

# Análise Forense em Pacotes de Dados: Captura e Remontagem

Guilherme de M. M. Taschetto<sup>1</sup>

<sup>1</sup>Faculdade de Informática – Pontifícia Universidade  
Católica do Rio Grande do Sul (PUCRS)  
Porto Alegre – RS – Brazil

**Abstract.** *This article presents an introduction to network capture and analysis forensics tools, enlightening its importance, operation and discursing about its usage implications and results.*

**Resumo.** *Este artigo apresenta uma introdução às ferramentas forenses de captura e análise de rede, salientando a sua importância, a sua forma de funcionamento e discorrendo sobre as implicações do seu uso e de seus resultados.*

## 1. Introdução

A importância da perícia forense digital cresceu muito nos últimos anos. Através de processos, metodologias e ferramentas, a perícia forense digital é capaz de produzir as evidências e provas necessárias para condenar (ou inocentar) criminosos digitais.

Os 4 passos da análise forense de rede são:

- Identificar fontes de evidência aceitas pelo sistema legal;
- Preservar a evidência;
- Analisar a evidência;
- Apresentar o resultado da análise.

Abordaremos ferramentas para a realização dos passos 2 e 3 da análise forense, as chamadas ferramentas de rede para perícia forense.

O segundo passo (preservar a evidência) consiste na coleta de dados de todos os ativos disponíveis na rede como: IDS, IPS, Servidor de Logs, conexões capturadas por sniffers de rede etc. Após a coleta de evidências, é necessário realizar a análise destes pacotes - terceiro passo da análise forense. Para isto, são utilizadas as ferramentas de análise forense de redes, onde os pacotes capturados serão usados como base para remontar os dados reais das camadas mais altas de aplicação.

Tais dados ajudam o perito computacional na obtenção de evidências para solução dos casos, fornecendo provas irrefutáveis que serão apresentadas no laudo pericial.

## 2. Ferramentas de Captura de Pacotes - Preservando a Evidência

O PCAP é uma API (Application Programming Interface) para a captura de tráfego de rede. Em sistemas operacionais baseados em Linux, o PCAP pode ser utilizado através da biblioteca `libpcap`, mantida pelo time de desenvolvedores do TCPDump; já em sistemas Windows, a biblioteca é distribuída por meio da `WinPCap`, mantida pela Riverbed Tech, principal patrocinador do projeto Wireshark.

O PCAP provê uma forma integrada para realizar a captura e a filtragem inteligente de pacotes de rede, suportando uma imensa gama de protocolos, sejam abertos ou proprietários. Além disso, a biblioteca provê a funcionalidade de persistência destes pacotes capturados em arquivos, o que atende o objetivo do segundo passo da análise forense de rede: preservar a evidência.

A API do PCAP é escrita em linguagem C, possuindo wrappers para diversas linguagens e frameworks populares, como C++, Java, Ruby, Haskell, Perl, OCaml, TCL, Python, GoLang, entre outros.

## 2.1. Wireshark

O Wireshark<sup>1</sup> é um analisador de pacotes multi-plataforma, grátis e de código-fonte aberto. Possui uma interface gráfica bastante amigável e de fácil utilização com vários recursos integrados, desenvolvida utilizando o toolkit GTK+ a API PCAP para realizar a captura e filtragem de pacotes. Também existe uma versão baseada em linhas de comando chamada TShark (menos famosa).

O Wireshark permite a operação no modo promíscoo, onde todo o tráfego da rede visível pela interface é capturada, e não somente tráfego direcionado aos endereços locais ou de broadcast/multicast. Entretanto, o modo promíscoo tem sua utilidade ceifada em redes que utilizam switches, uma vez que o tráfego dos pacotes já é direcionado.

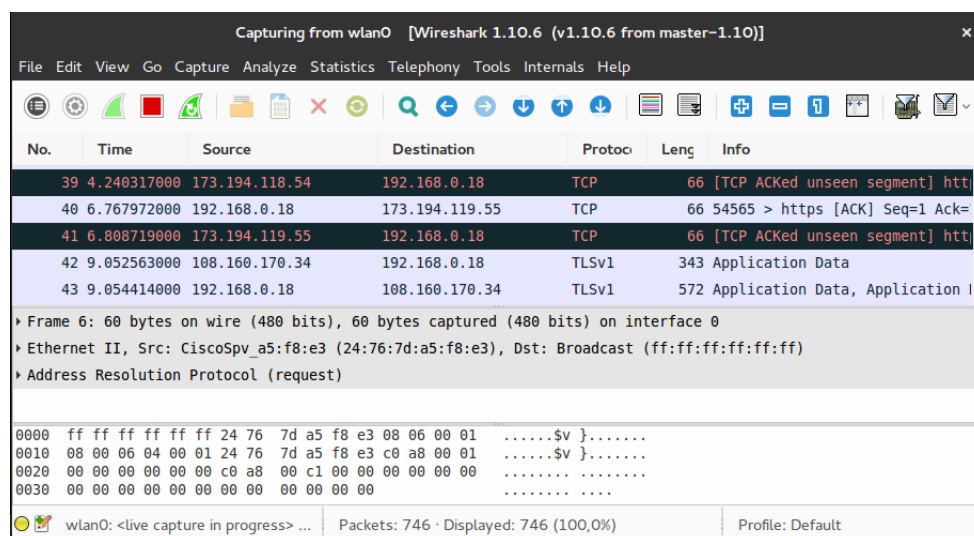


Figura 1. Wireshark, o predador de pacotes.

## 2.2. tcpdump

O tcpdump<sup>2</sup> é um analisador de pacotes operado por linha de comando. No modo padrão de operação, o tcpdump imprime o conteúdo de pacotes no terminal, sejam pacotes capturados ao vivo ou proveniente de arquivos de pacotes armazenados.

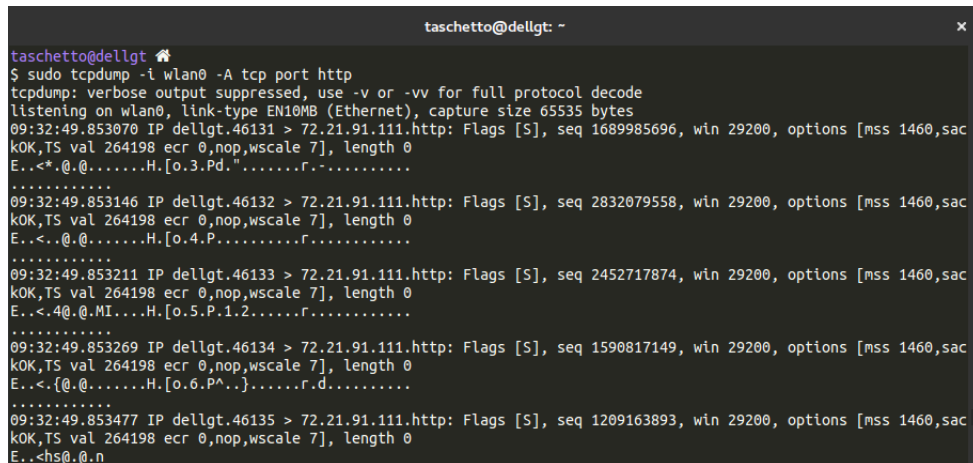
Assim como o Wireshark, o tcpdump também pode operar no modo promíscoo.

<sup>1</sup><https://www.wireshark.org/>

<sup>2</sup><http://www.tcpdump.org/>

**Exemplo de utilização:** `tcpdump -i wlan0 -A tcp port http -w capture.pcap`

**Explicação:** capture todos os pacotes HTTP (TCP) na interface wlan0 e salve-os no arquivo capture.pcap.



```
taschetto@dellgt: ~  
$ sudo tcpdump -i wlan0 -A tcp port http  
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode  
listening on wlan0, link-type EN10MB (Ethernet), capture size 65535 bytes  
09:32:49.853070 IP dellgt.46131 > 72.21.91.111.http: Flags [S], seq 1689985696, win 29200, options [mss 1460,sackOK,TS val 264198 ecr 0,nop,wscale 7], length 0  
E...<*.@.@.....H.[o.3.Pd.".....r.....  
.....  
09:32:49.853146 IP dellgt.46132 > 72.21.91.111.http: Flags [S], seq 2832079558, win 29200, options [mss 1460,sackOK,TS val 264198 ecr 0,nop,wscale 7], length 0  
E...<..@.@.....H.[o.4.P.....r.....  
.....  
09:32:49.853211 IP dellgt.46133 > 72.21.91.111.http: Flags [S], seq 2452717874, win 29200, options [mss 1460,sackOK,TS val 264198 ecr 0,nop,wscale 7], length 0  
E...<.4@.@.MI.....H.[o.5.P.1.2.....r.....  
.....  
09:32:49.853269 IP dellgt.46134 > 72.21.91.111.http: Flags [S], seq 1590817149, win 29200, options [mss 1460,sackOK,TS val 264198 ecr 0,nop,wscale 7], length 0  
E...<{.@.@.....H.[o.6.P^..}.....r.d.....  
.....  
09:32:49.853477 IP dellgt.46135 > 72.21.91.111.http: Flags [S], seq 1209163893, win 29200, options [mss 1460,sackOK,TS val 264198 ecr 0,nop,wscale 7], length 0  
E...<hs@.@.n
```

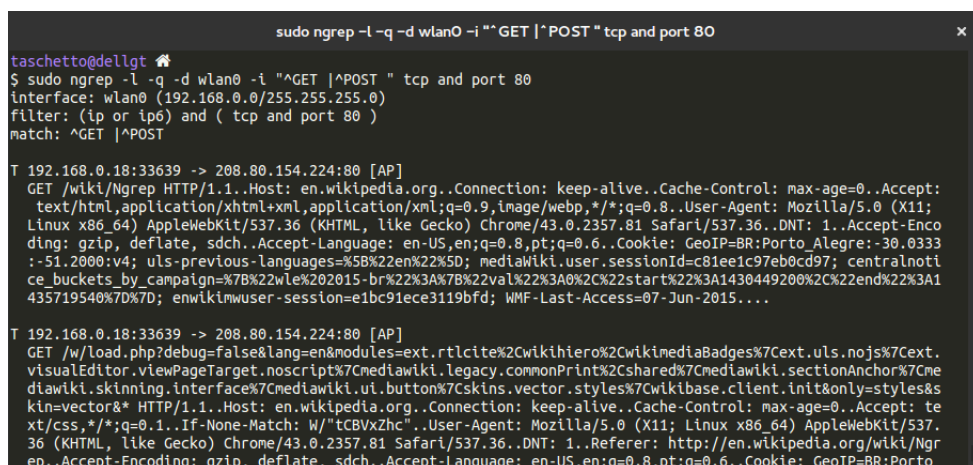
Figura 2. tcpdump furiosamente capturando pacotes HTTP.

## 2.3. ngrep

O ngrep<sup>3</sup> é uma ferramenta bastante parecida com o tcpdump, com o diferencial de possuir a capacidade de filtrar pacotes aplicando expressões regulares nos respectivos payloads.

**Exemplo de utilização:** `ngrep -l -q -d wlan0 -i "^GET | ^POST" tcp and port 80 -O capture.pcap`

**Explicação:** capture todos os pacotes na porta TCP 80 (HTTP) na interface wlan0 que sejam do tipo GET ou POST e salve-os no arquivo capture.pcap.



```
sudo ngrep -l -q -d wlan0 -i ""^GET|^POST" tcp and port 80  
taschetto@dellgt: ~  
$ sudo ngrep -l -q -d wlan0 -i ""^GET|^POST" tcp and port 80  
interface: wlan0 (192.168.0.0/255.255.255.0)  
filter: (ip or ip6) and ( tcp and port 80 )  
match: ^GET|^POST  
  
T 192.168.0.18:33639 -> 208.80.154.224:80 [AP]  
GET /wiki/Ngrep HTTP/1.1..Host: en.wikipedia.org..Connection: keep-alive..Cache-Control: max-age=0..Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8..User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/43.0.2357.81 Safari/537.36..DNT: 1..Accept-Encodin: gzip, deflate, sdch..Accept-Language: en-US,en;q=0.8,pt;q=0.6..Cookie: GeoIP=BR:Porto Alegre:-30.0333 :-51.2000:v4; uls-previous-languages=%5B%22en%22%5D; mediaWiki.user.sessionId=c81ee1c97eb0cd97; centralnoti ce_buckets_by_campaign=%7B%22wle%202015-br%22%3A%7B%22val%22%3A0%2C%22start%22%3A1430449200%2C%22end%22%3A1 435719540%7D%7D; enwikiwuser-session=e1bc91ece3119bfd; WMF-Last-Access=07-Jun-2015....  
  
T 192.168.0.18:33639 -> 208.80.154.224:80 [AP]  
GET /w/load.php?debug=false&lang=en&modules=ext.rtlcite%2Cwikihiero%2CwikimediaBadges%7Cext.uls.nojs%7Cext. visualEditor.viewPageTarget.noscript%7Cmediawiki.legacy.commonPrint%2Cshared%7Cmediawiki.sectionAnchor%7Cme diawiki.skinning.interface%7Cmediawiki.ui.button%7Cskins.vector.styles%7Cwikibase.client.init&only=styles& kln=vector&* HTTP/1.1..Host: en.wikipedia.org..Connection: keep-alive..Cache-Control: max-age=0..Accept: te xt/css,*/*;q=0.1..If-None-Match: W/"tCBVxZhc"..User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537. 36 (KHTML, like Gecko) Chrome/43.0.2357.81 Safari/537.36..DNT: 1..Referer: http://en.wikipedia.org/wiki/Ngr ep..Accept-Encoding: gzip, deflate, sdch..Accept-Language: en-US,en;q=0.8,pt;q=0.6..Cookie: GeoIP=BR:Porto
```

Figura 3. O ngrep é capaz filtrar pacotes de forma sagaz utilizando expressões regulares.

<sup>3</sup><http://ngrep.sourceforge.net/>

### 3. Ferramentas de Análise Forense de Redes - Analisando a Evidência

Ferramentas de Análise Forense de Redes (Network Forensic Analysis Tool, NFAT) normalmente realizam análises à partir de arquivos PCAP. Em arquivos PCAP são armazenados fluxos de comunicação de rede de diversos protocolos. Ao analisar estes arquivos, as ferramentas NFAT conseguem extrair os dados das comunicações. Por exemplo:

- Requisições e páginas HTML (com JS e CSS) sobre o protocolo HTTP;
- E-mails sobre os protocolos POP, SMTP e IMAP;
- Conversas telefônicas sobre o protocolo SIP;
- Entre diversas outros protocolos, como DNS, ARP, SIP, FTP, TFTP etc.

Porém não há mágica: não há como remontar os dados trocados em comunicações seguras (criptografadas), porém é possível identificar diversas informações da comunicação em si - por exemplo, endereços físicos e lógicos das partes envolvidas.

Além disso é importante salientar que as ferramentas NFAT não são ferramentas de análise e captura de protocolos de rede, e sim ferramentas de análise forense de redes. Embora algumas até possuam o recurso de capturar pacotes, este não é um requisito para considerar a ferramenta NFAT, tampouco é o foco de seus recursos.

#### 3.1. Xplico

O Xplico é uma ferramenta NFAT open-source cujo objetivo é extrair dados contidos em arquivos de capturas de pacotes (formato PCAP). O Xplico é distribuído sob a GNU General Public License e alguns scripts sob a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0) License.

As principais características do Xplico são:

- Protocolos suportados: HTTP, SMTP, POP, IMAP, SIP, FB, chat, FTP, MSN, IRC, Telnet etc;<sup>4</sup>
- Identificação de Protocolo Independente de Porta (PIPI) para cada protocolo de aplicação;
- Multithreading;
- Cada dado remontado pelo Xplico é associado à um arquivo XML contendo todo o fluxo de pacotes correspondentes ao dado remontado;
- Remontagem TCP com verificação de ACK para qualquer pacote;
- Lookup de DNS reverso;
- Sem limite de tamanhos de dados de entrada;
- Suporte a IPv4 e IPv6;
- Modularidade. Cada componente do Xplico é modular.

O Xplico pode ser instalado e executado em qualquer distribuição Linux. Além disso, o projeto hospeda uma versão de demonstração na web suportando arquivos PCAP com no máximo 5 MB. A versão de demonstração pode ser acessada em <http://demo.xplico.org/>.

---

<sup>4</sup>A tabela completa de protocolos suportados pode ser encontrada em <http://www.xplico.org/status>.

### 3.2. CapAnalysis

O CapAnalysis é uma ferramenta NFAT web proprietária que também extrai dados à partir de arquivos PCAP. Diferentemente do Xplico, o CapAnalysis é proprietário e, portanto, sua licença de uso deve ser paga.

As principais características do CapAnalysis são:

- Suporte aos protocolos mais utilizados diariamente no mundo inteiro;
- Remontagem TCP;
- Filtro avançado de fluxos PCAP;
- Inspeção profunda e detalhada de pacotes;
- Geolocalização.

Assim como o Xplico, o CapAnalysis pode ser instalado e executado em qualquer distribuição Linux. Também conta com uma versão de demonstração online, não informando se há limite para o tamanho do arquivo PCAP. A versão de demonstração pode ser acessada em <http://pcap.capanalysis.net/>.

### 4. Estudo de Caso: Xplico

Como o foco aqui é a análise forense dos pacotes, foi gerado um arquivo PCAP realizando uma captura de alguns segundos através do Wireshark. Durante a captura foram acessadas páginas web não criptografadas, realizadas sincronias utilizando o Dropbox, entre outras operações de rede menos significativas. À partir deste arquivo gerado foi possível extrair diversas informações no Xplico.

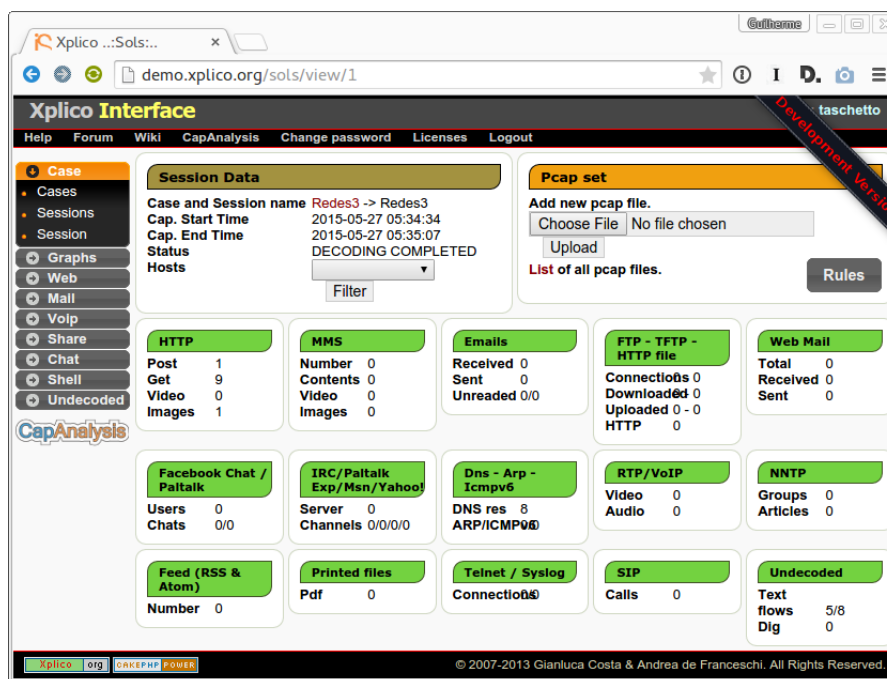


Figura 4. Resumo da Análise

For a complete view of html page set your browser to use Proxy, and point it to Web server.

Web URLs: ☐ HTML ☐ Image ☐ Flash ☐ Video ☐ Audio ☒ JSON ☐ All

Search:  Go

Date	Url	Size	Method	Info
2015-05-27 05:35:04	www.fidedigi.com/promotions/5/buy	102	GET	info.xml
2015-05-27 05:35:04	www.fidedigi.com/promotions/5	1488	GET	info.xml
2015-05-27 05:35:02	www.fidedigi.com/promotions/5	1366	GET	info.xml
2015-05-27 05:35:02	robobash.org/earumsintrecusandae.png?size=300x300	25264	GET	info.xml
2015-05-27 05:34:54	www.fidedigi.com/promotions/1/buy	102	GET	info.xml
2015-05-27 05:34:54	www.fidedigi.com/promotions/1	1468	GET	info.xml
2015-05-27 05:34:52	www.fidedigi.com/promotions/1	1340	GET	info.xml
2015-05-27 05:34:51	www.fidedigi.com/promotions/all	1841	GET	info.xml
2015-05-27 05:34:47	www.fidedigi.com/users/sign_in	90	POST	info.xml
2015-05-27 05:34:47	www.fidedigi.com/	1002	GET	info.xml

Previous 1 of 1 Next

Xplico.org CAKEPHP POWER © 2007-2013 Gianluca Costa & Andrea de Franceschi. All Rights Reserved.

Figura 5. Requisições HTTP

## 5. Conclusão

Após a pesquisa e o estudo de ferramentas NFAT, fica evidente a sua relevância e aplicação na computação forense moderna. Executar ferramentas de captura em modo promíscuo e capturar todo o tráfego de uma rede é uma tarefa trivial. Ser capaz de filtrar e analisar de forma rápida e simples estes dados capturados é de grande importância, seja em investigações públicas como em investigações corporativas. Por exemplo, poder-se-ia identificar usuários acessando conteúdos indevidos na empresa (redes sociais, pornografia, etc) ou até mesmo buscar por consumidores de pedofilia ao analisar o tráfego em pontos estratégicos em WANs.

Entretanto, considero estas ferramentas uma faca de dois gumes: ao mesmo tempo que auxilia na identificação e condenação de infratores e criminosos, também afeta a privacidade dos bons usuários das redes, que não cometem crimes e tem suas informações acessadas muitas vezes sem o seu consentimento.

## Referências

- [Pereira 2015] Pereira, E. D. V. (2015). Notas de aula da disciplina de redes 3. Notas de aula da disciplina de Redes 3.
- [Santos 2013] Santos, F. (2013). Estudo sobre as ferramentas de rede para perícia forense: Estudo de caso do arquivo evidencias.pcap. <http://saomateus.multivix.edu.br/wp-content/uploads/2013/05/Ferramentas-de-rede-para-pericia-forense.pdf>.
- [Xplico 2015] Xplico (2015). <http://www.xplico.org>. Xplico.

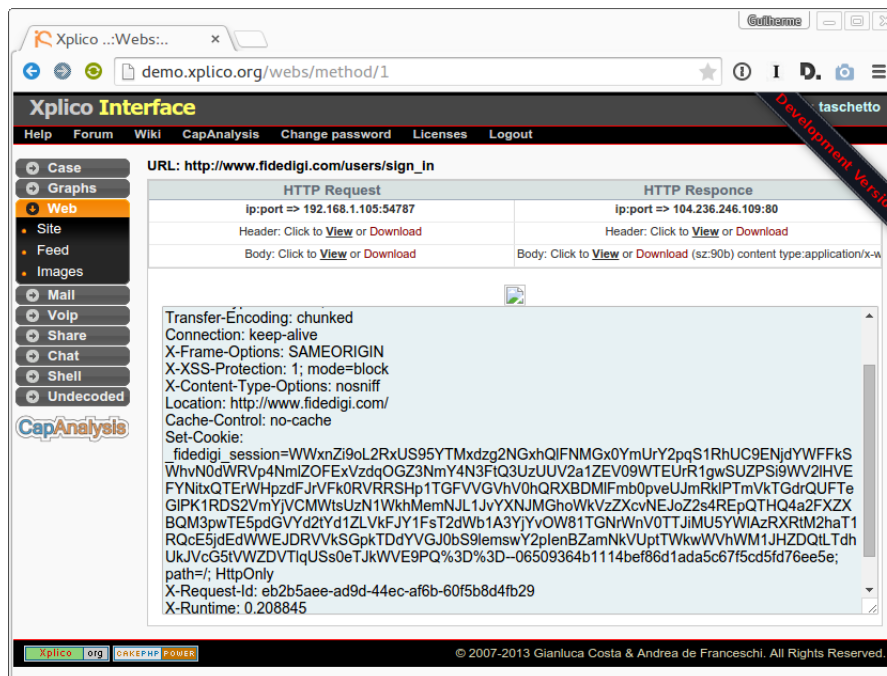


Figura 6. Detalhe do HTTP POST contendo o cookie de sessão

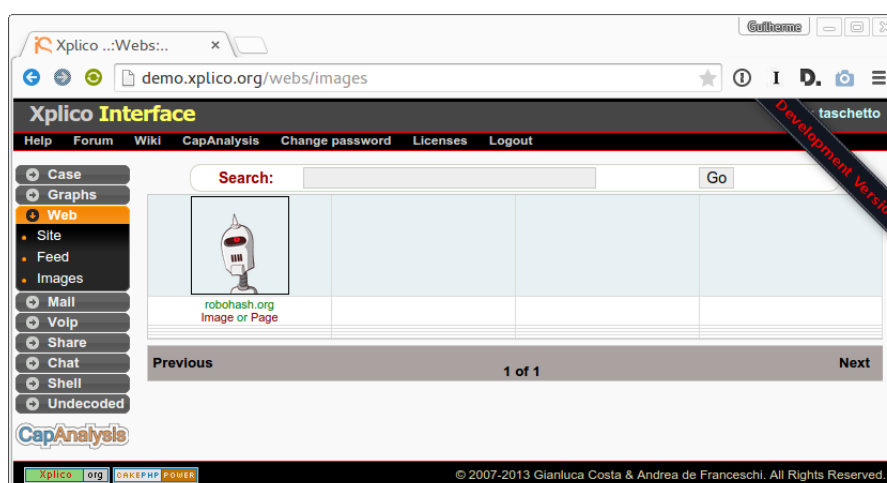


Figura 7. Imagem Recuperada