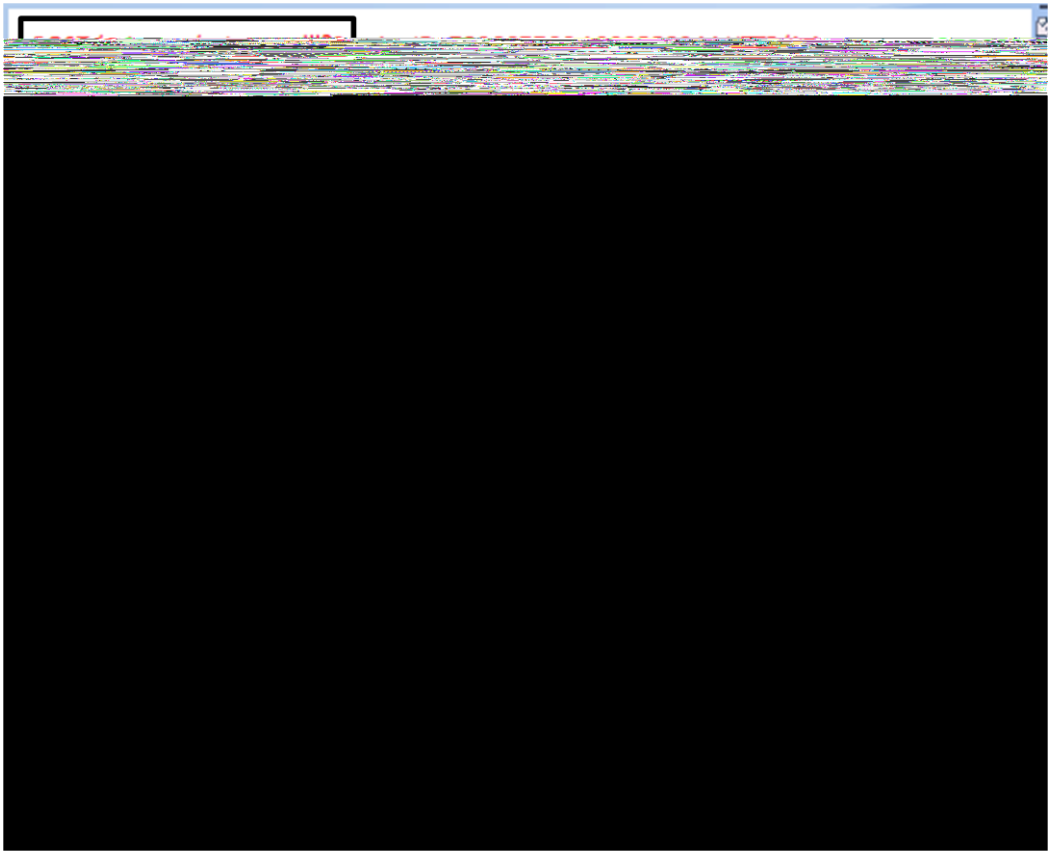


Figura 3-22 – Conversa digitada no aplicativo WLM encapsulada em HTTP

3.2.1.10 Filtro YMSG

O Filtro YMSG é responsável por decodificar conversas escritas realizadas através do aplicativo YahooMSG (*Yahoo!Messenger*). O YMSG utiliza o protocolo proprietário YMSG (Khoshbakhtian et al., 2008). Sua decodificação foi implementada baseado nas documentações do protocolo produzidas por Jennings et al. (2006), Khoshbakhtian et al. (2008), Tellis (2010) e Venky (2011), além de experiências de uso do aplicativo e seu monitoramento.

O YahooMSG faz uma conexão TCP com o seu servidor de mensagens (porta padrão 5500) para funções de autenticação, carregamento de lista de contatos, alteração de situação de presença dos contatos, envio e recebimento de mensagens, etc. Todo o dado transmitido nesta conexão é encapsulado em um cabeçalho de 20 *bytes* com os campos assinatura (valor fixo “YMSG”), versão, *vendor id*, tamanho do pacote, serviço, status e sessão. O campo serviço define o tipo de informação transmitida, sendo o valor “6” atribuído para mensagens escritas. Os dados são codificados em uma sequência de campos chave e valor com os caracteres separadores “<0xC0><0x80>”, sendo que para o serviço de mensagens escritas a chave “1” e “4” significam usuário de origem, chave “5” é o usuário de destino e a chave “14” é a mensagem digitada.

O Filtro YMSG identifica a conexão do YahooMSG quando a primeira *string* transmitida no fluxo for “YMSG”. Então varre os arquivos ArquivoSND e ArquivoRCV, obedecendo a ordem de captura dos pacotes no fluxo, em busca de cabeçalhos com valor “6” no campo serviço, decodificando os valores de usuário de origem, usuário de destino e texto digitado. Para cada conversa identificada (grupo de mensagens trocadas entre os mesmos usuários) é gerado um arquivo HTML e um conjunto de metadados com Tipo de objeto = “YMSGCHAT”, Informação adicional = “Chat YMSG”, Classificação = “Alerta”, Carga = “ObjetoFile”, Arquivo = nome do arquivo HTML gerado, Tipo de conteúdo = “text/html" e Observação = usuários que participaram da conversa.

Não são decodificas pelo Filtro YMSG implementado as transferências de arquivo realizadas pelo aplicativo YahooMSG, nem conversas escritas realizadas através de sua

versão *web*. Conversas de áudio são realizadas através do protocolo RTP, sendo decodificadas pelo Filtro RTP.

3.2.1.11 Filtro ICQ

O Filtro ICQ é responsável por decodificar conversas escritas e transferências de arquivos realizados através do aplicativo ICQ. O ICQ utiliza o protocolo proprietário OSCAR (*Open System for CommunicAtion in Real-time*) cuja documentação oficial não é disponível (Fritsch e Schiller, 2007). Sua decodificação foi implementada baseado nas documentações do protocolo produzidas por Clark (2006), Fritsch e Schiller (2007), Jennings et al. (2006) e Shutko (2005), além de experiências de uso do aplicativo e seu monitoramento.

O ICQ faz uma conexão TCP com o seu servidor de mensagens (porta padrão 5190) para funções de carregamento de lista de contatos, alteração de situação de presença dos contatos, envio e recebimento de mensagens, etc. (conexão principal) e, quando necessário, uma conexão para transferência de arquivos, que pode ser direta entre os usuários envolvidos na transferência (conexão P2P) ou através do servidor de mensagens (conexão *Proxy*).

Todo o dado transmitido na conexão principal é encapsulado em um cabeçalho de 6 *bytes* com os campos assinatura (valor fixo “0x2a”), canal, número de sequência e tamanho dos dados. O campo canal define o tipo de informação transmitida, sendo que o valor “1” indica início de sessão e o valor “2” indica outras transferências, que são encapsuladas em um novo cabeçalho de 10 *bytes* com os campos família, subtipo, *flags* e identificação, sendo que mensagens escritas possuem valor “4” no campo família e valor “6” ou “7” no campo subtipo.

Os arquivos são transferidos na conexão P2P através do protocolo proprietário OFT (*OSCAR File Transfer*) detalhadamente documentado em Clark (2006), formado basicamente por comandos de requisição de envio, aceitação e confirmação de recebimento. Os comandos possuem pelo menos 256 *bytes*, compostos por assinatura (valor fixo “OFT3”), tamanho do cabeçalho, comando (valor <0x0101> para requisição de envio, valor <0x0202> para aceitação, valor <0x0204> para confirmação de recebimento),

tamanho do arquivo, nome do arquivo, dentre outros. Na Figura 3-23 é exibido um exemplo da transferência do arquivo “arquivo.txt” com conteúdo “Conteúdo do arquivo.” (tamanho 20 *bytes* - <0x00000014>), onde se observa a solicitação de envio (comando OFT3 <0x0101>), a aceitação (comando OFT3 <0x0202>), o envio do conteúdo do arquivo e a confirmação de recebimento (comando OFT3 <0x0204>).

Na conexão *Proxy*, antes dos comandos OFT3 para transferência dos arquivos, ocorrem comandos de inicialização da conexão compostos por pelo menos 12 *bytes* com os campos tamanho, assinatura (valor fixo <0x044a>, comando (valor <0x0002> para inicialização de envio, valor <0x0004> para inicialização de envio e valor <0x0003> para reconhecimento), dentre outros.

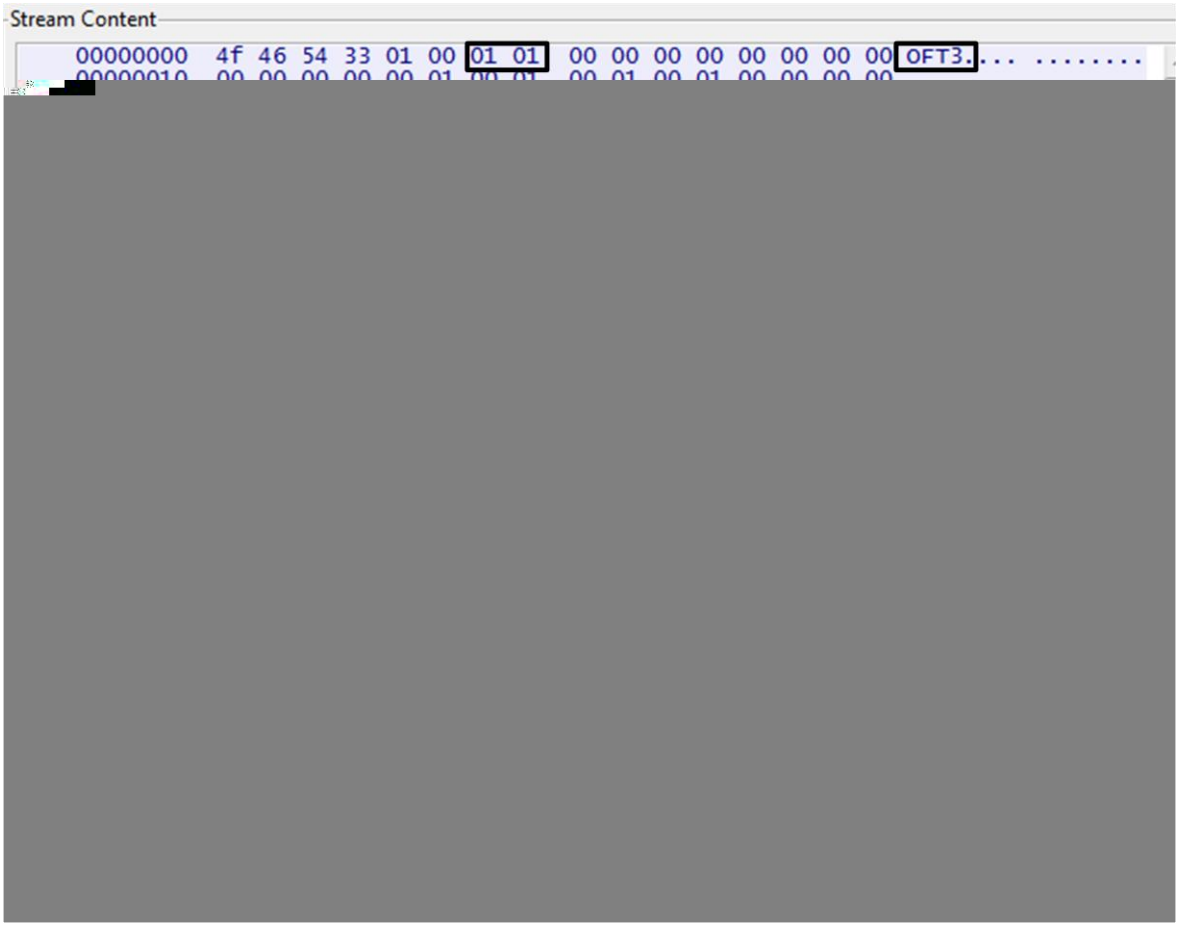


Figura 3-23 – Transferência de arquivo através do ICQ

O Filtro ICQ identifica a conexão principal quando os primeiros 10 *bytes* seguirem o padrão “<0x2a><0x01>??<0x00><0x04><0x00><0x00><0x00><0x01>” (comando de

início de sessão), onde “?” significa um *byte* com qualquer valor. Então varre os arquivos ArquivoSND e ArquivoRCV, obedecendo a ordem de captura dos pacotes no fluxo, em busca de cabeçalhos valor “2” no campo canal, valor “4” no campo família e valor “6” ou “7” no campo subtipo, decodificando os valores de usuário e texto digitado. Para cada conversa identificada (grupo de mensagens trocadas entre os mesmos usuários) é gerado um arquivo HTML e um conjunto de metadados com Tipo de objeto = “ICQ”, Informação adicional = “Chat ICQ”, Classificação = “Alerta”, Carga = “ObjetoFile”, Arquivo = nome do arquivo HTML gerado, Tipo de conteúdo = “text/html" e Observação = usuários que participaram da conversa.

A conexão P2P é identificada pela *string* inicial “OFT3” no fluxo e a conexão Proxy é identificada quando os primeiros 10 *bytes* seguirem o seguinte padrão “??<0x04><0x4a><0x00>#” (comando de inicialização de envio ou recebimento), onde “?” significa um *byte* com qualquer valor e “#” um *byte* com valor 2 ou 4. Então varre os arquivos ArquivoSND e ArquivoRCV, obedecendo a ordem de captura dos pacotes no fluxo, em busca do conteúdo dos arquivos que são transferidos logo após o comando OFT de aceite. Neste caso é gerado um arquivo para cada transferência identificada, um arquivo HTML com *links* para os arquivos transferidos indicando seu sentido (recebido ou enviado) e um conjunto de metadados com Tipo de objeto = “ICQ”, Informação adicional = “ICQ Transf. Arq.”, Classificação = “Alerta”, Carga = “ObjetoFile”, Tipo de conteúdo = “text/html", Arquivo = nome do arquivo HTML gerado e Observação = nome dos arquivos transferidos.

Não são decodificas pelo Filtro ICQ conversas escritas realizadas através de sua versão *web*. Conversas de áudio são realizadas através do protocolo RTP, sendo decodificadas pelo Filtro RTP.

3.2.1.12 Especificação de Filtros Externos

Filtros Externos são programas de decodificação de protocolos que podem ser integrados ao programa Importador. Para configurar sua execução deve-se adicionar uma linha no arquivo “Filtros.txt” na pasta do programa Importador contendo o comando a ser executado. Todos os Filtros Externos são executados para cada Arquivo de Fluxo identificado.

O Filtro Externo deve ser desenvolvido obedecendo à seguinte especificação de entrada e saída para funcionar adequadamente:

- Receber como parâmetros de linha de comando a pasta inicial (pasta onde está o banco de dados da conexão monitorada, sendo que nomes de arquivos com caminho devem ser relativos a esta pasta) e o nome do Arquivo de Fluxo a ser decodificado (com caminho relativo a pasta inicial);
- Produzir arquivo de resposta (nome do Arquivo de Fluxo recebido concatenado com a *string* ".resp") contendo os nomes dos arquivos de metadado gerados, um por linha. O arquivo de metadado deve conter pares "nome do campo=valor do campo", um por linha, usando os nomes dos campos da tabela INDEXDAT, detalhados na seção 3.2.1.2. Caso não identifique o fluxo, o Filtro Externo pode finalizar sem a criação do arquivo de resposta.

O Filtro Externo também deve seguir as seguintes especificações:

- Não solicitar qualquer informação ao usuário, podendo escrever mensagens na saída padrão;
- Utilizar a pasta "log" para gravação de arquivos de registro de informações de depuração, erros e auditoria;
- Gravar os arquivos de decodificação sob a pasta "dados"\data do fluxo\identificador do Filtro, e utilizar como base o nome do Arquivo de Fluxo recebido, devendo sobrescrevê-los quando o mesmo fluxo for decodificado novamente;
- Utilizar o campo "Info" dos metadados para identificar conteúdo por ele produzido.

Na Figura 3-24 é mostrado um exemplo de configuração e execução de um programa Filtro Externo com nome "c:\CLIT\FiltroPOP.exe":

- Conteúdo do arquivo "Filtros.txt" em a);
- Chamada ao FiltroPOP.exe pelo programa Importador com os parâmetros pasta inicial e Arquivo de Fluxo em b);
- Arquivo de resposta "str\20110831\192.168.9.83-51427-67.195.133.185-110.txt.resp" produzido pelo FiltroPOP.exe contendo o nome dos dois arquivos de metadados gerados em c);

- Conteúdo do arquivo de metadado “dados\20110831\pop\192.168.9.83-51427-67.195.133.185-110.31082011190929948345.001.eml.html.ini” produzido pelo FiltroPOP.exe em d).

```
a) Conteúdo do arquivo “Filtros.txt”

C:\CLIT\FiltroPOP.exe

b) Chamada ao “Filtro.exe”

C:\CLIT\FiltroPOP.exe c:\CLIT\Alvo1\ str\20110831\192.168.9.83-51427-
67.195.133.185-110.txt

c) Conteúdo do Arquivo de resposta

dados\20110831\pop\192.168.9.83-51427-67.195.133.185-110.001.html.ini
dados\20110831\pop\192.168.9.83-51427-67.195.133.185-110.002.html.ini

d) Conteúdo do arquivo de metadado

TipoObj=EMAIL
Info=FE:Email recebido(POP)
Coment=De: Senlegen <senlegen@yahoo.com.br> / Para: senlegen@hotmail.com
/ Assunto: Assunto digitado.
Status=Alerta
Web=N
Arquivo=dados\20110831\pop\192.168.9.83-51427-67.195.133.185-
110.001.html
ContentType=text/html
ipOrig=192.168.9.83
ptOrig=51427
ipDest=67.195.133.185
ptDest=110
Prot=TCP
DataHoraIni=31/08/2011 19:09:29
DataHoraFim=31/08/2011 19:09:44
```

Figura 3-24 – Exemplo de configuração e execução de um programa Filtro Externo

3.2.2 Programa Analisador

O programa Analisador é o programa que permite ao Investigador analisar os objetos decodificados pelos módulos Filtros dos arquivos de captura, cujos metadados estão em um banco de dados que apontam para seu conteúdo no sistema de arquivos. É composto pelos módulos Interface e ProxyWebServer (Figura 3-25).

O módulo ProxyWebServer é responsável por atender todas as requisições HTTP feitas pelo módulo Interface, que são sempre respondidas com conteúdo do sistema de arquivos

ou com mensagens de erro, impedindo requisições diretamente ao servidores reais na Internet.

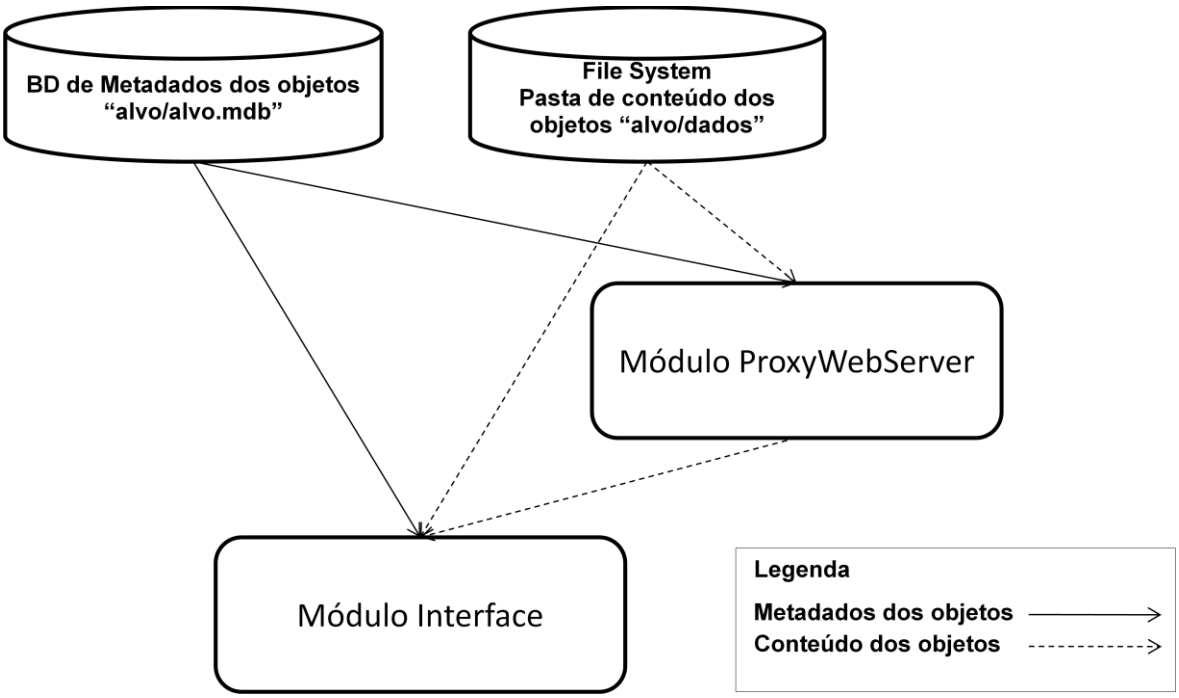


Figura 3-25 – Módulos do programa Analisador

O módulo Interface (Figura 3-26) é composto por:

- Menu Principal: possui comandos para escolha/abertura de banco de dados da conexão monitorada, exibição de informações sobre o programa e encerramento;
- Tabela de Metadados: exibe os metadados dos objetos (um por linha) carregados da tabela INDEXDAT do banco de dados aberto que atendam os critérios estabelecidos no Painel de Filtragem. O conteúdo do objeto selecionado é exibido no Visualizador de Objetos. Possui campo para digitação de observação, comandos (menu de contexto e teclas de atalho) de alteração da classificação do objeto ("Importante" e "Não importante"), exportação de um ou de todos objetos listados para imagens JPEG (*Joint Photographic Experts Group*);
- Painel de Filtragem: permite a escolha de critérios de filtragem de objetos a serem exibidos na Tabela de Metadados (*Content-Type*, URL, Classificação, Observação e Período). Campos de texto podem utilizar o caractere coringa "*" no início, meio ou fim para indicar qualquer *string*;

- Visualizador de Objetos (VO): exibe o conteúdo do objeto selecionado na Tabela de Metadados em um navegador *web*.

Figura 3-26 – Interface do Programa Analisador

O conteúdo dos objetos pode ser carregado no VO de duas formas, de acordo com o valor do campo Carga dos seus metadados. Objetos com valor “ObjetoFile” no campo Carga são carregados no VO diretamente do sistema de arquivos baseado no seu campo Arquivo. Já objetos com valor “ObjetoWeb” no campo Carga são carregados no VO através de requisição HTTP/GET da URL armazenada em seu campo URL. A requisição é atendida pelo módulo ProxyWebServer do programa Analisador, que responde com o conteúdo do arquivo do objeto selecionado. Como o objeto *web* carregado no navegador pode solicitar outros objetos *web* (URLs diferentes do objeto selecionado) tais com imagens, *scripts* e folhas de estilo, o ProxyWebServer busca por todos os objetos que possuam a URL solicitada e adota a seguinte política de resposta:

- Nenhum objeto encontrado: responde com “Objeto não encontrado” (HTTP Status 404);

- Um objeto encontrado: responde com o conteúdo do arquivo do objeto;
- Diversos objetos encontrados: responde com o conteúdo do arquivo do objeto escolhido conforme a seguinte prioridade: (1) objeto cuja origem seja o mesmo fluxo do objeto selecionado; (2) objeto mais próximo com data/hora posterior, mas não maior que 30 segundos, ao objeto selecionado; (3) objeto mais próximo com data/hora inferior ao objeto selecionado.

Essa política de prioridade foi adotada baseando-se na observação do comportamento de navegadores de Internet que, ao usuário selecionar uma URL, solicita ao servidor *web* correspondente através de conexão TCP o recurso indicado e, de sua análise, solicita outros recursos que o compõem através da mesma conexão TCP (apenas recursos que estão no mesmo servidor *web*) ou em novas conexões TCP logo após a primeira, sendo que recurso disponíveis no seu *cache* que estejam válidos, ou seja, baixados recentemente, não são solicitados.

A forma de carregamento de ObjetosWeb foi desenvolvida com o intuito de permitir a visualização de páginas *web* de forma rápida e mais próxima possível ao visualizado na conexão interceptada, se mostrando bastante eficiente mesmo em páginas complexas (objetos de diversos sítios, *scripts*, páginas de estilo, etc.), mas limitada em situações de página carregadas parcialmente através de tecnologias de conteúdo dinâmico (ex.: AJAX - *Asynchronous Javascript and XML*). Apresenta ainda como vantagens a simulação do *cache* do navegador da conexão monitorada, já que sem essa função muitas páginas seriam exibidas de forma incompleta (sem algumas imagens e outros objetos) e também permite que o Investigador navegue pelo conteúdo interceptado clicando nos *links* visitados pelo investigado. Uma solução alternativa é apresentada em Cohen (2008) através da sanitização dos documentos HTML (remoção de *tags* indesejáveis a ajuste nas URLs do objetos vinculados).

3.3 METODOLOGIA DE INVESTIGAÇÃO COM SIT E CLIT

Com a utilização do SIT e o CLIT em um OICB, uma nova metodologia de investigação passa a ser aplicada com procedimentos mais simples dentro das fases de Procedimentos Iniciais, Rotineiros e Finais nas ICIs, conforme apresentado na Tabela 3.4.

Tabela 3.4 – Metodologia Atual x Metodologia SIT/CLIT

	Metodologia Atual						**Metodologia SIT/CLIT**	
	SFTPServer		**SFTPClient**			**Enc. Tráfego**	**SFTPServer**	**SFTPClient / Enc.**
Proc. Iniciais	Envio Mandado Config. Remota Config. Local (Cliente SFTP)		Config. Local (Servidor SFTP) Envio Mandado Config. Remota			Envio Mandado Config. Remota Config. Local (sniffer)	Envio Mandado Config Remota Config Local (CLIT)	Pedido Conta SIT Config SIT Envio Mandado Config Remota Config Local (CLIT)
Proc. Rotineiros	**Pcap** Obtenção (Cliente SFTP) Importação (NFAT) Análise (NFAT, Word)	**Pcap c/ encap**. Obtenção (Cliente SFTP) Tratamento (bittwiste) Importação (NFAT) Análise (NFAT, Word)	**Pcap** Importação (NFAT) Análise (NFAT, Word)	**Snoop** Tratamento (editcap) Importação (NFAT) Análise (NFAT,Word)	**ETSI** Incompat.	**Pcap c/ encap**. Tratamento (bittwiste) Importação (NFAT) Análise (NFAT, Word)	**Qualquer formato de arq. ou encap.** Análise (CLIT)	
Proc. Finais	Relatório (NFAT, Word) Preservação						Relatório (CLIT, Word) Preservação	

3.3.1 Procedimentos Iniciais

Os Procedimentos Iniciais variam dependendo da forma de entrega de dados pela operadora. Para operadoras que entregam na forma SFTPServer, as etapas são:

1. Envio de Mandado: Investigador encaminha para operadora pedido de ICI acompanhado de mandado judicial;
2. Configuração Remota: operadora cria conta SFTP em seu servidor de interceptação; configura programa de captura para armazenar os pacotes que trafegam na conexão solicitada em arquivos de captura de determinado tamanho (20MB normalmente) na pasta *home* da conta SFTP criada; informa dados de acesso aos arquivos (IP/porta/usuário/senha);
3. Configuração Local: Investigador cria pasta para a nova interceptação e configura o programa CLIT com as informações da conta SFTP e pasta criada e sua execução automática.

Para operadoras que entregam na forma SFTPClient, as etapas são:

1. Pedido Conta SIT: Investigador envia pedido de criação de SFTP para interceptação a equipe gestora do SIT;
2. Configuração SIT: equipe gestora do SIT cria conta SFTP e informa dados de acesso (IP/porta/usuário/senha);
3. Envio de mandado: Investigador encaminha para a operadora pedido de ICI acompanhado de mandado judicial, informando dados de acesso (IP/porta/usuário/senha) para envio dos arquivos de captura;

4. Configuração remota: operadora, em seu servidor de interceptação, configura programa de captura para armazenar os pacotes que trafegam na conexão solicitada em arquivos de captura de tamanho pequeno (50KB ou 500KB normalmente) em determinada pasta; configura programa que varre esta pasta a cada intervalo de tempo e os envia para a pasta *home* do usuário do servidor SFTP informado pelo Investigador. Após enviados, os arquivos são imediatamente apagados do servidor da operadora. O mesmo ocorre com arquivos que não puderam ser enviados após determinadas tentativas sem sucesso por problemas no servidor SFTP informado;
5. Configuração Local: Investigador cria pasta para a nova interceptação e configura o programa CLIT com as informações da conta SFTP e pasta criada e sua execução automática.

Para operadoras que entregam na forma encapsulamento de tráfego, as etapas são:

1. Pedido Conta SIT: Investigador envia pedido de criação de SFTP para interceptação, indicando se tratar de ICI com entrega de tráfego na forma encapsulamento de tráfego, a equipe gestora do SIT;
2. Configuração SIT: equipe gestora do SIT cria conta SFTP, reserva um IP da faixa disponível e ativa programa de captura para gravar os pacotes destinados ao IP reservado na pasta *home* da conta SFTP criada. Informa dados de acesso (IP/porta/usuário/senha) e IP para envio dos pacotes espelhados;
3. Envio de mandado: Investigador encaminha para a operadora pedido de ICI, acompanhado de mandado judicial, informando o IP para envio dos pacotes espelhados;
4. Configuração remota: operadora, em seu servidor de interceptação, configura programa para copiar os pacotes que trafegam na conexão solicitada para envio para o IP informado (o pacote IP copiado é enviado na área de dados de um pacote IP/GRE ou outro encapsulamento onde o endereço de origem é o IP do equipamento da operadora e o endereço de destino é o IP informado pelo Investigador);
5. Configuração Local: Investigador cria pasta para a nova interceptação e configura o programa CLIT com as informações da conta SFTP e pasta criada e sua execução automática.

3.3.2 Procedimentos Rotineiros

Os Procedimentos Rotineiros passam a ser compostos por uma etapa, já que as etapas de Obtenção, Tratamento e Importação são realizadas de forma automática pelo programa Importador do CLIT:

1. Análise: Investigador executa o programa Analisador e abre o arquivo de banco de dados da ICI, seleciona apenas os itens com Status igual a “Não Analisados”, visualiza em ordem cronológica as informações interpretadas pelo programa de análise (páginas *web* acessadas, *e-mails*, conversas realizadas em comunicadores instantâneos, etc.), classifica os itens analisados como “Importante” ou “Não Importante” para a investigação e registra anotações pertinentes.

3.3.3 Procedimentos Finais

Os Procedimentos Finais, realizados no final do prazo de interceptação, continua a apresentar as etapas de Relatório e Preservação, sendo que a última permanece como antes:

1. Relatório: Investigador, através de comando no CLIT, seleciona todos os objetos marcados como “Importantes”, exporta-os como imagens e insere-as no seu relatório no programa editor de textos, registra seus comentário, podendo realizar novas buscas nos metadados dos objetos no CLIT a fim de localizar novas informações relevantes. No relatório, o Investigador também conclui pelo pedido de renovação ou interrupção da interceptação;
2. Preservação: Investigador gera mídia não regravável com os arquivos de captura originais e calcula os seus *hashs*, que são listados no relatório. A mídia passa a ser anexo do relatório durante todo o processo legal.

3.3.4 Ganhos da Metodologia Proposta

Os Procedimentos Iniciais continuam variando dependendo da forma de entrega de dados pela operadora apenas nas suas etapas administrativas (envio e recebimento de pedidos e mandados), já que a etapa de Configuração Local (etapa técnica) passa a ser única, podendo ser facilmente realizada por Investigador com conhecimentos básicos de informática.

Nos Procedimentos Rotineiros, que é a fase repetitiva do processo e, portanto, mais desgastante para o Investigador, não existem mais as etapas de Obtenção, Tratamento e Importação, já que o CLIT faz *download* dos arquivos de captura, é compatível com os formatos de arquivos e pacotes enviados pelas operadoras brasileiras e faz o gerenciamento de arquivos importados/não importados de forma automática. A etapa de Análise também é simplificada, já que o CLIT permite que o Investigador selecione todos os objetos ainda não classificados, ou seja, o ponto em que parou sua análise, evitando que o Investigador necessite anotar o seu ponto de parada. Na Análise também não há mais a necessidade de ficar montando o relatório, já que o CLIT permite a classificação dos objetos e a anotação de informações pertinentes, que poderão ser recuperadas na etapa de Relatório dos Procedimentos Finais.

Nos Procedimentos finais a etapa de relatório é simplificada, pois o Investigador pode, através de comando no CLIT, selecionar todos os objetos marcados como “Importantes”, exporta-os como imagens e insere-as no seu relatório no programa editor de textos.

4 EXPERIMENTOS E RESULTADOS

O objetivo principal do projeto desenvolvido é a simplificação do uso de ICI para os Investigadores nos OICBs a fim de aumentar seu uso nas investigações. Uma avaliação ideal seria uma análise na variação da adoção do uso de ICIs nas investigações a partir do uso das novas ferramentas, mas isso envolveria um tempo maior de avaliação, pois demandaria também treinamento e convencimento de usuários para seu uso.

A avaliação da solução proposta foi realizada então com base nos objetivos específicos apresentados na Introdução deste trabalho. Os seguintes critérios foram estabelecidos:

1. Abrangência da análise: analisa se todos os pacotes de rede recebidos puderam ser processados, podendo receber os valores “Total” (todos os pacotes são processados), “Parcial” (parte dos pacotes foram descartados durante o processo de conversão ou edição de arquivos e/ou pacotes), “Nenhuma” (nenhum pacote foi processado por algum tipo de incompatibilidade);
2. Confiabilidade de recebimento do tráfego: analisa os riscos de perda de informações enviadas pelas operadoras devido à infraestrutura do OICB para recebimento, podendo receber os valores “Alta” (entrega SFTPServer ou entrega SFTPClient com infraestrutura confiável), “Média” (entrega SFTPClient com infraestrutura não confiável ou entrega Encapsulamento de Tráfego com infraestrutura confiável) e “Baixa” (entrega Encapsulamento de Tráfego com infraestrutura não confiável). Infraestrutura confiável refere-se ao uso de computadores servidores, boas condições de refrigeração e energia ininterrupta e de *links* de Internet com banda garantida e suficiente. Infraestrutura não confiável é quando algum destes fatores não é garantido;
3. Quantidade de processos não automatizados: analisa a quantidade de procedimentos manuais a serem realizados pelo Investigador para analisar o tráfego, podendo receber valores de zero a *n*;
4. Quantidade de controles manuais: analisa a quantidade de controles que devem ser mantidos pelo Investigador de forma a analisar todo o tráfego, podendo receber valores de zero a *n;*

5. Quantidade de programas externos: analisa a quantidade de programas/*scripts* que além da NFAT precisam ser manipuladas pelo Investigador, podendo receber valores de zero a *n*;
6. Acompanhamento em tempo real: analisa se o tráfego pode ser analisado em tempo real, fator imprescindível em determinados tipos de investigações, podendo receber os valores “sim” ou “não”;
7. Padronização dos procedimentos: analisa se os procedimentos apresentam grande variação nas diferentes ICIs;
8. Aumento de quantidade de vestígios extraídos, especialmente de protocolos/aplicativos de uso comum no Brasil.

Foi comparada a Metodologia Atual (forma atual de trabalho considerando a infraestrutura e ferramentas disponíveis) com a Metodologia CLIT/SIT (nova forma de trabalho considerando a infraestrutura e ferramentas propostas) em 6 ICIs que envolvem diferentes formas de disponibilização de tráfego apresentadas na Tabela 4.1. Os critérios 1 a 6 foram analisados individualmente em cada experimento. O critério 7 é analisado globalmente, pois envolve a comparação na variação dos procedimentos nos 6 experimentos. O critério 8 também é analisado globalmente, pois independe da forma de disponibilização de tráfego.

As 6 ICIs experimentadas são simulações de interceptações reais escolhidas de forma a representar todas as combinações de disponibilização de tráfego atualmente adotadas pelas grandes operadoras. Os experimentos não foram realizados em ICIs reais devido aos rigores da Lei da Interceptação e normativos internos do OICB quanto à manipulação de dados interceptados, que limita o acesso a apenas Investigadores ligados diretamente a investigação. Esse fator comprometeu a avaliação do critério “Aumento de quantidade de vestígios extraídos”, pois o tráfego simulado não representa o tráfego real de ICIs em diferentes tipos de investigação.

4.1 AMBIENTE DOS EXPERIMENTOS

Os experimentos foram realizados em uma unidade descentralizada e no *datacenter* localizado no órgão central de um OICB.

Tabela 4.1 – Formas de disponibilização do tráfego nas ICIs utilizadas nos experimentos

Experimento	**Formas de disponibilização do tráfego**			
	Forma de Entrega	**Formato arquivo de captura**	**Enlace**	**Encapsulamento**
ICI A	SFTPServer	Pcap	Ethernet	-
ICI B	SFTPServer	Pcap	Ethernet	PCLI, IP/UDP ou TZSP
ICI C	SFTPClient	Pcap	Ethernet	-
ICI D	SFTPClient	Snoop	Ethernet	-
ICI E	SFTPClient	ETSI	-	-
ICI F	Encapsulamento de Tráfego	Pcap	Ethernet/PPPoE	GRE ou Juniper

Para análise da Metodologia Atual, foi utilizado ambiente comumente utilizado nas ICIs do OICB, composto de:

- Computador *desktop* com 1 TB de HD, 4 GB de memória RAM;
- Programas Windows (sistema operacional), NetResident (NFAT), WinSCP (cliente SFTP), Windump (programa de captura), freeFTPd (servidor SFTP), MSWord (editor de textos), Editcap (conversor de formatos de arquivos de captura) e BitTwiste (programa editor de pacotes);
- Conexão à intranet do OICB com velocidade de 2 Mbps ao *datacenter*, que permite acesso à Internet saindo com o IP "IP_OICB";
- Conexão de Internet ADSL com velocidade de 1 Mbps, IP fixo "IP_fixo_ADSL" e modem ADSL que opera nos modos *router* ou *bridge*;
- Infraestrutura de energia estabilizada do edifício da unidade descentralizada com autonomia de 10 minutos em caso de falta de energia e de climatização central ativo apenas no horário comercial.

Para análise da Metodologia CLIT/SIT, foi utilizado o mesmo ambiente da Metodologia Atual para o CLIT. Para o SIT foi preparado o seguinte ambiente:

- Computador servidor com 4 TB de HD, 12 GB de memória RAM;

- Programas Linux (sistema operacional), OpenSSH (servidor SFTP) e Tcpdump (programa de captura);
- Conexão à Internet através de ligação ao *backbone* do OICB, que possui *links* redundantes com altos níveis de qualidade de serviço garantidos por contrato (garantia de banda, redundância, alta disponibilidade, etc.) e faixa de 100 IP fixos reservados para uso das interceptações;
- Infraestrutura da sala cofre do *datacenter* da OICB, com *no-breaks*, geradores e climatização com sistemas redundantes.

4.2 EXPERIMENTO ICI A

Neste experimento são comparadas as metodologias em uma ICI de tecnologia ADSL da operadora A (ICI A), que entrega o tráfego na forma SFTPServer, arquivos de captura no formato pcap de tamanho de 20MB apresentando pacotes IP com enlace Ethernet e sem encapsulamento IP.

4.2.1 Metodologia Atual

Para a Metodologia Atual temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ICI encaminhando mandado para a operadora e informando o IP de acesso “IP_OICB”;
2. Operadora, após fazer as devidas configurações nos seus equipamentos, responde à solicitação informando a conta SFTP de acesso ao tráfego da ICI solicitada (servidor SFTP: “IP_SFTP_Op”, porta: “porta_SFTP_Op”, usuário: “conta_ICI” e senha: “senha_ICI”). Os arquivos de captura são disponibilizados na pasta *home* do usuário criado, sendo que o arquivo aberto (arquivo que não alcançou o tamanho limite e continua recebendo pacotes do programa de captura) também fica disponível para ser baixado;
3. Investigador cria as pastas “C:\intercept\OpA\AlvoA\não importados”, “C:\intercept\OpA\AlvoA\já importados” e “C:\intercept\OpA\AlvoA\NFAT”;
4. Configura o programa WinSCP com os dados de acesso a conta SFTP (Figura 4-1) e pasta local para baixar os arquivos;
5. Configura o programa NetResident para armazenar os dados desta ICI na pasta “C:\intercept\OpA\AlvoA\NFAT”.

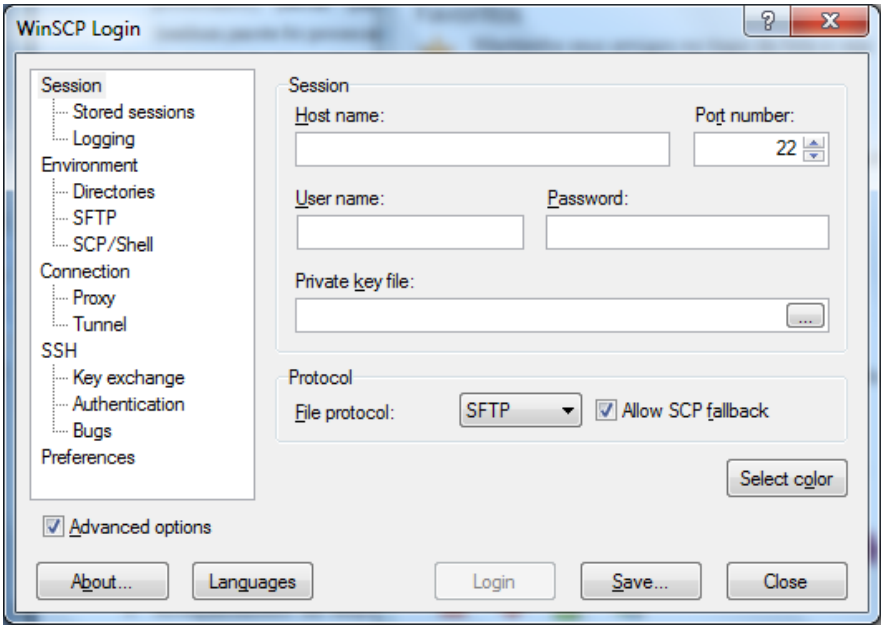


Figura 4-1 – Programa WinSCP: criação de perfil de acesso à conta SFTP

Os Procedimentos Rotineiros são:

1. Investigador, utilizando o programa WinSCP, se conecta ao servidor da operadora e baixa os arquivos de captura ainda não baixados (a verificação é manual: inspeção visual das pastas “já importados” e “não importados”);
2. Investigador, utilizando o programa NetResident, importa os arquivos que estão na pasta “não importados”, exceto o arquivo de captura aberto. Os arquivos importados são então movidos pelo Investigador para a pasta “já importados”. A importação do arquivo de captura aberto gera o inconveniente de gerar objetos duplicados quando o mesmo arquivo tiver que ser importado novamente quando o receber mais pacotes, pois a ferramenta não os reconhece as duplicidades nem arquivos parcialmente importados;
3. Investigador, utilizando o programa NetResident, visualiza o conteúdo de cada objeto identificado em ordem cronológica. Cada objeto julgado como relevante é relatado em documento do MSWord. Como a ferramenta não possui uma forma de exportação simples, são utilizados mecanismos de copiar e colar textos ou mesmo imagens através de comandos “*printscreen*”. Ao terminar a análise, o Investigador anota a data/hora do último objeto analisado a fim de retomar o trabalho no ponto onde parou quando novos arquivos de captura forem importados, pois a ferramenta não tem mecanismo de marcação de itens já analisado ou pendentes de análise.

Os Procedimentos Finais são:

1. Investigador consolida seu relatório fazendo novas buscas na ferramenta NetResident e em suas anotações, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.2.2 Metodologia SIT/CLIT

Para a Metodologia SIT/CLIT temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ICI encaminhando mandado para a operadora e informando o IP de acesso “IP_OICB”;
2. Operadora, após fazer as devidas configurações nos seus equipamentos, responde à solicitação informando a conta SFTP de acesso ao tráfego da ICI solicitada: servidor SFTP: “IP_SFTP_Op”, porta: “porta_SFTP_Op”, usuário: “conta_ICI” e senha: “senha_ICI”. Os arquivos de captura são disponibilizados na pasta *home* do usuário criado, sendo que o arquivo aberto (arquivo que não alcançou o tamanho limite e continua recebendo pacotes do programa de captura) também fica disponível para ser baixado;
3. Investigador cria a pasta “C:\intercept\OpA\AlvoA\pcaps” para armazenamento dos arquivos de captura e nela o arquivo de *download* automático (“.script.txt”) com o conteúdo apresentado na Figura 4-2-a;
4. Investigador cria arquivo de *script* (arquivo c:\intercept\importador.bat) para execução do programa Importador a cada 60 segundos informando a pasta local dos arquivos de captura e o arquivo de banco de dados da ICI, conforme apresentado na Figura 4-2-b;
5. Investigador executa o arquivo “c:\intercept\importador.bat”.

Os Procedimentos Rotineiros são:

1. Investigador executa o programa Analisador e abre o arquivo de banco de dados da ICI (c:\intercept\OpA\AlvoA\AlvoA.mdb), seleciona apenas os itens com Status igual a “Não Analisados”, visualiza cada um dos itens, registra anotações e marca-os como “importante” ou “não importante” para a investigação.

a) Script de *download* automático de arquivos de captura

```
open sftp://usuario_ICI:senha_ICI@IP_SFTP_Op:porta_SFTP_Op
lcd C:\intercept\OpA\AlvoA\pcaps
get *
bye
```

b) Script de execução automática do programa importador

```
:inicio
C:\intercept\importador.exe c:\intercept\OpA\AlvoA\pcaps  \
                            c:\intercept\OpA\AlvoA\AlvoA.mdb –auto -now
Goto inicio
```

Figura 4-2 – *Scripts* de *download* e execução do programa Importador

Os Procedimentos Finais são:

1. Investigador seleciona todos os itens marcados como “Importante” no programa Analisador e os exporta em formato de imagens e as arrasta para dentro de um novo documento do MSWord. Neste documento realizam-se os relatos baseado nos comentários já informados na ferramenta. Então consolida o relatório fazendo novas buscas na ferramenta, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.2.3 Comparação

Comparando as metodologias com base nos critérios estabelecidos, temos:

1. Abrangência da análise: ambas as metodologias têm abrangência “Total” da análise (existe total compatibilidade das ferramentas com os formatos de arquivos e pacotes);
2. Confiabilidade de recebimento do tráfego: “Alta” para ambas as metodologias (inerente a entrega de tráfego SFTPServer);
3. Quantidade de processos não automatizados: dois para a Metodologia Atual (*download* e importação de arquivos de captura) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* e gerenciamento automático dos arquivos “já importados” / “não importados”);
4. Quantidade de controles manuais: três para a Metodologia Atual (arquivos “baixados” / “não baixados”, arquivos “já importados” / “não importados” e de

ponto de parada da análise) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* automático apenas de arquivos não baixados, importa de forma automática apenas arquivos de captura ainda não importados e permite que o Investigador marque os objetos já analisados indicando com clareza o ponto onde parou a análise anterior);

5. Quantidade de programas externos: dois para a Metodologia Atual (WinSCP e MSWord) e um para a Metodologia SIT/CLIT (MSWord);
6. Acompanhamento em tempo real: "Não" para a Metodologia Atual (Investigador deve esperar arquivo de captura "encher", ou seja, alcançar limite de 20MB, para então baixá-lo e importá-lo) e "Sim" para a Metodologia SIT/CLIT (CLIT pode baixar do servidor SFTP o arquivo aberto pelo programa de captura em qualquer momento e, em uma próxima importação do mesmo arquivo, continuar a importação a partir do último pacote IP analisado anteriormente, sem causar duplicação de objetos extraídos).

Conforme resumo apresentado na Tabela 4.2, para este cenário a Metodologia SIT/CLI apresentou resultados vantajosos em quatro critérios, sendo que manteve os mesmos resultados da Metodologia Atual nos dois critérios em que não eram possíveis evoluções.

Tabela 4.2 – ICI A: comparação das metodologias

Critério	**Metodologia Atual**	**Metodologia SIT/CLIT**
Abrangência da análise	Total	Total
Confiabilidade de recebimento do tráfego	Alta	Alta
Quantidade de processos não automatizados	2	0
Quantidade de controles manuais	3	0
Quantidade de programas externos	2	1
Acompanhamento em tempo real	Não	Sim

4.3 EXPERIMENTO ICI B

Neste experimento são comparadas as metodologias em uma ICI de tecnologia Cabo ou *Wireless* da Operadora B1, B2 ou B3 (ICI B), que entrega o tráfego na forma SFTPServer, arquivos de captura no formato pcap de tamanho de 20MB apresentando pacotes IP com enlace Ethernet e encapsulamentos PCLI, IP/UDP ou TZSP.

4.3.1 Metodologia Atual

Para a Metodologia Atual temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ICI encaminhando mandado para a operadora e informando o IP de acesso “IP_OICB”;
2. Operadora, após fazer as devidas configurações nos seus equipamentos, responde à solicitação informando a conta SFTP de acesso ao tráfego da ICI solicitada (servidor SFTP: “IP_SFTP_Op”, porta: “porta_SFTP_Op”, usuário: “conta_ICI” e senha: “senha_ICI”). Os arquivos de captura são disponibilizados na pasta *home* do usuário criado, sendo que o arquivo aberto (arquivo que não alcançou o tamanho limite e continua recebendo pacotes do programa de captura) também fica disponível para ser baixado;
3. Investigador cria as pastas “C:\intercept\OpA\AlvoA\não tratados”, “C:\intercept\OpA\AlvoA\não importados”, “C:\intercept\OpA\AlvoA\já importados” e “C:\intercept\OpA\AlvoA\NFAT”;
4. Configura o programa WinSCP com os dados de acesso a conta SFTP (Figura 4-1) e pasta local para baixar os arquivos;
5. Configura o programa NetResident para armazenar os dados desta ICI na pasta “C:\intercept\OpA\AlvoA\NFAT”.

Os Procedimentos Rotineiros são:

1. Investigador, utilizando o programa WinSCP, se conecta ao servidor da operadora e baixa os arquivos de captura ainda não baixados para a pasta “não tratados” (a verificação é manual: inspeção visual das pastas “não tratados”, “já importados” e “não importados”);
2. Investigador executa *script* “RetiraEncap”que varre a pasta “não tratados” e, para cada arquivo encontrado, edita os pacotes IP para retirar o encapsulamento

(programa Bittwiste) gerando novos arquivos de captura na pasta “não importados” e move o arquivo original para a pasta “já importados”. O processo de edição dos pacotes IP elimina os pacotes encapsuladores fragmentados (apenas o primeiro fragmento é mantido), causando perda de informações para análise. Essa etapa é necessária devido à incompatibilidade da NFAT com os encapsulamentos PCLI, IP/UDP e TZSP;

3. Investigador, utilizando o programa NetResident, importa os arquivos que estão na pasta “não importados”. Os arquivos importados são deletados (os arquivos originais em formato já foram movidos para a pasta “já importados” pelo *script* “RetiraEncap” para preservação);
4. Investigador, utilizando o programa NetResident, visualiza o conteúdo de cada objeto identificado em ordem cronológica. Cada objeto julgado como relevante é relatado em documento do MSWord. Como a ferramenta não possui uma forma de exportação simples, são utilizados mecanismos de copiar e colar textos ou mesmo imagens através de comandos “*printscreen*”. Ao terminar a análise, o Investigador anota a data/hora do último objeto analisado a fim de retomar o trabalho no ponto onde parou quando novos arquivos de captura forem importados, pois a ferramenta não tem mecanismo de marcação de itens já analisado ou pendentes de análise.

Os Procedimentos Finais são:

1. Investigador consolida seu relatório fazendo novas buscas na ferramenta NetResident e em suas anotações, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.3.2 Metodologia SIT/CLIT

Para a Metodologia SIT/CLIT temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ICI encaminhando mandado à operadora e informando o IP de acesso “IP_OICB”;
2. Operadora, após fazer as devidas configurações nos seus equipamentos, responde à solicitação informando a conta SFTP de acesso ao tráfego da ICI solicitada: servidor SFTP: “IP_SFTP_Op”, porta: “porta_SFTP_Op”, usuário: “conta_ICI” e senha: “senha_ICI”. Os arquivos de captura são disponibilizados na pasta *home* do

usuário criado, sendo que o arquivo aberto (arquivo que não alcançou o tamanho limite e continua recebendo pacotes do programa de captura) também fica disponível para ser baixado;

3. Investigador cria a pasta "C:\intercept\OpA\AlvoA\pcaps" para armazenamento dos arquivos de captura e nela o arquivo de *download* automático (".script.txt") com o conteúdo apresentado na Figura 4-2-a;
4. Investigador cria arquivo de *script* (arquivo c:\intercept\importador.bat) para execução do programa Importador a cada 60 segundos, informando a pasta local dos arquivos de captura e o arquivo de banco de dados da ICI, conforme apresentado na Figura 4-2-b, e executa-o.

Os Procedimentos Rotineiros são:

1. Investigador executa o programa Analisador e abre o arquivo de banco de dados da ICI (c:\intercept\OpA\AlvoA\AlvoA.mdb), seleciona apenas os itens com Status igual a "Não Analisados", visualiza cada um dos itens, registra anotações e marca-os como "importante" ou "não importante" para a investigação.

Os Procedimentos Finais são:

1. Investigador seleciona todos os itens marcados como "Importante" no programa Analisador e os exporta em formato de imagens e as arrasta para dentro de um novo documento do MSWord. Neste documento realizam-se os relatos baseado nos comentários já informados na ferramenta. Então consolida o relatório fazendo novas buscas na ferramenta, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.3.3 Comparação

Comparando as metodologias com base nos critérios estabelecidos, temos:

1. Abrangência da análise: "Parcial" para a Metodologia Atual (parte dos pacotes são desprezados no processo de retirada do cabeçalho IP encapsulador) e "Total" para a Metodologia SIT/CLIT (CLIT é compatível com os encapsulamentos PCLI, IP/UDP e TZSP);

2. Confiabilidade de recebimento do tráfego: “Alta” para ambas as metodologias (inerente a entrega de tráfego SFTPServer);
3. Quantidade de processos não automatizados: três para a Metodologia Atual (*download* dos arquivos de captura, remoção de encapsulamento e importação de arquivos de captura) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* e gerenciamento automático dos arquivos “já importados” / “não importados” e é compatível com os encapsulamentos PCLI, IP/UDP e TZSP);
4. Quantidade de controles manuais: três para a Metodologia Atual (arquivos “baixados” / “não baixados”, arquivos “já importados” / “não importados” e de ponto de parada da análise) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* automático apenas de arquivos não baixados, importa de forma automática apenas arquivos de captura ainda não importados e permite que o Investigador marque os objetos já analisados indicando com clareza o ponto onde parou a análise);
5. Quantidade de programas externos: três para a Metodologia Atual (WinSCP, MSWord e o *script* “RetiraEncap”) e um para a Metodologia SIT/CLIT (MSWord);
6. Acompanhamento em tempo real: “Não” para a Metodologia Atual (Investigador deve esperar arquivo de captura “encher”, ou seja, alcançar limite de 20MB, para então baixá-lo e importá-lo) e “Sim” para a Metodologia SIT/CLIT (CLIT pode baixar do servidor SFTP o arquivo aberto pelo programa de captura em qualquer momento e, em uma próxima importação do mesmo arquivo, continuar a importação a partir do último pacote IP analisado anteriormente, sem causar duplicação de objetos extraídos).

Conforme resumo apresentado na Tabela 4.3, para este cenário a Metodologia SIT/CLI apresentou resultados vantajosos em cinco critérios, sendo que manteve o mesmo resultado da Metodologia Atual em um critério em que não era possível evolução.

Tabela 4.3 – ICI B: comparação das metodologias

Critério	**Metodologia Atual**	**Metodologia SIT/CLIT**
Abrangência da análise	Parcial	Total
Confiabilidade de recebimento do tráfego	Alta	Alta
Quantidade de processos não automatizados	3	0
Quantidade de controles manuais	3	0
Quantidade de programas externos	3	1
Acompanhamento em tempo real	Não	Sim

4.4 EXPERIMENTO ICI C

Neste experimento são comparadas as metodologias em uma ICI de tecnologia 3G ou ADSL da operadora C1 ou C2, que entrega o tráfego na forma SFTPClient, arquivos de captura no formato pcap de tamanho de 50KB ou 500KB apresentando pacotes IP com enlace Ethernet e sem encapsulamento IP.

4.4.1 Metodologia Atual

Para a Metodologia Atual temos os seguintes Procedimentos Iniciais:

1. Investigador cria as pastas “C:\intercept\OpA\AlvoA\não tratados”, “C:\intercept\OpA\AlvoA\não importados”, “C:\intercept\OpA\AlvoA\já importados” e “C:\intercept\OpA\AlvoA\NFAT”;
2. Investigador configura o programa NetResident para armazenar os dados desta ICI na pasta “C:\intercept\OpA\AlvoA\NFAT”;
3. Investigador configura o programa SFTPServer para escutar (atender conexões) na porta “portaSFTP_ADSL” do “IP_fixo_ADSL” e cria conta “conta_ICI” com senha “senha_ICI” com pasta *home* “C:\intercept\OpA\AlvoA\não tratados” para receber o tráfego da operadora (Figura 4-3);
4. Investigador solicita ICI encaminhando mandado para a Operadora e informando os dados de acesso a conta “conta_ICI”;

5. Operadora, após fazer as devidas configurações nos seus equipamentos, inicia o envio dos dados.

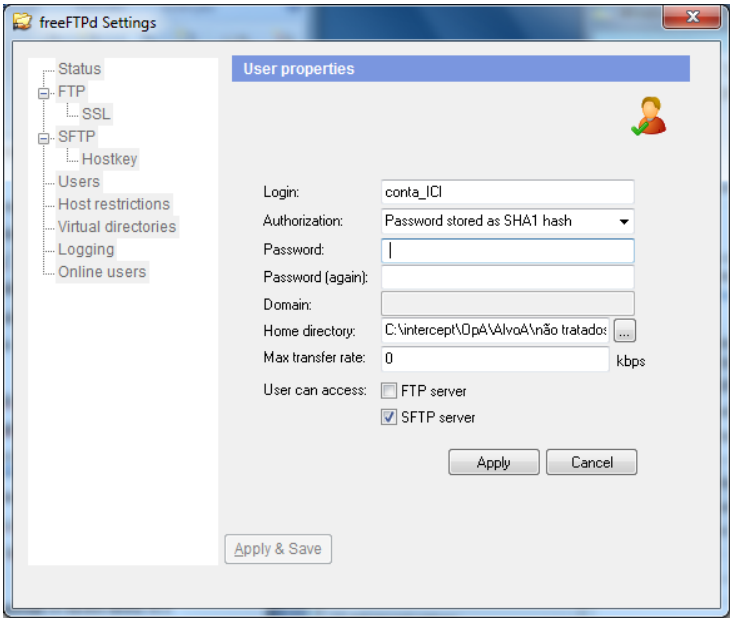


Figura 4-3 – Programa freeFTPd: criação de usuário SFTP

Os Procedimentos Rotineiros são:

1. Investigador executa *script* “MoveCopyPcap”, que varre a pasta “não tratados” e para cada arquivo encontrado move-o para a pasta “não importados” e copia-o para a pasta “já importados”. Esse *script* é opcional. Seu uso é para evitar que o usuário deixe de importar algum arquivo de captura no próximo passo;
2. Investigador, utilizando o programa NetResident, importa os arquivos que estão na pasta “não importados”. Os arquivos importados são então deletados pelo Investigador (os arquivos originais já foram copiados para a pasta “já importados” para preservação no procedimento anterior);
3. Investigador, utilizando o programa NetResident, visualiza o conteúdo de cada objeto identificado em ordem cronológica. Cada objeto julgado como relevante é relatado em documento do MSWord. Como a ferramenta não possui uma forma de exportação simples, são utilizados mecanismos de copiar e colar textos ou mesmo imagens através de comandos “*printscreen*”. Ao terminar a análise, o Investigador anota a data/hora do último objeto analisado a fim de retomar o trabalho no ponto onde parou quando novos arquivos de captura forem importados, pois a ferramenta não tem mecanismo de marcação de itens já analisado ou pendentes de análise.

Os Procedimentos Finais são:

1. Investigador consolida seu relatório fazendo novas buscas na ferramenta NetResident e em suas anotações, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.4.2 Metodologia SIT/CLIT

Para a Metodologia SIT/CLIT temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ao Gestor do SIT criação de conta SFTP para interceptação;
2. Gestor do SIT cria conta SFTP "conta_ICI" com senha gerada "senha_ICI", sendo que o IP "IP_SFTP_SIT" e porta "porta_SFTP_SIT" são únicos para todas as interceptações. Dados são informados ao Investigador;
3. Investigador solicita ICI encaminhando mandado para a operadora e informando os dados de acesso a conta "conta_ICI";
4. Operadora, após fazer as devidas configurações nos seus equipamentos, inicia o envio dos dados;
5. Investigador cria a pasta "C:\intercept\OpA\AlvoA\pcaps" para armazenamento dos arquivos de captura e nela o arquivo de *download* automático (".script.txt") com o conteúdo apresentado na Figura 4-4;
6. Investigador cria arquivo de *script* (arquivo c:\intercept\importador.bat) para execução do programa Importador a cada 60 segundos, informando a pasta local dos arquivos de captura e o arquivo de banco de dados da ICI, conforme apresentado na Figura 4-2-b, e executa-o.

```
open sftp://conta_ICI:senha_ICI@IP_SFTP_SIT:porta_SFTP_SIT
lcd C:\intercept\OpA\AlvoA\pcaps
get *
bye
```

Figura 4-4 – *Script* de *download* automático de arquivos de captura do SIT

Os Procedimentos Rotineiros são:

1. Investigador executa o programa Analisador e abre o arquivo de banco de dados da ICI (c:\intercept\OpA\AlvoA\AlvoA.mdb), seleciona apenas os itens com Status igual a “Não Analisados”, visualiza cada um dos itens, registra anotações e marca-os como “importante” ou “não importante” para a investigação.

Os Procedimentos Finais são:

1. Investigador seleciona todos os itens marcados como “Importante” no programa Analisador e os exporta em formato de imagens e as arrasta para dentro de um novo documento do MSWord. Neste documento realizam-se os relatos baseado nos comentários já informados na ferramenta. Então consolida o relatório fazendo novas buscas na ferramenta, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.4.3 Comparação

Comparando as metodologias com base nos critérios estabelecidos, temos:

1. Abrangência da análise: ambas as metodologias têm abrangência “Total” da análise (existe total compatibilidade das ferramentas com os formatos de arquivos e pacotes);
2. Confiabilidade de recebimento do tráfego: “Média” para a Metodologia Atual (entrega SFTPClient com infraestrutura não confiável) “Alta” a Metodologia SIT/CLIT (entrega SFTPClient com infraestrutura confiável do SIT);
3. Quantidade de processos não automatizados: um para a Metodologia Atual (importação de arquivos de captura) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* e gerenciamento automático dos arquivos “já importados” / “não importados”);
4. Quantidade de controles manuais: dois para a Metodologia Atual (arquivos “já importados” / “não importados” e de ponto de parada da análise) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* automático apenas de arquivos não baixados, importa de forma automática apenas arquivos de captura ainda não importados e permite que o Investigador marque os objetos já analisados indicando com clareza o ponto onde parou a análise);

5. Quantidade de programas externos: dois para a Metodologia Atual (freeFTPd e MSWord, sendo que o *script* “MoveCopyPcap” não é contado por ser opcional) e um para a Metodologia SIT/CLIT (MSWord);
6. Acompanhamento em tempo real: “Sim” para ambas as metodologias (operadoras trabalham com pequenos arquivos de captura que, mesmo não atingindo o tamanho limite após um pequeno intervalo de tempo, são fechados e enviados para o OICB).

Conforme resumo apresentado na Tabela 4.4, para este cenário a Metodologia SIT/CLI apresentou resultados vantajosos em quatro critérios, sendo que manteve os mesmos resultados da Metodologia Atual nos dois critérios em que não eram possíveis evoluções.

Tabela 4.4 – ICI C: comparação das metodologias

Critério	**Metodologia Atual**	**Metodologia SIT/CLIT**
Abrangência da análise	Total	Total
Confiabilidade de recebimento do tráfego	Média	Alta
Quantidade de processos não automatizados	1	0
Quantidade de controles manuais	2	0
Quantidade de programas externos	2	1
Acompanhamento em tempo real	Sim	Sim

4.5 EXPERIMENTO ICI D

Neste experimento são comparadas as metodologias em uma ICI de tecnologia 3G da operadora D, que disponibiliza o tráfego na forma SFTPClient, arquivos de captura no formato Snoop de tamanho de 50KB apresentando pacotes IP com enlace Ethernet e sem encapsulamento IP.

4.5.1 Metodologia Atual

Para a Metodologia Atual temos os seguintes Procedimentos Iniciais:

1. Investigador cria as pastas "C:\intercept\OpA\AlvoA\não tratados", "C:\intercept\OpA\AlvoA\não importados", "C:\intercept\OpA\AlvoA\já importados" e "C:\intercept\OpA\AlvoA\NFAT";
2. Investigador configura o programa NetResident para armazenar os dados desta ICI na pasta "C:\intercept\OpA\AlvoA\NFAT";
3. Investigador configura o programa SFTPServer para escutar (atender conexões) na porta "portaSFTP_ADSL" do "IP_fixo_ADSL" e cria conta "conta_ICI" com senha "senha_ICI" com pasta *home* "C:\intercept\OpA\AlvoA\não tratados" para receber o tráfego da operadora (Figura 4-3);
4. Investigador solicita ICI encaminhando mandado à operadora e informando os dados de acesso a conta "conta_ICI";
5. Operadora, após fazer as devidas configurações nos seus equipamentos, inicia o envio dos dados.

Os Procedimentos Rotineiros são:

1. Investigador executa *script* "ConvSnoopToPcap", que varre a pasta "não tratados" e para cada arquivo encontrado converte-o para o formato pcap na pasta "não importados" com o programa Editcap, então move o arquivo original (formato Snoop) para a pasta "já importados";
2. Investigador, utilizando o programa NetResident, importa os arquivos que estão na pasta "não importados". Os arquivos importados são então deletados pelo Investigador (os arquivos originais em formato Snoop já foram movidos para a pasta "já importados" para preservação no procedimento anterior);
3. Investigador, utilizando o programa NetResident, visualiza o conteúdo de cada objeto identificado em ordem cronológica. Cada objeto julgado como relevante é relatado em documento do MSWord. Como a ferramenta não possui uma forma de exportação simples, são utilizados mecanismos de copiar e colar textos ou mesmo imagens através de comandos "*printscreen*". Ao terminar a análise, o Investigador anota a data/hora do último objeto analisado a fim de retomar o trabalho no ponto onde parou quando novos arquivos de captura forem importados, pois a ferramenta não tem mecanismo de marcação de itens já analisado ou pendentes de análise.

Os Procedimentos Finais são:

1. Investigador consolida seu relatório fazendo novas buscas na ferramenta NetResident e em suas anotações, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.5.2 Metodologia SIT/CLIT

Para a Metodologia SIT/CLIT temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ao Gestor do SIT criação de conta SFTP para interceptação;
2. Gestor do SIT cria conta SFTP “conta_ICI” com senha gerada “senha_ICI”, sendo que o IP “IP_SFTP_SIT” e porta “porta_SFTP_SIT” são únicos para todas as interceptações. Dados são informados ao Investigador;
3. Investigador solicita ICI encaminhando mandado para a operadora e informando os dados de acesso a conta “conta_ICI”;
4. Operadora, após fazer as devidas configurações nos seus equipamentos, inicia o envio dos dados;
5. Investigador cria a pasta “C:\intercept\OpA\AlvoA\pcaps” para armazenamento dos arquivos de captura e nela o arquivo de *download* automático (“.script.txt”) com o conteúdo apresentado na Figura 4-4;
6. Investigador cria arquivo de *script* (arquivo c:\intercept\importador.bat) para execução do programa Importador a cada 60 segundos, informando a pasta local dos arquivos de captura e o arquivo de banco de dados da ICI, conforme apresentado na Figura 4-2-b, e executa-o.

Os Procedimentos Rotineiros são:

1. Investigador executa o programa Analisador e abre o arquivo de banco de dados da ICI (c:\intercept\OpA\AlvoA\AlvoA.mdb), seleciona apenas os itens com Status igual a “Não Analisados”, visualiza cada um dos itens, registra anotações e marca-os como “importante” ou “não importante” para a investigação.

Os Procedimentos Finais são:

1. Investigador seleciona todos os itens marcados como “Importante” no programa Analisador e os exporta em formato de imagens e as arrasta para dentro de um novo documento do MSWord. Neste documento realizam-se os relatos baseado nos comentários já informados na ferramenta. Então consolida o relatório fazendo novas buscas na ferramenta, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.5.3 Comparação

Comparando as metodologias com base nos critérios estabelecidos, temos:

1. Abrangência da análise: ambas as metodologias têm abrangência “Total” da análise (embora a NFAT NetResident não seja compatível com o formato Snoop, o programa EditCap converte-os para formato pcap sem perda de informações através do script “ConvSnoopToPcap” e o CLIT é compatível com o formato Snoop);
2. Confiabilidade de recebimento do tráfego: “Média” para a Metodologia Atual (entrega SFTPClient com infraestrutura não confiável) “Alta” para a Metodologia SIT/CLIT (entrega SFTPClient com infraestrutura confiável do SIT);
3. Quantidade de processos não automatizados: dois para a Metodologia Atual (conversão de formato de arquivo e importação de arquivos de captura) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* e gerenciamento automático dos arquivos “já importados” / “não importados” e é compatível com formato Snoop);
4. Quantidade de controles manuais: dois para a Metodologia Atual (arquivos “já importados” / “não importados” e de ponto de parada da análise) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* automático apenas de arquivos não baixados, importa de forma automática apenas arquivos de captura ainda não importados e permite que o Investigador marque os objetos já analisados indicando com clareza o ponto onde parou a análise);
5. Quantidade de programas externos: três para a Metodologia Atual (freeFTPd, MSWord e o *script* “ConvSnoopToPcap”) e um para a Metodologia SIT/CLIT (MSWord);

6. Acompanhamento em tempo real: “Sim” para ambas as metodologias (operadora trabalha com pequenos arquivos de captura que, mesmo não atingindo o tamanho limite após um pequeno intervalo de tempo, são fechados e enviados para o OICB).

Conforme resumo apresentado na Tabela 4.5, para este cenário a Metodologia SIT/CLI apresentou resultados vantajosos em quatro critérios, sendo que manteve os mesmos resultados da Metodologia Atual nos dois critérios em que não eram possíveis evoluções.

Tabela 4.5 – ICI D: comparação das metodologias

Critério	**Metodologia Atual**	**Metodologia SIT/CLIT**
Abrangência da análise	Total	Total
Confiabilidade de recebimento do tráfego	Média	Alta
Quantidade de processos não automatizados	2	0
Quantidade de controles manuais	2	0
Quantidade de programas externos	3	1
Acompanhamento em tempo real	Sim	Sim

4.6 EXPERIMENTO ICI E

Neste experimento são comparadas as metodologias em uma ICI de tecnologia 3G da operadora E, que disponibiliza o tráfego na forma SFTPClient, arquivos de captura no formato ESTI de tamanho de 50KB apresentando pacotes IP sem enlace e sem encapsulamento IP.

4.6.1 Metodologia Atual

Devido à incompatibilidade das NFATs disponíveis com o formato de captura ETSI e a ausência de ferramentas conhecidas pra conversão deste formato para outro compatível, não são realizadas ICIs em investigados clientes desta operadora pelos OICBs.

4.6.2 Metodologia SIT/CLIT

Para a Metodologia SIT/CLIT temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ao Gestor do SIT criação de conta SFTP para interceptação;
2. Gestor do SIT cria conta SFTP “conta_ICI” com senha gerada “senha_ICI”, sendo que o IP “IP_SFTP_SIT” e porta “porta_SFTP_SIT” são únicos para todas as interceptações. Dados são informados ao Investigador;
3. Investigador solicita ICI encaminhando mandado para a operadora e informando os dados de acesso a conta “conta_ICI”;
4. Operadora, após fazer as devidas configurações nos seus equipamentos, inicia o envio dos dados;
5. Investigador cria a pasta “C:\intercept\OpA\AlvoA\pcaps” para armazenamento dos arquivos de captura e nela o arquivo de *download* automático (“.script.txt”) com o conteúdo apresentado na Figura 4-4;
6. Investigador cria arquivo de *script* (arquivo c:\intercept\importador.bat) para execução do programa Importador a cada 60 segundos, informando a pasta local dos arquivos de captura e o arquivo de banco de dados da ICI, conforme apresentado na Figura 4-2-b, e executa-o.

Os Procedimentos Rotineiros são:

1. Investigador executa o programa Analisador e abre o arquivo de banco de dados da ICI (c:\intercept\OpA\AlvoA\AlvoA.mdb), seleciona apenas os itens com Status igual a “Não Analisados”, visualiza cada um dos itens, registra anotações e marca-os como “importante” ou “não importante” para a investigação.

Os Procedimentos Finais são:

1. Investigador seleciona todos os itens marcados como “Importante” no programa Analisador e os exporta em formato de imagens e as arrasta para dentro de um novo documento do MSWord. Neste documento realizam-se os relatos baseado nos comentários já informados na ferramenta. Então consolida o relatório fazendo novas buscas na ferramenta, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.6.3 Comparação

Comparando as metodologias com base nos critérios estabelecidos, temos:

1. Abrangência da análise: “Nenhuma” para a Metodologia Atual (as NFATs disponíveis não são compatíveis com o formato de captura ETSI) e “Total” para a Metodologia SIT/CLIT (CLIT é compatível com o formato ETSI);
2. Confiabilidade de recebimento do tráfego: “Média” para a Metodologia Atual (entrega SFTPClient com infraestrutura não confiável) “Alta” para a Metodologia SIT/CLIT (entrega SFTPClient com infraestrutura confiável do SIT);
3. Quantidade de processos não automatizados: Não se aplica para a Metodologia Atual e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* e gerenciamento automático dos arquivos “já importados” / “não importados” e é compatível com formato ETSI);
4. Quantidade de controles manuais: Não se aplica para a Metodologia Atual e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* automático apenas de arquivos não baixados, importa de forma automática apenas arquivos de captura ainda não importados e permite que o Investigador marque os objetos já analisados indicando com clareza o ponto onde parou a análise);
5. Quantidade de programas externos Não se aplica para a Metodologia Atual e um para a Metodologia SIT/CLIT (MSWord);
6. Acompanhamento em tempo real: Não se aplica para a Metodologia Atual. “Sim” para a Metodologia SIT/CLIT (operadora trabalha com pequenos arquivos de captura que, mesmo não atingindo o tamanho limite após um pequeno intervalo de tempo, são fechados e enviados para o OICB).

Conforme resumo apresentado na Tabela 4.5, para este cenário a Metodologia SIT/CLI além de apresentar resultados vantajosos nos dois critérios em que foi possível fazer a comparação, permite que se realize ICIs em operadoras que entregam arquivos de captura em formato ETSI, anteriormente não realizadas.

Tabela 4.6 – ICI E: comparação das metodologias

Critério	**Metodologia Atual**	**Metodologia SIT/CLIT**
Abrangência da análise	Nenhuma	Total
Confiabilidade de recebimento do tráfego	Média	Alta
Quantidade de processos não automatizados	NA	0
Quantidade de controles manuais	NA	0
Quantidade de programas externos	NA	1
Acompanhamento em tempo real	NA	Sim

4.7 EXPERIMENTO ICI F

Neste experimento são comparadas as metodologias em uma ICI de tecnologia ADSL da operadora F1 ou F2, que disponibiliza o tráfego na forma encapsulamento de tráfego, e pacotes IP com encapsulamento IP/GRE ou Juniper.

4.7.1 Metodologia Atual

Para a Metodologia Atual temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ICI encaminhando mandado à operadora e informando o endereço IP “IP_Fixo_ADSL” para envio dos pacotes;
2. Operadora, após fazer as devidas configurações nos seus equipamentos, inicia o envio dos dados e informa ao Investigador o endereço IP “IP_Operadora” do equipamento gerador dos pacotes “espelhados”;
3. Investigador cria as pastas “C:\intercept\OpA\AlvoA\não tratados”, “C:\intercept\OpA\AlvoA\não importados”, “C:\intercept\OpA\AlvoA\já importados” e “C:\intercept\OpA\AlvoA\NFAT”;
4. Investigador configura o programa NetResident para armazenar os dados desta ICI na pasta “C:\intercept\OpA\AlvoA\NFAT”;
5. Investigador configura o programa Windump para capturar os pacotes IP com endereço de origem o “IP_Operadora”, arquivos de captura de tamanho 20MB na pasta “C:\intercept\OpA\AlvoA\não tratados” (ex: ‘*windump.exe -i 2 -C 20 -s 0 -w*

"C:\intercept\OpA\AlvoA\não tratados \%date:~-4,4%-%date:~-7,2%-%date:~-10,2%-%time:~0,2%%time:~3,2%.pcap" host IP_Operadora'). Como o modem ADSL necessita ser configurado em modo *bridge* para que os pacotes IP/GRE cheguem à interface de rede do computador, os pacotes IP são capturados com enlace Ethernet/PPPoE.

Os Procedimentos Rotineiros são:

1. Investigador executa *script* "RetiraEnlaceEEncap"que varre a pasta "não tratados" e, para cada arquivo encontrado (exceto o arquivo aberto pelo Windump), edita os pacotes IP para retirar enlaces adicionais e encapsulamentos (programa Bittwiste) gerando novos arquivos de captura na pasta "não importados" e move o arquivo original para a pasta "já importados". O processo de edição dos pacotes IP elimina os pacotes encapsuladores fragmentados (apenas o primeiro fragmento é mantido). Essa etapa é necessária devido a incompatibilidade da NFAT com o enlace Ethernet/PPPoE e o encapsulamento IP/GRE;
2. Investigador, utilizando o programa NetResident, importa os arquivos que estão na pasta "não importados". Os arquivos importados são deletados (os arquivos originais em formato já foram movidos para a pasta "já importados" pelo *script* "RetiraEnlaceEEncap" para preservação);
3. Investigador, utilizando o programa NetResident, visualiza o conteúdo de cada objeto identificado em ordem cronológica. Cada objeto julgado como relevante é relatado em documento do MSWord. Como a ferramenta não possui uma forma de exportação simples, são utilizados mecanismos de copiar e colar textos ou mesmo imagens através de comandos "*printscreen*". Ao terminar a análise, o Investigador anota a data/hora do último objeto analisado a fim de retomar o trabalho no ponto onde parou quando novos arquivos de captura forem importados, pois a ferramenta não tem mecanismo de marcação de itens já analisado ou pendentes de análise.

Os Procedimentos Finais são:

1. Investigador consolida seu relatório fazendo novas buscas na ferramenta NetResident e em suas anotações, opinando pela renovação ou não da ICI;
2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.7.2 Metodologia SIT/CLIT

Para a Metodologia SIT/CLIT temos os seguintes Procedimentos Iniciais:

1. Investigador solicita ao Gestor do SIT criação de conta SFTP para interceptação indicando que se trata de ICI com entrega de tráfego na forma encapsulamento de tráfego;
2. Gestor do SIT cria conta SFTP “conta_ICI” com senha gerada “senha_ICI”, sendo que o IP “IP_SFTP_SIT” e porta “porta_SFTP_SIT” são únicos para todas as interceptações. Separa o endereço disponível “IP_Fixo_ICI” da faixa de IPs para alocada para este tipo de interceptação. Configura o programa Tcpdump para capturar os pacotes IP com endereço de destino o “IP_Fixo_ICI”, arquivos de captura de tamanho 20MB na pasta “/home/conta_ICI”. Dados da conta SFTP e o “IP_Fixo_ICI” são informados ao Investigador;
3. Investigador solicita ICI encaminhando mandado à operadora e informando o endereço IP “IP_Fixo_ICI” para envio dos pacotes;
4. Operadora, após fazer as devidas configurações nos seus equipamentos, inicia o envio dos dados;
5. Investigador cria a pasta “C:\intercept\OpA\AlvoA\pcaps” para armazenamento dos arquivos de captura e nela o arquivo de *download* automático (“.script.txt”) com o conteúdo apresentado na Figura 4-4;
6. Investigador cria arquivo de *script* (arquivo c:\intercept\importador.bat) para execução do programa Importador a cada 60 segundos, informando a pasta local dos arquivos de captura e o arquivo de banco de dados da ICI, conforme apresentado na Figura 4-2-b, e executa-o.

Os Procedimentos Rotineiros são:

1. Investigador executa o programa Analisador e abre o arquivo de banco de dados da ICI (c:\intercept\OpA\AlvoA\AlvoA.mdb), seleciona apenas os itens com Status igual a “Não Analisados”, visualiza cada um dos itens, registra anotações e marca-os como “importante” ou “não importante” para a investigação.

Os Procedimentos Finais são:

1. Investigador seleciona todos os itens marcados como “Importante” no programa Analisador e os exporta em formato de imagens e as arrasta para dentro de um

novo documento do MSWord. Neste documento realizam-se os relatos baseado nos comentários já informados na ferramenta. Então consolida o relatório fazendo novas buscas na ferramenta, opinando pela renovação ou não da ICI;

2. Investigador gera mídia não regravável com os arquivos de captura baixados da operadora, calcula os *hashs* dos arquivos e os faz constar no relatório.

4.7.3 Comparação

Comparando as metodologias com base nos critérios estabelecidos, temos:

1. Abrangência da análise: “Parcial” para a Metodologia Atual (parte dos pacotes são desprezados no processo de retirada do cabeçalho IP encapsulador) e “Total” para a Metodologia SIT/CLIT (CLIT é compatível com os encapsulamentos IP/GRE e Juniper);
2. Confiabilidade de recebimento do tráfego: “Baixa” para a Metodologia Atual (entrega encapsulamento de tráfego com infraestrutura não confiável) “Média” para a Metodologia SIT/CLIT (entrega encapsulamento de tráfego com infraestrutura confiável);
3. Quantidade de processos não automatizados: dois para a Metodologia Atual (remoção de encapsulamento e importação de arquivos de captura) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* e gerenciamento automático dos arquivos “já importados” / “não importados” e é compatível com os encapsulamentos IP/GRE e Juniper);
4. Quantidade de controles manuais: dois para a Metodologia Atual (arquivos “já importados” / “não importados” e de ponto de parada da análise) e nenhum para a Metodologia SIT/CLIT (CLIT faz o *download* automático apenas de arquivos não baixados, importa de forma automática apenas arquivos de captura ainda não importados e permite que o Investigador marque os objetos já analisados indicando com clareza o ponto onde parou a análise);
5. Quantidade de programas externos: três para a Metodologia Atual (Windump, MSWord e o *script* “RetiraEnlaceEEncap”) e um para a Metodologia SIT/CLIT (MSWord);
6. Acompanhamento em tempo real: “Não” para a Metodologia Atual (Investigador deve esperar arquivo de captura “encher”, ou seja, alcançar limite de 20MB, para então baixá-lo e importá-lo) e “Sim” para a Metodologia SIT/CLIT (CLIT pode

baixar do servidor SFTP o arquivo aberto pelo programa de captura em qualquer momento e, em uma próxima importação do mesmo arquivo, continuar a importação a partir do último pacote IP analisado anteriormente, sem causar duplicação de objetos extraídos).

Conforme resumo apresentado na Tabela 4.7, para este cenário a Metodologia SIT/CLI apresentou resultados vantajosos em todos os seis critérios em relação à Metodologia Atual.

Tabela 4.7 – ICI F: comparação das metodologias

Critério	**Metodologia Atual**	**Metodologia SIT/CLIT**
Abrangência da análise	Parcial	Total
Confiabilidade de recebimento do tráfego	Baixa	Média
Quantidade de processos não automatizados	2	0
Quantidade de controles manuais	2	0
Quantidade de programas externos	3	1
Acompanhamento em tempo real	Não	Sim

4.8 ANÁLISE DOS RESULTADOS

Com base nos experimentos realizados e nos resultados obtidos, analisou-se se a solução proposta atingiu os objetivos específicos propostos. Ressalva-se que no experimento ICI E, não é possível a análise do tráfego pela Metodologia Atual por incompatibilidade com as ferramentas disponíveis, tendo resultados comparáveis apenas nos dois primeiros critérios. Os procedimentos executados nos experimentos são apresentados resumidamente na Tabela 4.8.

Tabela 4.8 – Procedimentos realizados nos experimentos

	Metodologia Atual						**Metodologia SIT/CLIT**	
	ICI A	**ICI B**	**ICI C**	**ICI D**	**ICI E**	**ICI F**	**ICI A e B**	**ICI C, D, E e F**
Proc. Iniciais	Env. Mandado Config. Remota Config. Local (WinSCP, NetResident)	Env. Mandado Config. Remota Config. Local (WinSCP, NetResident)	Config. Local (FreeFTPd, NetResident) Envio Mandado Config. Remota	Config. Local (FreeFTPd, NetResident) Envio Mandado Config. Remota	Config. Local (FreeFTPd, NetResident) Envio Mandado Config. Remota	Envio Mandado Config. Remota Config. Local (windump, NetResident)	Envio Mandado Config Remota Config Local (CLIT)	Pedido cta. SIT Config SIT Envio Mandado Config Remota Config Local (CLIT)
Proc. Rotineiros	Obtenção (Cliente SFTP) Importação (NetResident) Análise (NetResident, MSWord)	Obtenção (Cliente SFTP) Tratamento (RetiraEncap) Importação (NetResident) Análise (NetResident, MSWord)	Importação (NetResident) Análise (NetResident, MSWord)	Tratamento (ConvSnoopToPcap) Importação (NetResident) Análise (NetResident, MSWord)	Prejudicado (NFAT incompatível)	Tratamento (RetiraEnlaceEEncap) Importação (NetResident) Análise (NetResident, MSWord)	Análise (CLIT)	Análise (CLIT)
Proc. Finais	Relatório (NetResident, MSWord) Preservação	Relatório (NetResident, MSWord) Preservação	Relatório (NetResident, MSWord) Preservação	Relatório (NetResident, MSWord) Preservação	Relatório (NetResident, MSWord) Preservação	Relatório (NetResident, MSWord) Preservação	Relatório (CLIT, MSWord) Preservação	Relatório (CLIT, MSWord) Preservação

4.8.1 Abrangência de Análise

Conforme apresentado na Figura 4-5, a Metodologia SIT/CLIT atinge seu objetivo de aumentar a abrangência da análise, já que se conseguiu abrangência total de análise em todos os experimentos, enquanto que na Metodologia Atual temos 3 experimentos (ICI B, E e F) em que não se alcançou a abrangência total, sendo que na ICI E não é possível fazer qualquer análise do tráfego.

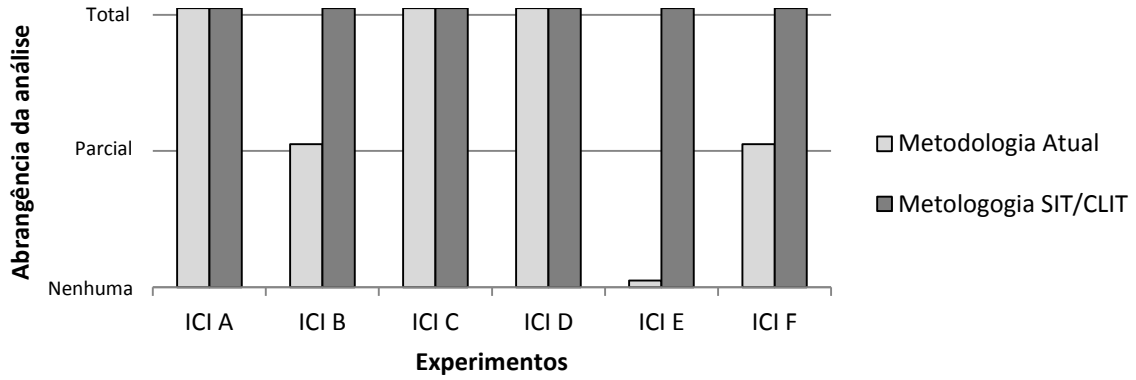


Figura 4-5 – Resultado da abrangência da análise nos experimentos

4.8.2 Confiabilidade de Recebimento do Tráfego

Na Figura 4-6 são apresentados os resultados do critério “Confiabilidade de recebimento do tráfego”, mostrando que a Metodologia SIT/CLIT alcançou seu objetivo de aumentar a

confiabilidade, já que nos experimentos ICI C, D e E aumentou de “Média” para “Alta” e no experimento ICI F de “Baixa” para “Média” em relação à Metodologia Atual.

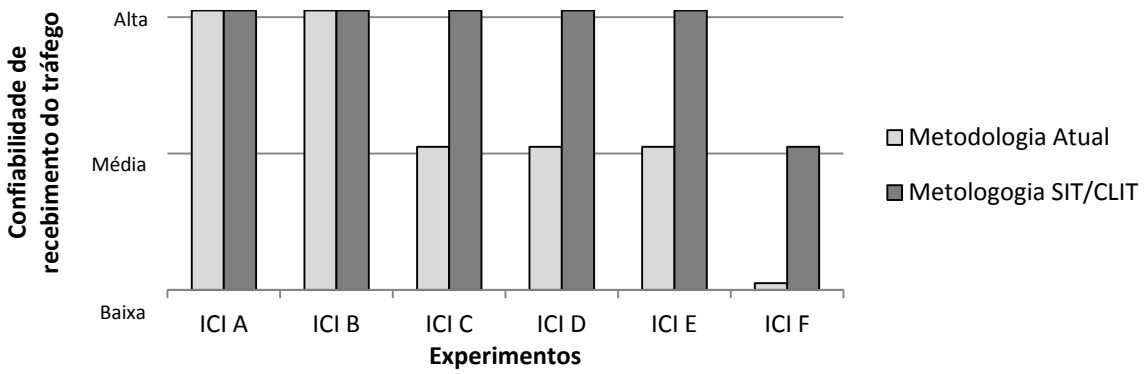


Figura 4-6 – Resultados da confiabilidade de recebimento do tráfego nos experimentos

4.8.3 Quantidade de Processos Não Automatizados

Os resultados do critério “Quantidade de processos não automatizados” são apresentados na Figura 4-7. Enquanto na Metodologia Atual foram necessários de 1 a 3 processos manuais nos experimentos, a Metodologia SIT/CLIT não foram identificados nenhum processo não automatizado (inclusive no experimento ICI E), alcançando seu objetivo de automatizar os processos manuais.

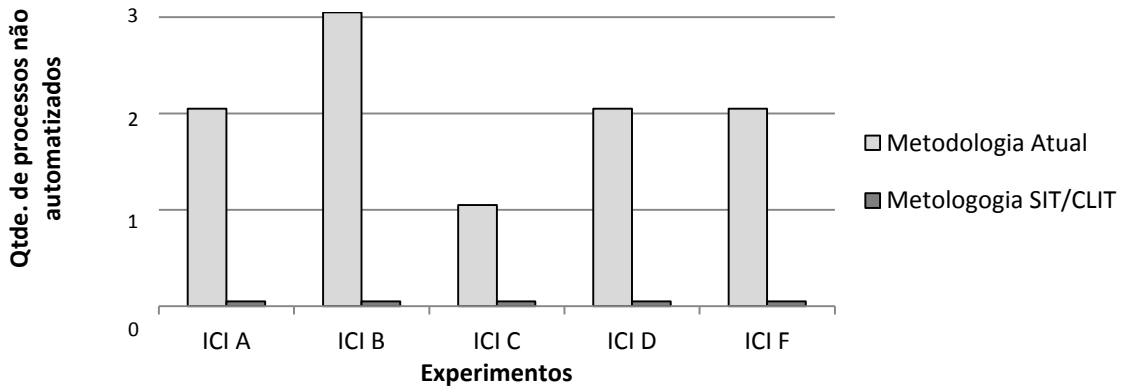


Figura 4-7 – Resultados da quantidade de processos não automatizados nos experimentos

4.8.4 Quantidade de Controles Manuais

No critério “Quantidade de controles manuais”, cujos resultados dos experimentos são apresentados na Figura 4-8, foram necessários de 2 a 3 controles manuais na Metodologia Atual, enquanto na Metodologia SIT/CLIT nenhum controle manual foi necessário (inclusive no experimento E). Portanto a metodologia proposta atingiu seu objetivo de eliminar os controles manuais hoje necessários.

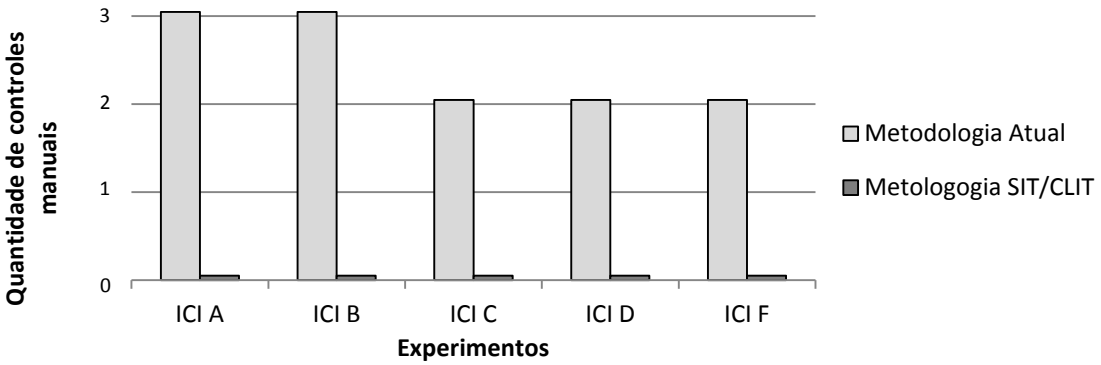


Figura 4-8 – Resultados da quantidade de controles manuais nos experimentos

4.8.5 Quantidade de Programas Externos

Na Figura 4-9 são apresentados os resultados dos experimentos para o critério “Quantidade de programas externos”. Na Metodologia Atual foram necessários a manipulação de 2 a 3 programas que não a NFAT pelo Investigador em cada um dos experimentos realizados, sendo que 7 são diferentes (programas WinSCP, freeFTPd, Windump, MSWord e *scripts* RetiraEncap, ConvSnoopToPcap, RetiraEnlaceEEncap). A Metodologia SIT/CLIT atingiu seu objetivo de reduzir a quantidade de programas externos diminuindo para uma única ferramenta externa (programa MSWord) em todos os experimentos.

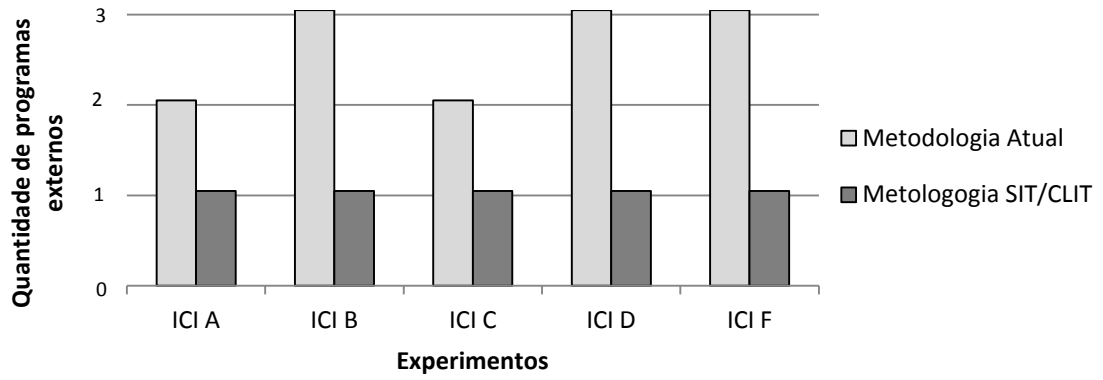


Figura 4-9 – Resultados da quantidade de programas externos nos experimentos

4.8.6 Acompanhamento em Tempo Real

A metodologia SIT/CLIT também atingiu o seu objetivo de permitir o acompanhamento em tempo real do tráfego do investigado, conforme resultados apresentados na Figura 4-10, já que em todos os experimentos (inclusive no experimento ICI E) foi possível esse tipo de acompanhamento. O mesmo não ocorreu com a Metodologia Atual, em que apenas 2 experimentos isso foi possível (ICI C e D).

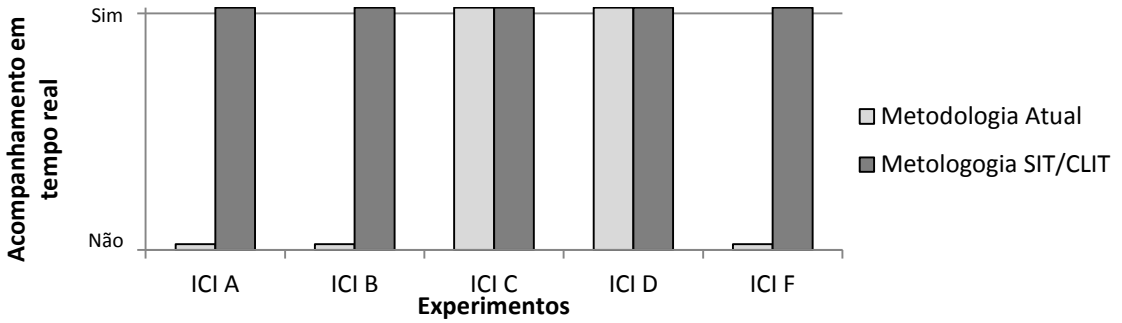


Figura 4-10 – Resultados do acompanhamento em tempo real nos experimentos

4.8.7 Padronização dos Procedimentos

Como pôde ser verificado nos experimentos, na Metodologia Atual ocorreu uma variedade de procedimentos realizados. Nos Procedimentos Iniciais o Investigador precisou configurar diferentes ferramentas de obtenção de tráfego: um programa cliente SFTP (WinSCP) nas ICIs A e B, um programa servidor SFTP (freeFTPd) e nas ICIs C e D e um

programa de captura (Windump) na ICI F. Nos Procedimentos Rotineiros da ICI A e B o Investigador precisou usar ferramenta para buscar os arquivos de captura (WinSCP), enquanto que nas demais os arquivos de captura já ficam disponíveis diretamente em pasta local. Também nos Procedimentos Rotineiros, o Investigador teve que utilizar ferramentas para compatibilização de arquivos ou pacotes a ferramenta NFAT na ICI B (*script* RetiraEncap), ICI D(*script* ConvSnoopToPcap) e ICI F(*script* RetiraEnlaceEEncap), sendo que nas ICI A e C não foi necessário este procedimento. Os Procedimentos Finais não variaram nos experimentos realizados. Na Tabela 4.9 é apresentado um resumo das variações dos procedimentos.

Tabela 4.9 – Variação nos procedimentos da Metodologia Atual nos Experimentos

Experimento	**Proc. Iniciais**	**Procedimentos Rotineiros**	
	Config. de ferramenta p/ obtenção de tráfego	**Obtenção de tráfego**	**Conversão de arquivos / pacotes**
ICI A	WinSCP	WinSCP	-
ICI B	WinSCP	WinSCP	RetiraEncap
ICI C	freeFTPd	-	-
ICI D	freeFTPd	-	ConvSnoopToPcap
ICI F	Windump	-	RetiraEnlaceEEncap

Na Metodologia SIT/CLIT não há quase variação de procedimentos executados pelo Investigador. Nos Procedimentos Iniciais foi configurado o *script* de *download* automático dos arquivos de captura e o *script* de execução automática do programa Importador em todos os experimentos. Nos Procedimentos Rotineiros foi realizada apenas a etapa de Análise da mesma forma em todos os experimentos. Nos Procedimentos Finais foram realizadas as etapas de Relatório e Preservação também da mesma forma em todos os experimentos. A variação observada é nos Procedimentos Iniciais: o Investigador solicitou a criação de conta SFTP para o Gestor do SIT e então solicita a interceptação à operadora nas ICIs C, D, E e F, enquanto nas ICIs A e B o primeiro passo não ocorre. Diante destes

resultados, a Metodologia SIT/CLIT atingiu seu objetivo de padronizar os procedimentos, independente das diversas formas de disponibilização de tráfego pelas operadoras.

4.8.8 Aumento na Quantidade de Vestígios Decodificados

Conforme anteriormente exposto, a medição do critério "Aumento na quantidade de vestígios decodificados" ficou prejudicada devido aos rigores da Lei da Interceptação e normativos internos do OICB quanto à manipulação de dados interceptados, que limita o acesso a apenas Investigadores ligados diretamente a investigação. O acesso ao tráfego de diversas interceptações reais permitiria uma análise comparativa da quantidade de vestígios extraídos entre as metodologias permitindo um parecer conclusivo.

Mesmo com as limitações impostas, as seguintes situações puderam ser observadas:

- Nas ICIs B e F, parte dos pacotes IP é perdida durante o processo de edição para compatibilização com a NFAT, tornando comprometida a decodificação de diversos protocolos na Metodologia Atual, o que não ocorreu na Metodologia SIT/CLIT;
- Na ICI E, não é possível decodificar nenhum protocolo do tráfego devido à incompatibilidade com o formato de arquivo de captura ETSI na Metodologia Atual, sendo que na Metodologia SIT/CLIT o tráfego é decodificado normalmente;
- Nas ICIs C, D e F a confiabilidade de recebimento do tráfego da operadora é menor na Metodologia Atual em comparação com a Metodologia SIT/CLIT, sendo que a perda de tráfego compromete a decodificação dos protocolos;
- Existe um conjunto de protocolos que são decodificados em ambas as metodologias (páginas *web*, *e-mails*, VoIP, *chats* do MSN, Yahoo!Messenger, IG, *webmails* Hotmail e Yahoo), protocolos decodificados somente na Metodologia Atual (*chats* Jabber[43], IRC – *Internet Relay Chat* e NNTP – *Network News Transfer Protocol*) e protocolos decodificados somente na Metodologia SIT/CLIT (*webmails* UOL/BOL, IG, *chats* via *web* do Orkut, Facebook, eBuddy e Hotmail), sendo que o perfil de uso da Internet pelo investigado é que definirá em qual das metodologias haverá maior quantidade de vestígios decodificados (segundo informações de Investigadores são raras as situações em ocorrem os protocolos somente

[43] Disponível em http://www.jabber.org/

decodificados pela Metodologia Atual, enquanto são bastante comuns os protocolos decodificados pela Metodologia SIT/CLIT). Cabe destacar que o CLIT pode ter expandida a quantidade de protocolos decodificados de forma simples através do desenvolvimento de Módulos Filtros internos ao programa ou externos, utilizando qualquer linguagem de programação seguindo as especificações da seção 3.2.1.12.

5 CONCLUSÕES

Nesta dissertação apresenta-se uma solução para simplificação do uso das ICIs nas investigações. Mais do que uma ferramenta de análise de tráfego, tratam-se de dois sistemas independentes, o SIT (Servidor de Interceptação Telemática) e o CLIT (Cliente de Interceptação Telemática), que atuam em todas as fases da interceptação, desde a obtenção até a análise, e uma nova Metodologia da Investigação mais simples. O SIT é uma infraestrutura centralizada para obtenção do tráfego capturado pelas operadoras. O CLIT é uma ferramenta de obtenção, tratamento, importação e análise de tráfego de rede.

Apresentam-se também as questões envolvidas no uso de Interceptações Conexões à Internet (ICIs) nas investigações criminais conduzidas no Brasil. Trata-se de técnica de investigação essencial em uma sociedade incluída digitalmente, mas atualmente muito pouco utilizada se comparada à interceptação telefônica devido, principalmente, às dificuldades envolvidas desde a obtenção do tráfego até a sua análise. O trabalho é realizado pelo Investigador, policial especializado no crime sob investigação e não em TI.

Embora exista legislação que torna válidas as provas obtidas por esse meio e obrigue as operadoras a fornecer o tráfego interceptado de um investigado quando determinado por um juiz competente, não há padronização na forma de disponibilização do tráfego aos OICBs. Assim as operadoras entregam esse tráfego nas mais variadas formas, além de diversos formatos de arquivos e pacotes, buscando o mais simples e menos oneroso para elas considerando sua tecnologia instalada. Aos OICBs cabe se adequar a todas essas variações, utilizando diversas ferramentas nas fases de obtenção, tratamento, importação e análise desse tráfego. Com as ferramentas atualmente em uso, ocorrem perdas de informações em todas essas fases, seja por incompatibilidade das ferramentas utilizadas com os arquivos e pacotes, seja por erros humanos (muitos processos não automatizados e controles manuais a serem realizados pelo Investigador).

Foram realizados experimentos envolvendo simulações de ICIs reais, com diversas combinações de formas de entrega e formatos de arquivos e pacotes, a fim de testar se a solução alcançou os objetivos propostos comparando-a com a metodologia atual de investigação.

Os experimentos demonstraram que as ferramentas (SIT e CLIT) e a metodologia de trabalho propostas, quanto utilizados em investigações com ICIs, aumentaram a abrangência de dados analisados, aumentaram a confiabilidade na recepção do tráfego enviado pelas operadoras, automatizaram processos manuais, eliminaram controles manuais, reduziram a quantidade de programas que precisam ser manipulados pelo Investigador. Além disso, elas permitem o acompanhamento em tempo real do tráfego do investigado e padronizam procedimentos realizados pelo Investigador. Diante destes resultados, pode-se afirmar que se alcançou o objetivo principal do trabalho: a simplificação o uso de ICIs nas investigações em OICBs, sendo esperado, como consequência, o aumento do seu uso a partir da adoção dos sistemas desenvolvidos.

5.1 TRABALHOS FUTUROS

A aplicação da solução proposta nas ICIs nas investigações criminais e os resultados obtidos sugerem diversas oportunidades de trabalhos futuros para ampliar seu uso e suas funcionalidades tanto para o Investigador quanto para um especialista em TI.

A fase de Preservação é apresentada com a última etapa dos Procedimentos Finais a ser executada pelo Investigador. O ideal é que tão logo o arquivo de captura seja recebido, já tenha o seu *hash* calculado para que qualquer manipulação indevida seja detectada, dando mais segurança ao processo e evitando questionamentos futuros. O SIT é a porta de entrada dos arquivos, sendo o local ideal de ser realizado este processo. Sugere-se a implementação de modificações no servidor SFTP e no programa de captura do SIT para que o primeiro gere registro de arquivo de recebido com o seu *hash* e o segundo faça o mesmo quando o arquivo de captura for fechado, ou seja, alcance o tamanho máximo configurado.

O módulo Extrator de Streams do programa Importador do CLIT é compatível com o IP versão 4 apenas. Sugere-se a implementação neste módulo de suporte ao IP versão 6, que tende a ser cada vez mais encontrado no tráfego de Internet capturado. Salienta-se que os módulos Filtros já desenvolvidos não são afetados já que são independentes da camada de rede.

A possibilidade de desenvolvimentos de módulos Filtros Externos do programa Importador do CLIT em qualquer linguagem de programação abre uma grande variedade de projetos a serem desenvolvidos para decodificação de protocolos ainda não implementados. Especial atenção deve ser dada a aplicativos para dispositivos móveis (*smartphones*, *tablets*, etc.), principalmente de *e-mails*, *chats* e VoIPs, que vêm se tornando cada vez mais populares. Nesta linha, em Lange (2011) foi desenvolvido um módulo filtro externo para decodificação do protocolo utilizado pelo programa de compartilhamento de arquivos P2P Emule[44].

O módulo Gerenciador de Streams gera três arquivos no sistema de arquivos para cada fluxo TCP ou UDP identificado em pasta que tem como nome a data de início do fluxo. Quando se decodifica tráfego que possui muitos fluxos em um mesmo dia, é notada perda de desempenho considerável, que parece estar relacionada à criação de grande número de arquivos em uma mesma pasta do sistema de arquivos. Estudos devem ser feitos para tentar reduzir esse efeito indesejável.

O desenvolvimento de módulo com estatísticas do tráfego, tais como portas, protocolos e tráfego decodificado / não decodificado, ajudariam um especialista a identificar aplicações utilizadas pelo investigado que necessitam da implementação de módulo Filtro. Outro dado importante a ser estudado é a quantidade de tráfego perdido durante o processo de captura, a fim de validar a confiabilidade dos dados enviados pela operadora. Essa análise pode ser feita, por exemplo, através da comparação das confirmações TCP capturadas com os dados TCP capturados.

O sistema CLIT desenvolvido permite apenas a busca pelos metadados do objeto decodificado do tráfego. A agregação de um sistema de indexação no conteúdo dos objetos decodificados facilitaria o trabalho de análise para o Investigador. Já a indexação do conteúdo de fluxos não decodificados pode ajudar na identificação de dados importantes para a investigação mesmo em tráfego não decodificado.

[44] Disponível em http://www.emule-project.net

5.2 PUBLICAÇÃO

Com base em informações presentes neste trabalho, foi submetido um artigo intitulado “Ferramentas e Metodologia para Simplificar Investigações Criminais Utilizando Interceptação Telemática” à ICoFCS 2011 (*The Sixth International Conference on Forensic Computer Science*), que foi publicado nos anais dessa conferência e apresentado na VIII Conferência Internacional de Perícias em Crimes Cibernéticos (ICCyber 2011) no período de 05 a 07 de outubro de 2011.

REFERÊNCIAS BIBLIOGRÁFICAS

Agarwal, V.; Gbormittah, F.; Hanley, M. e Rollins, D. (2010). Privacy of Facebook's Native Chat Application, Disponível em <http://theagarwals.net/Vikrant/Privacy_of_Facebook.pdf>. Acesso em 06/09/2011.

Baloo, J. (2003). Lawful Interception of IP Traffic: The European Context. *Black Hat Europe 2003*.

Bellard, F. (2011). FFmpeg Project. Disponível em <http://ffmpeg.org/>. Acesso em 12/09/2011.

Berners-Lee, T.; Fielding, R. e Frystyk, H. (1996). RFC 1945: Hypertext Transfer Protocol -- HTTP/1.0.

Branch, P. (2003). Lawful Interception of the Internet. *Centre for Advanced Internet Architectures. Technical Report 030606A.*

Branch, P.; Pavlicic, A. e Armitage, G. (2004). Using MAC Addresses in the Lawful Interception of IP Traffic. *Australian Telecommunications Networks & Applications Conference 2004 (ATNAC2004), Sydney, Australia*.

Brasil (1941). Decreto-lei nº 3.689 de 3 de outubro de 1941.

Brasil (1962). Lei nº 4.117 de 27 de agosto de 1962.

Brasil (1998). Constituição da República Federativa do Brasil, 1998.

Brasil (1996). Lei nº 9.296 de 24 de julho de 1996.

Brasil (2008). CNJ: Resolução nº 59 de 9 de agosto de 2008.

Bray, T.; Paoli, J.; Sperberg-Mcqueen, C. M.; Maler, E. e Yergeau, F. (2008). Extensible markup language (XML) 1.0 (third edition), W3C recommendation.

Broadway, J.; Turnbull, B. e Slay, J. (2008). Improving the Analysis of Lawfully Intercepted Network Packet Data Captured for Forensic Analysis. *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on*, pp.1361-1368.

CableLabs (2004). PacketCable Electronic Surveillance Specification PKT-SP-ESP-I03-040113.

Callaghan, B. e Gilligan, R. (2005). RFC 1761: Snoop Version 2 Packet Capture File Format.

Carvalho, A. C. A. P.; Sousa, F. B.; Neto, J. F.; Neves, P. H. C. ; Fragoso, R. e Mazzonetto, R. P. (2008) Crimes da Informática no Código Penal Brasileiro.

Proceedings of the Third International Conference of Forensic Computer Science - ICoFCS, pp. 97-103.

Clark, J. (2006). Jonathan Clark's AIM Documentation, Disponível em <http://www.cs.cmu.edu/~jhclark/aim/>. Acesso em 02/09/2011.

CNJ (2011). Portal CNJ – Sobre o CNJ. Disponível em <http://www.cnj.jus.br/sobre-o-cnj>. Acesso em 15/09/2011.

Cohen, M. I. (2008). PyFlag - An advanced network forensic framework, *The Proceedings of the Eighth Annual DFRWS* Conference, pp. S112-S120.

Constantinou, F.; Mavrommatis, P. (2006). Identifying Known and Unknown Peer-to-Peer Traffic. *Network Computing and Applications, 2006. NCA 2006. Fifth IEEE International Symposium on*. pp.93-102.

Corey, V.; Peterman, C.; Shearin, S.; Greenberg, M. S. e Van Bokkelen, J. (2002). Network Forensics Analysis. *Internet Computing, IEEE* , vol.6, no.6, pp. 60- 66.

Costeux, J. L.; Guyard, F. e Bustos, A. M. (2006). QRP08-5: Detection and Comparison of RTP and Skype Traffic and Performance. *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE* , pp.1-5.

Crockford, D. (2006a). JSON: The fat-free alternative to XML. *In Proc. of XML 2006, Boston, USA*.

Crockford, D. (2006b). RFC 4627: The application/json Media Type for JavaScript Object Notation (JSON).

Dainotti, A.; De Donato, W. e Pescapé, A. (2009). TIE: a Community-Oriented Traffic Classification Platform, *TMA'09*, Aachen (Germany).

ECMA International (1999). ECMA-262: ECMAScript Language Specification.

ETSI (2000). ETSI EN 300 961 V8.1.1 (2000-11): Digital cellular telecommunications system (Phase 2+); Full rate speech; Transcoding (GSM 06.10 version 8.1.1 Release 1999).

ETSI (2006). ETSI TS 102 232-3 v2.1.1 - Lawful Interception (LI); Handeover Interface and Servide-Specific Details (SSD) for IP delivery; Part 3: Service-specific details for Internet access services.

ETSI (2008). Disponível em <http://portal.etsi.org/li/Summary.asp>. Acesso em 16/09/2011.

Ficara, D.; Giordano, S.; Oppedisano, F.; Procissi, G. e Vitucci, F. (2008). A cooperative PC/Network-Processor architecture for multi gigabit traffic analysis,

Telecommunication Networking Workshop on QoS in Multiservice IP Networks, 2008. IT-NEWS 2008. 4th International, pp.123-128.

Fielding, R.; Gettys, J.; Mogul, J.; Frystyk, H.; Masinter, L.; Leach, P. e Berners-Lee, T. (1999). RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1.

Fusco, F. e Deri, L. (2010). High Speed Network Traffic Analysis with Commodity Multi-core Systems, *in 10th ACM Internet Measurement Conference (IMC 2010)*. Melbourne, Australia: ACM, pp. 218–224.

Fritsch, T.; Voigt, B. e Schiller, J. (2007). Next Generation In-game Message Interfaces, *Applied Wearable Computing (IFAWC), 2007 4th International Forum on* , pp.1-11.

Han S.H.; Kim M.; Ju H. e Hong J. (2002). The Architecture of NG-MON: A Passive NetworkMonitoring System," *LNCS 2506, DSOM 2002*. pp. 4-27.

Ho, J.; Ping Ji; Weifeng Chen e Hsieh, R. (2009). Identifying Google Talk packets. *Intelligence and Security Informatics, 2009. ISI '09. IEEE International Conference on* , pp.285-290.

Huebner, E., Bem, D., e Bem, O. (2003). Computer Forensics: Past, Present And Future. *Information Security Technical Report*, vol. 8, issue 2, pp. 32-36.

Hyun-Chul, K.; Claffy, K.; Fomenkov, M.; Barman,D.; Faloutsos, M. e Lee, K. (2008). Internet Traffic Classification Demystified: Myths, Caveats, and the BestPractices. *In Proc. ACM CoNEXT*.

Information Sciences Institute (1981a). RFC 791: Internet Protocol – DARPA Internet Program – Protocol Specification.

Information Sciences Institute (1981b). RFC 793: Transmission Control Protocol - DARPA Internet Program - Protocol Specification.

IAB e IESG (2000) RFC 2804: IETF Policy on Wiretapping. IETF Network Working Group.

ISO/IEC (2002). ISO/IEC FDIS 8824: Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation, 2002.

ITU-T (1988). G.711: Pulse code modulation (PCM) of voice frequencies. ITU-T Recommendation.

ITU-T (2005a). 40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM). ITU-T Recommendation G.726 (1990) – Corrigendum 1.

ITU-T (2005b). Low complexity coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss, Annex C: 14 kHz Mode at 24, 32, and 48 kbit/s. ITU-T Recommendation G.722.1 Annex C.

ITU-T (2006). G.723.1: Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s. ITU-T Recommendation.

ITU-T (2007). Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP). ITU-T Recommendation G.729.

ITU-T (2008a). Low-complexity coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss. ITU-T Recommendation G.722.1 (2005) – Corrigendum 1.

ITU-T (2008b). Technical Aspects of Lawful Interception. ITU-T Technology Watch Report #6.

Jeffree, T. (2005). IEEE Standard P802.1Q IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks.

Jennings, R.B.; Nahum, E.M.; Olshefski, D.P.; Saha, D.; Zon-Yin Shae e Waters, C. (2006). A study of Internet instant messaging and chat protocols, *Network, IEEE* , vol.20, no.4, pp.16-21.

Juniper Networks (2010). JUNOS e Internet Software for E-series Routing Platforms - System Basics Configuration Guide. Disponível em <http://www.juniper.net/techpubs/software/erx/junose60/swconfig-system-basics/html/lawful-intercept-config6.html>. Acesso em 06/09/2011.

Khoshbakhtian, M.; Darvishan, A.H. e Eghtedari, P. (2008). Comparative Analysis of IMP services, *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on* , pp.1-6, 7-11.

Lange, R (2011). Identificação de Tráfego do Emule Usando Redes Neurais Artificiais. Dissertação de mestrado em informática, Universidade de Brasília (UnB).

Li, B.; Jin, Z. e Ma, M. (2010). VoIP Traffic Identification Based on Host and Flow Behavior Analysis. *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*. pp.1-4.

Li, T.; Hanks, S.; Meyer D. e Traina, P. (2000). RFC 2784: Generic Routing Encapsulation (GRE), 2000.

Mamakos, L.; Lidl, K.; Evarts, J.; Carrel, D.; Simone, D. e Wheeler, R. (1999). RFC 2516: A Method for Transmitting PPP Over Ethernet (PPPoE).

McKemmish, R (1999). What is forensic computing. Trends and Issues in Crime and Criminal Justice, 118.

Microsoft (2006). Overview of the Microsoft RTAudio Speech codec. Disponível em <http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=7515>. Acesso em 12/09/2011.

Movva, R. e Lai, W. (1999) MSN Messenger Service 1.0 Protocol, Disponível em <http://tools.ietf.org/html/draft-movva-msn-messenger-protocol-00>. Acessado em 02/09/2011.

Myers, J. e Rose, M. (1996). RFC 1939: Post Office Protocol - Version 3.

Palmer, G. (2001). A Road Map for Digital Forensic Research. Technical Report DTR - T001-01 FINAL, DFRWS. Report from the First Digital Forensic Research Workshop (DFRWS).

Pilli, E. S.; Joshi, R. C. e Niyogi, R. (2010) Network Forensic frameworks: Survey and research challenges. *Digital Investigation, Vol. 7*, pp. 14-27.

Ponec, M. ; Giura , P.; Brönnimann, H. e Wein J. (2007). Highly efficient techniques for network forensics, *Proceedings of the 14th ACM conference on Computer and communications security*, Alexandria, Virginia, USA.

Postel, J. (1980). RFC 768: User Datagram Protocol. Internet Engineering Task Force.

Postel, J. (1982). RFC 821: Simple Mail Transfer Protocol.

Postel J. e Reynolds J. (1988). RFC 1042: A Standard for the Transmission of IP Datagrams over IEEE 802 Networks.

Ravnaas, O. A. V. (2008). RTAudio (x-msrta) interop. Disponível em <http://oleandre.wordpress.com/2008/05/31/rtaudio-x-msrta-interop/>. Acesso em 12/09/2011.

Rui Lu; Jia Mi e Bo Huang (2010). Design and implementation of instant messenger security monitoring system based on protocol analysis, *Control and Decision Conference (CCDC), 2010 Chinese* , pp.4290-4293.

Teleco (2011a). Internet Banda Larga no Brasil. Disponível em <http://www.teleco.com.br/blarga.asp>. Acesso em 15/09/2011.

Teleco (2011b). Market Share de acessso via Aparelhos 3G das operadoras de Celular. Disponível em <http://www.teleco.com.br/mshare_wcdma.asp>. Acesso em 15/09/2011.

Tellis, P. S. (2010). Yahoo Messenger Protocol v 9, Disponível em <http://libyahoo2.sourceforge.net/ymsg-9.txt>, Acessado em 02/09/2011.

Schulzrinne, H.; Casner, S.; Frederick R. e Jacobson, V. (2003). RFC 3550: RTP: A Transport Protocol for Real-Time Applications.

Shutko, A. (2005). OSCAR (ICQ v7/v8/v9) protocol documentation, Disponível em <http://iserverd.khstu.ru/oscar/>. Acesso em 02/09/2011.

Sureswaran, R.; Al Bazar, H.; Abouabdalla, O.; Manasrah, A.M. e El-Taj, H. (2009). Active e-mail system SMTP protocol monitoring algorithm. *Broadband Network & Multimedia Technology, 2009. IC-BNMT '09. 2nd IEEE International Conference on* , pp.257-260.

Venky (2011). Yahoo Messenger Protocol (UNOFFICIAL DOCUMENTATION), disponível em <http://www.venkydude.com/articles/yahoo.htm>. Acesso em 02/09/2011.

Wagener, G.; Dulaunoy, A. e Engel, T. (2008). "Towards an Estimation of the Accuracy of TCP Reassembly in Network Forensics," *Future Generation Communication and Networking, 2008. FGCN '08. Second International Conference on* , vol.2, pp.273-278.

WenQi, W. e WeiGuang, L. (2009). The Research on Email Forensic Based Network. *Information Science and Engineering (ICISE), 2009 1st International Conference on*. pp.1912-1915.

Wikimedia Foundation (2011). TZSP - Wikipedia, the free encyclopedia. Disponível em <http://en.wikipedia.org/wiki/TZSP>. Acesso em 06/09/2011.

Wireshark Foundation (2010). SLL - The Wireshark Wiki. Disponível em <http://wiki.wireshark.org/SLL>. Acesso em 06/09/2011.

Wireshark Foundation (2011). Development-LibpcapFileFormat - The Wireshark Wiki. Disponível em <http://wiki.wireshark.org/Development/LibpcapFileFormat>. Acesso em 06/09/2011.

Zhenyu, H.; Zaiqiang L.; Purui S. e Dengguo F. (2005). Blocking MSN: A Case Study of Preventing the Abuse of IM, *Communications, 2005 Asia-Pacific Conference on*, pp.1112-1116.