

Modelo de armazenamento de fluxos de rede para análises de tráfego e de segurança.

VIII Simpósio Brasileiro em Segurança da Informação e de Sistemas
Computacionais
UFRGS – Gramado – RS
03 de setembro de 2008

**Jorge Luiz Corrêa
André Proto
Adriano Mauro Cansian**

UNESP – Universidade Estadual Paulista – Instituto de Biociências, Letras e
Ciências Exatas (IBILCE) - Campus de São José do Rio Preto, SP
www.acmeseecurity.org

Organização da apresentação

- **Fluxo de rede (Netflow)**
- **Motivações**
- **Apresentação do modelo**
 - **Coletor**
 - **Banco de dados de fluxos**
 - **Ambiente**
- **Aferências e detecção de eventos (consultas SQL)**
- **Comparações com outras ferramentas**
- **Conclusão**
 - **Trabalhos em desenvolvimento**

- **Fluxo de rede (Netflow)**
- **Motivações**
- **Apresentação do modelo**
 - Coletor
 - Banco de dados de fluxos
 - Ambiente
- **Aferências e detecção de eventos (consultas SQL)**
- **Comparações com outras ferramentas**
- **Conclusão**
 - Trabalhos em desenvolvimento

- **O que é um fluxo?**
 - Uma seqüência unidirecional de pacotes entre dois *hosts*.
 - Uma tupla na qual as seguintes informações aparecem com o mesmo valor:
 - Endereço IP de origem e destino;
 - Porta de origem e destino (camada de transporte);
 - Campo *Protocol* do datagrama IP;
 - Campo *Type of Service* do datagrama IP;
 - Interface lógica do datagrama no roteador ou switch.

Flow Header Format (Netflow v.5)

0	1	2	3	4	5	6	7
version		count		sys_uptime			
unix_secs				unix_nsecs			
flow_sequence				engine_type	engine_id	sampling_interval	

Flow Record Format (Netflow v.5)

0	1	2	3	4	5	6	7
srcaddr				dstaddr			
nexthop				input		output	
dPkts				dOctets			
first				last			
srcport		dstport		pad1	tcp_flags	prot	tos
src_as		dst_as		src_mask	dst_mask	pad2	

- Fluxo de rede (Netflow)
- **Motivações**
- Apresentação do modelo
 - Coletor
 - Banco de dados de fluxos
 - Ambiente
- Aferências e detecção de eventos (consultas SQL)
- Comparações com outras ferramentas
- Conclusão
 - Trabalhos em desenvolvimento

- **Motivações:**
 - O IPFIX não define como armazenar.
 - Dependência de ferramentas que manipulam fluxos.
 - Facilitar a criação de novos modelos de análise de tráfego utilizando fluxos de dados:
 - Rapidez e versatilidade no acesso às informações.

- Fluxo de rede (Netflow)
- Motivações
- **Apresentação do modelo**
 - Coletor
 - Banco de dados de fluxos
 - Ambiente
- Aferências e detecção de eventos (consultas SQL)
- Comparações com outras ferramentas
- Conclusão
 - Trabalhos em desenvolvimento

- **Coletor**

- Desenvolvido em linguagem Java;
- Identifica os campos do *Netflow* (v5);
- Realiza transformações necessárias nos dados;
- Grava as informações coletadas no Banco de Dados.

- **Banco de Dados de Fluxos**

- Modelo relacional;
- SGBD escolhido: MySQL;
 - Nada impede que outros sejam utilizados;

- **Campos da tabela no BD:**

- Maioria dos campos do *Netflow v5*;
- Baseado na proposta de John-Paul Navarro, Bill Nickless e Linda Winkler;
- **Alguns campos foram otimizados tornando o tamanho de cada tupla menor.**

Proposta inicial (Tamanho da tupla: 57 bytes):


Field	Type
srcaddr	bigint(20) unsigned
dstaddr	bigint(20) unsigned
nexthop	bigint(20) unsigned
input	smallint(5) unsigned
output	smallint(5) unsigned
dPkts	int(10) unsigned
dOctets	int(10) unsigned
first	timestamp
last	timestamp
srcport	smallint(5) unsigned
dstport	smallint(5) unsigned
tcp_flags	tinyint(3) unsigned
prot	tinyint(3) unsigned
tos	tinyint(3) unsigned
src_as	smallint(5) unsigned
dst_as	smallint(5) unsigned
src_mask	tinyint(3) unsigned
dst_mask	tinyint(3) unsigned


Nova proposta (Tamanho da tupla: 45 bytes):

Field	Type
srcaddr	int(10) unsigned
dstaddr	int(10) unsigned
nexthop	int(10) unsigned
input	smallint(5) unsigned
output	smallint(5) unsigned
dPkts	int(10) unsigned
dOctets	int(10) unsigned
first	timestamp
last	timestamp
srcport	smallint(5) unsigned
dstport	smallint(5) unsigned
tcp_flags	tinyint(3) unsigned
prot	tinyint(3) unsigned
tos	tinyint(3) unsigned
src_as	smallint(5) unsigned
dst_as	smallint(5) unsigned
src_mask	tinyint(3) unsigned
dst_mask	tinyint(3) unsigned

- A conversão dos inteiros em IPv4 se dá através das funções **INET_NTOA(expr)** e **INET_ATON(expr)**.

- Exemplo:

mysql> SELECT INET_NTOA(3232240785);  192.168.20.145

mysql> SELECT INET_ATON('192.168.20.145');  3232240785

- Proposta inicial:
7.000.000 x 57 bytes = 399.000.000 bytes ~ 380MB
- Nova proposta:
7.000.000 x 45 bytes = 315.000.000 bytes ~ 300MB

Economia de ~ 21%!

Características da base de dados

- Cada tabela do banco armazena fluxos referentes a 1 dia;
 - Restrição com base na perda de desempenho para tabelas maiores.
- Campo base: marcação do início do fluxo (*first*);
- Indexação por este campo.
- Tabela especial ***last30minutes***:
 - Contém os últimos 30 minutos de fluxos (restrição dos fluxos);
 - A consulta nessa tabela é mais rápida (*heap table*);
 - O coletor é responsável pela sincronia *last30minutes*→disco;
 - Facilita a geração de estatísticas e consulta imediatas.

Outras tabelas têm papéis importantes:

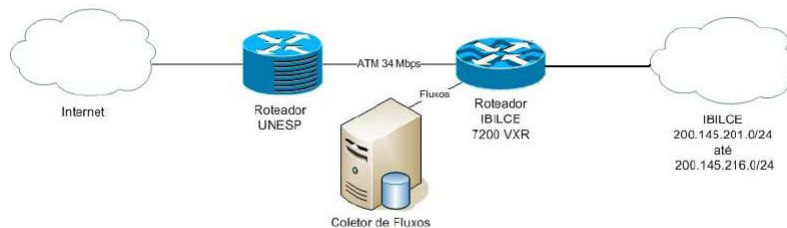
- **Input** (sessões e conexões de fora do ambiente para dentro);
- **Output** (sessões e conexões de dentro do ambiente para fora);
 - São tabelas mantidas na memória com os fluxos de um intervalo de tempo limitado;
 - Um SP é responsável por cruzar informações destas tabelas e reconstruir todas as *comunicações* do ambiente.
 - A tabela gerada a partir de *input* e *output* é denominada *Connections_Sessions*.

- *Connections_Sessions*, bidirecional.

InsideHost	OutsideHost	BytesIn	BytesOut	dPktsIn	dPktsOut
192.168.20.5	192.168.81.247	69	85	1	1
192.168.20.5	192.168.129.129	286	418	4	4
192.168.20.5	192.168.85.78	126	95	2	1
192.168.20.5	192.168.123.182	238	334	3	3
192.168.20.5	192.168.124.72	158	179	2	1
192.168.20.5	192.168.61.36	66	230	1	1
192.168.20.5	192.168.2.44	156	452	2	2
192.168.20.5	192.168.219.3	156	476	2	2

Ambiente

- Esquema de coleta dos fluxos de dados:



- Máquina coletora:
 - Pentium IV 1.8GHz, 256KB Cache (Socket 478);
 - 768MB RAM;
 - HD IDE 80GB ATA-100;

- Fluxo de rede (Netflow)
- Motivações
- Apresentação do modelo
 - Coletor
 - Banco de dados de fluxos
 - Ambiente
- Aferências e detecção de eventos (consultas SQL)
- Comparações com outras ferramentas
- Conclusão
 - Trabalhos em desenvolvimento

- Sintaxe do comando de consultas no MySQL:
SELECT select_expr
FROM table_references
[WHERE where_condition];

Contagem de fluxos

```
mysql> select count(*) from afm20071010;
+-----+
| count(*) |
+-----+
| 9213879 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> select count(*) as input
-> from afm20071010
-> where input=28;
+-----+
| input  |
+-----+
| 4128368 |
+-----+
1 row in set (24.19 sec)
```

```
mysql> select count(*) as output
-> from afm20071010
-> where output=28;
+-----+
| output |
+-----+
| 4772825 |
+-----+
1 row in set (3.59 sec)
```

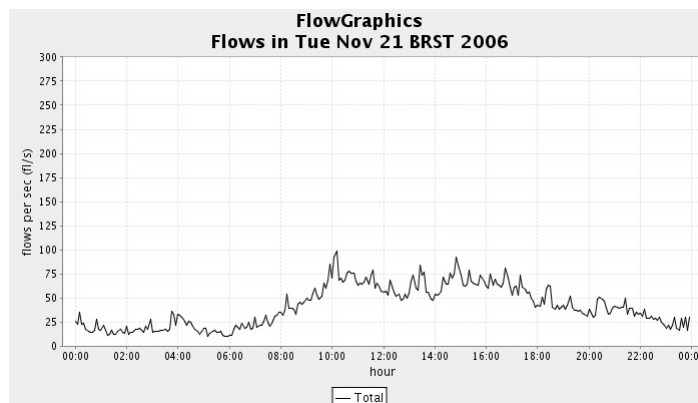
Média de fluxos por segundo.

```
mysql> select date_sub(first,interval second(first) second) as Date,count(*) as
flows, count(*)/60 as flows_per_sec from afm20071010 where inet_ntoa(srcaddr)
like '200.145.202.%%%' group by hour(first),minute(first) order by first;
```

Date	flows	flows_per_sec
2007-10-10 00:00:00	234	3.9000
2007-10-10 00:01:00	266	4.4333
2007-10-10 00:02:00	217	3.6167
2007-10-10 00:03:00	285	4.7500
2007-10-10 00:04:00	296	4.9333
2007-10-10 00:05:00	268	4.4667
2007-10-10 00:06:00	267	4.4500
2007-10-10 00:07:00	237	3.9500
2007-10-10 00:08:00	223	3.7167
2007-10-10 00:09:00	269	4.4833
...

1440 rows in set (13.18 sec)

- Gráfico gerado por aplicação JAVA utilizando dados obtidos no banco.



Os 10 hosts com maior número de bytes recebidos nos últimos 5 minutos.

```
mysql> select inet_ntoa(dstaddr) as dst_address, sum(dPkts)/300 as pkts_sec_in,
sum(dOctets)/300 as bytes_sec_in from last30minutes where input=28 and
first > date_sub(now(), interval 5 minute) group by dstaddr order by
sum(dOctets) desc limit 10;
```

dst_address	pkts_sec_in	bytes_sec_in
192.168.20.9	270.2733	335454.8567
192.168.10.91	208.0900	265581.1367
192.168.30.247	560.3167	62886.5700
192.168.30.74	29.5600	44336.2667
192.168.90.111	29.4100	39109.0867
192.168.10.41	28.6633	35481.3333
192.168.40.99	20.2567	30335.3100
192.168.10.46	18.8000	27250.6167
192.168.90.222	19.4033	24144.2300
192.168.90.96	12.3900	14373.5400

10 rows in set (0.65 sec)

Detecção de ataques: hosts realizando prospecções na rede.

```
mysql> select inet_ntoa(srcaddr) as src_address, count(distinct dstaddr) as
hosts_scanned, count(distinct dstport) as ports_scanned from last30minutes
where first > date_sub(now(), interval 30 minute) and input="28" and tcp_flags & 2 = "2"
and timediff(last, first) < "00:00:15" group by srcaddr having ports_scanned > 300 or
hosts_scanned > 300 order by hosts_scanned;
```

src_address	hosts_scanned	ports_scanned
10.139.136.159	1	3466
193.49.208.251	2	15968
192.152.161.101	8	980
196.109.112.136	8	741
192.188.2.109	29	652
192.96.239.18	510	1
192.188.154.181	573	4
192.188.249.167	581	4
192.188.226.145	1402	5

9 rows in set (0.78 sec)

Detecção de ataques de dicionário no SSH para uma determinada rede.

```
mysql> select inet_ntoa(srcaddr) as src_address,inet_ntoa(dstaddr) as dst_address,
count(*) as attempts from afm20071010 where dstport="22" and tcp_flags & 2 ="2"
and inet_ntoa(dstaddr) like "192.168.%%%.%%%" and timediff(last,first) < "00:00:15"
group by srcaddr,dstaddr having attempts > 100 order by attempts;
```

src_address	dst_address	attempts
192.194.17.103	192.168.30.204	113
192.17.109.114	192.168.30.21	177
194.77.105.210	192.168.30.73	319
192.17.109.114	192.168.211.65	447
192.17.109.114	192.168.30.11	686
192.17.109.114	192.168.30.230	795
192.17.109.114	192.168.30.17	940
192.17.109.114	192.168.30.62	1102
192.17.109.114	192.168.30.246	1176
192.17.109.114	192.168.30.58	1336

10 rows in set (5.90 sec)

Possíveis hosts utilizando serviços P2P.

```
mysql> select inet_ntoa(dstaddr) as Destination,count(distinct srcaddr) as
Num_Sources_Host,sum(dOctets) as Bytes,sum(dOctets)/time_to_sec(timediff(now(),
date_sub(now(), interval 30 minute))) as Bytes_per_second from last30minutes
where dstport>1024 and srcport>1024 and input=28 group by dstaddr having
Num_Sources_Host > 100 and Bytes > 1000000 order by Bytes;
```

Destination	Num_Sources_Host	Bytes	Bytes_per_second
192.168.30.107	131	1754815	974.8972
192.168.130.236	2494	2523820	1402.1222
192.168.20.175	141	7859254	4366.2522
192.168.30.42	2922	11820309	6566.8383
192.168.100.207	1911	11835903	6575.5017
192.168.30.25	1408	50975412	28319.6733
192.168.130.200	125	54694986	30386.1033

7 rows in set (0.32 sec)

- Fluxo de rede (Netflow)
- Motivações
- Apresentação do modelo
 - Coletor
 - Banco de dados de fluxos
 - Ambiente
- Aferências e detecção de eventos (consultas SQL)
- **Comparações com outras ferramentas**
- Conclusão
 - Trabalhos em desenvolvimento

Flow-tools

- Análise do tempo de consulta aos fluxos:

Tempo em segundos para realizar consulta aos fluxos		
Consulta	Banco de Dados	Flow-tools
Quantidade de fluxos em um dia.	0.00s	31.78s
Contagem de fluxos por minuto.	13.18s	31.78s
Tráfego de uma rede específica.	13.76s	33.52s
10 hosts com maior número de octetos recebidos (24 horas).	24.73s	53.43s

- Análise sobre o armazenamento dos fluxos:

Espaço utilizado em disco		
Consulta	7x10 ⁶ fluxos	1 fluxo
Banco de dados	~300 MB	45 bytes
Flow-tools	~134 MB	~20 bytes

- Fluxo de rede (Netflow)
- Motivações
- Apresentação do modelo
 - Coletor
 - Banco de dados de fluxos
 - Ambiente
- Aferências e detecção de eventos (consultas SQL)
- Comparações com outras ferramentas
- Conclusão
 - Trabalhos em desenvolvimento

- O estabelecimento de um modelo relacional para o armazenamento de fluxos permite:
 - Independência de bibliotecas para acessar os dados;
 - Grande versatilidade na manipulação dos fluxos;
 - Armazenamento de informações importantes para realização de monitoramento e perícias em redes.
- Diversos eventos puderam ser detectados:
 - Tentativas de prospecção;
 - Ataques de dicionário;
 - Possíveis atividades com *file sharing*;
 - Possíveis ataques de DoS;
 - Estatísticas em geral.

- Atualmente, esta infra-estrutura de armazenamento é a base de dois outros projetos:
 - Identificação de disseminação de artefatos maliciosos em redes de computadores;
 - Detecção de anomalias com sistemas inteligentes.
- Uma ferramenta obtida como consequência é:
 - Aplicação web para manipulação dos recursos do SGBD, geração de gráficos e estatísticas da rede, semelhante ao Flow-Scan.
 - Possibilidade de operações reativas (ex. regras temporárias de firewall).

- **Jorge Luiz Corrêa**
jorge@acmesecurity.org
PGP KeyID: 0x1BCB7255
- **André Proto**
andreproto@acmesecurity.org
PGP KeyID: 0xA6FC761A
- **Adriano Mauro Cansian**
adriano@acmesecurity.org
PGP KeyID: 0x3893CD28

Obrigado!