Conceitos e Configuração

de Agentes de Flows de Rede

O que é um agente ?

Agente é um software que captura as informações de pacotes IP que trafegam e são processadas pelo ativo de rede, agregando outras informações e encapsulam em datagramas UDP enviando para um coletor. Conceitualmente um agente Netflow é chamado de **Exporter** ao passo que um sFlow é chamado de **Agent** .

Características de um NetFlow Exporter

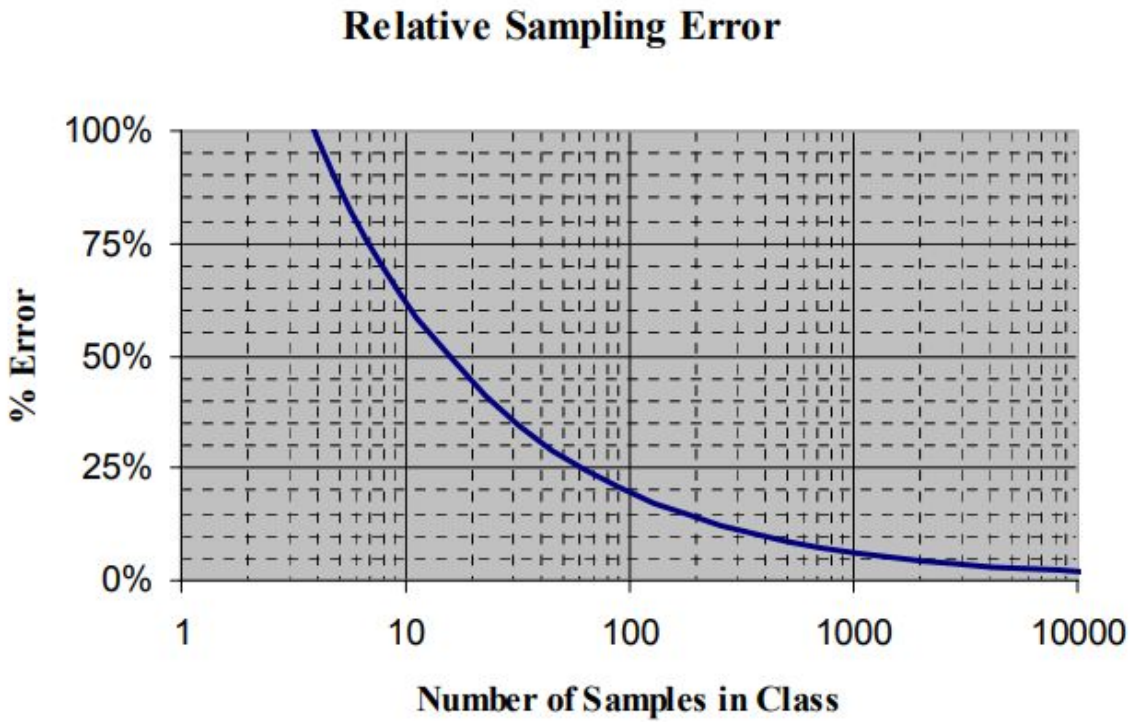
- O software, de acordo com a documentação, garante a captura de todos os pacotes que passam pelo ativo e pela interface ativa para captura.
- O programa Exporter dentro do ativo de rede captura as informações dos tráfegos por um determinado tempo, as processa e somente envia para um coletor se detecta o fim da conexão ou se a conexão não mais flui no tempo programado.
- A relação de campos que um NetFlow Exporter v9 pode enviar estão listados na RFC 3954 (https://www.ietf.org/rfc/rfc3954.txt) .

Características de um Agent sFlow

- O ativo de rede envia as informações do cabeçalho do pacote obtido pela amostragem, bem como as informações de interface de entrada e saída e outras informações como VLAN ID, máscara de rede, AS Path, next hop e os contadores da interface de rede e os envia para o agente e armazena. Estes dados de amostragem são processados e encapsulados em um datagrama UDP para ser enviado para um coletor..
- A relação de campos que um sFlow agent pode enviar estão listados na RFC 3176 (https://www.ietf.org/rfc/rfc3176.txt)

sFlow - Sampling - Amostragem

- Se caracteriza pela coleta 1/N .
- Utiliza a ideia de captura randômica evitando interferência por picos de tráfego que poderiam descaracterizar a amostra .
- Uma amostra sendo capturada por “x” tempo permite uma maior precisão e menor possibilidade de erro. Assim como uma amostragem menor.
- Uma explicação acadêmica do uso de amostragem para caracterizar tráfego de rede pode ser consultada aqui: https://sflow.org/packetSamplingBasics/



https://www.noction.com/blog/usage-based-billing-with-netflow-and-sflow

Agentes - exemplos de implementações

Tecnologia	Agentes/Exporters	Coletores e Frontends
sFlow	Aruba, Commscope, Dell, Extreme, F5, Fortinet, Huawei, Juniper, Linux(hsflowd) e FreeBSD (hsflowd), Windows(hsflowd).	nfdump-nfsen, ntop, PRTG, ELK, sflowtool*
IPFIX	Juniper, Mikrotik RouterOS , Linux(ipt-netflow), FreeBSD (ng_netflow), Windows(flowtraq).	PRTG, Scrutinizer, ELK
Netflow v9	Cisco, Mikrotik RouterOS , Linux(ipt-netflow), FreeBSD (ng_netflow), Windows (flowtraq).	nfdump-nfsen,ntop, PRTG, sflowtool*, ELK

Configuração - Netflow - Mikrotik - parte 1

- Diretiva ip traffic-flow - versões: 1,5,9 e IPFIX
- Opções do sub-menu ip traffic-flow
  - **interfaces**: Interfaces para habilitar para captura de flows. Default: todas as interfaces.
  - **cache-entries**: flows que podem ser armazenados na memória simultaneamente. O default é 4K .
  - **active-flow-timeout**: Qual o tempo de vida de um flow.
  - **inactive-flow-timeout:** Por quanto tempo o será considerado como ativo. Se após este tempo do pacote inicial não houver nenhum outro pacote relacionado a conexão , o flow é colocado como inativo e enviado ao coletor. Default 15 segundos.

Configuração - Netflow - Mikrotik - parte 2

- Sub-menu ip traffic-flow target
- Opções do sub-menu ip traffic-flow
  - **address**: IP:Porta do coletor que irá receber os datagramas. Podem ser configurados mais de 1.
  - **v9-template-refresh**: Número de pacotes depois dos quais o datagrama é enviado para o coletor. Default é 20.
  - **v9-template-timeout**: Se o datagrama não foi enviado depois de quanto tempo decorrido ele vai ser.
  - **version**: Qual versão será utilizada.

Configuração - Netflow - Mikrotik - parte 3

[admin@RouterOS] > **ip traffic-flow**
[admin@RouterOS] /ip traffic-flow> **set enabled=yes**
[admin@RouterOS] /ip traffic-flow> **set interfaces=ether1**
[admin@RouterOS] /ip traffic-flow> **print**
[admin@RouterOS] /ip traffic-flow> **target**
[admin@RouterOS] /ip traffic-flow target> **add dst-address=177.8.96.2 port=9995 version=9**
[admin@RouterOS] /ip traffic-flow target> **print detail**

Configuração - sFlow - Debian - hsflowd - parte 1

```
$ cd /opt
$ sudo wget
https://github.com/sflow/host-sflow/releases/download/v2.0.38-1/hsflowd-ubuntu2
0_2.0.38-1_amd64.deb
$ sudo dpkg -i hsflowd-ubuntu20_2.0.38-1_amd64.deb
$ sudo systemctl enable hsflowd
$ sudo vi /etc/hsflowd.conf            # Detalhes no próximo slide
$ sudo systemctl start hsflowd
```

Configuração - sFlow - Debian - hsflowd - parte 2

$ sudo vi /etc/hsflowd.conf

sflow {
  agent = ens192 # Interface que onde o agent irá escutar
  sampling.10G = 1 #Tx de amostragem. Interface detectada no modo debug
  collector { ip=177.8.96.2 udpport=9996 } #Coletor e porta para envio de flows
  pcap { dev = ens192 } # interface que serão capturados o tráfego
}

Configuração de um 2º coletor - parte 1

Podemos configurar nossos agentes para envio para mais de 1 coletor caso desejamos ter uma redundância de informações ou por exemplo utilizar um outro tipo de aplicação para interagir com os flows que estamos recebendo.

Exemplo:

- ELK - Elastiflow
- Fastnetmon
- Sflowtool

Configuração de um 2º coletor - parte 2

Vamos apresentar neste tutorial um exemplo de uso do sflowtool. Este aplicativo recebe datagramas UDP **sFlow** e fornece uma saída dos flows em ASCII e JSON por exemplo. Pode enviar via pipeline para um tcpdump, replicar o tráfego para outros coletores e até encapsular em UDP no formato netflow.

Mais informações: https://github.com/sflow/sflowtool

Configuração de um 2º coletor - parte 3

Mostra os flows e suas informações linha a linha separados por “,”
$ sflowtool -p 9997 -l

Mostra os flows em formato JSON
$ sflowtool -p 9997 -J

Converte em formato tcpdump
$ sflowtool -p 9997 -t | tcpdump -r -