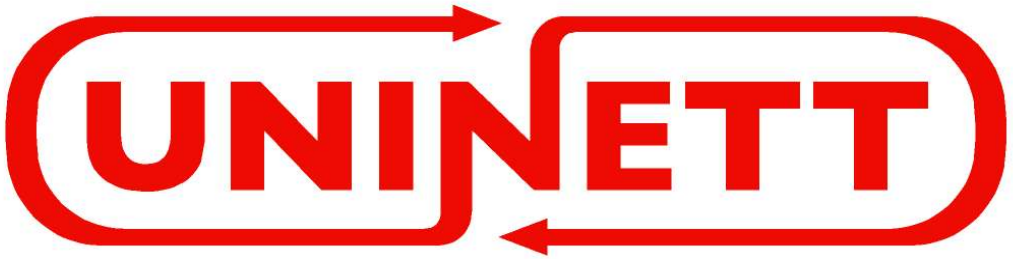


A Scalable Monitoring Platform for the Internet (SCAMPI)

RIPE 48 Meeting

5th May 2004

Arne Øslebø

UNINETT AS

SCAMPI overview

- 2 ½ year 5th Framework project
- Started April 2002
- 10 partners:
  - CESNET
  - FORTH
  - FORTHNET
  - IMEC
  - Leiden University
  - Masaryk University
  - NETIKOS
  - Siemens
  - TERENA
  - UNINETT

- Main goals:
  - Development of a high-performance intelligent monitoring adapter for 10Gbps
  - Development of an open and extensible architecture for network monitoring.
  - Development of monitoring and measurement tools
  - Investigate strategies and methodologies for monitoring systems operating at 100 Gbps and beyond.

SCAMPI adapter

- Developed by CESNET and Masaryk University
- Based on Liberouter project
- Open source
- Three cards:
  - Motherboard
  - Interface card
  - Timestamp unit
- 4-port 1GE interface card
- 2-port 10GE interface card
- Each card has programmable FPGAs, CAMs and DRAM.
- Nanoprocesors

Motherboard

UNINETT

Interface card

Hardware functions

- Header filtering (BPF syntax)
- Packet sampling (deterministic and probabilistic)
- Payload string searching
- Support for multiple simultaneous applications with different filtering/sampling requirements.

MAPI

- Monitoring Application Programming Interface
- Design goals:
  - Make it quick and easy to implement new monitoring applications
  - Low overhead
  - Support for multiple concurrent users and applications
  - Global optimization
    - Optimize processing of packets based on all applications from all users.
  - Transparent support for different hardware adapters
  - Easy to extend
    - New drivers
    - function libraries
- Support for:
  - SCAMPI adapter
  - DAG cards
  - NIC

UNINETT

MAPI basics

- Network flow
  - mapi_create_flow
  - Initially all packets seen on the network
- Apply functions to a flow
  - mapi_apply_functions
  - BPF filter, string search, packet counter, byte counter, Netflow, jitter etc.
- Read results
  - mapi_read_result
- MAPId
  - Daemon that communicates with hardware devices and processes packets in software.

MAPI example

Virus detection:

```
fd=mapi_create_flow("/dev/dag0");
mapi_apply_function(fd,BPF_FILTER,"src port 1234");
ctr_id1=mapi_apply_function(fd,PKT_COUNTER);
mapi_apply_function(fd,STR_SEARCH,"pattern",100,300);
ctr_id2=mapi_apply_function(fd,PKT_COUNTER);
mapi_apply_function(fd,TO_FILE,MFF_TCPDUMP,"tcpdump.dat",0);
mapi_connect(fd);

wile(1) {
      mapi_read_results(fd,ctr_id1,&ctr_val1);
      mapi_read_results(fd,ctr_id2,&ctr_val2);

      printf("BPF match: %d String search match: %d\n",
            ctr_val1,ctr_val2);
      sleep(10);
}
```

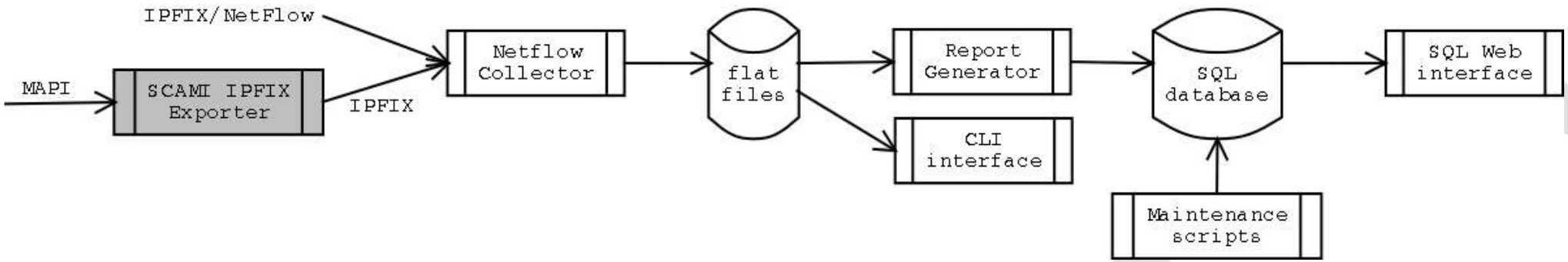

UNINETT

Applications

- Intrusion detection
  - uses Snort signatures
- QoS application
  - packet loss
  - jitter
  - delay
- Flowrep
  - generic report generator with a web frontend for Netflow/IPFIX records

Flowrep

- Generic netflow based report application
- Flow-tools
- Reports stored in database
- Web frontend



UNINETT