

Lobster project

Sven Ubik, CESNET
Zurich, 14 April 2005

General purpose of network monitoring:

1. What the network does with user traffic?
2. What is the current state of the network?
3. Is there some problem in the network?

Monitoring questions ...

- How long it takes to transport user data?
- How many packets get lost?
- What is available bandwidth and how it fluctuates?
- Why is my TCP connection slow on this fast network?
- What applications are people using most? (is our academic network loaded by sharing movies?)
- What is the performance of the DNS system?
- Are there viruses or worms being spread in the network?
- Is somebody doing some computer network attack?

Active vs. passive

Active monitoring is a probe into the network

- What is travelling next to the probe?
- Is the probe experiencing the same trip as others?

Passive monitoring is a watch

- I can see everything, but what can I analyse from it?

Active vs. passive

- How long it takes to transport user data?
- How many packets get lost?
- What is available bandwidth and how it fluctuates?
- Why is my TCP connection slow on this fast network?
- What applications are people using most? (is our academic network loaded by sharing movies?)
- What is the performance of the DNS system?
- Are there viruses or worms being spread in the network?
- Is somebody doing some computer network attack?

SCAMPI: Scaleable monitoring platform
for the Internet



IST project (April 2002 – March 2005)

- to develop programmable monitoring hardware
- to provide platform for easy writing of portable monitoring applications

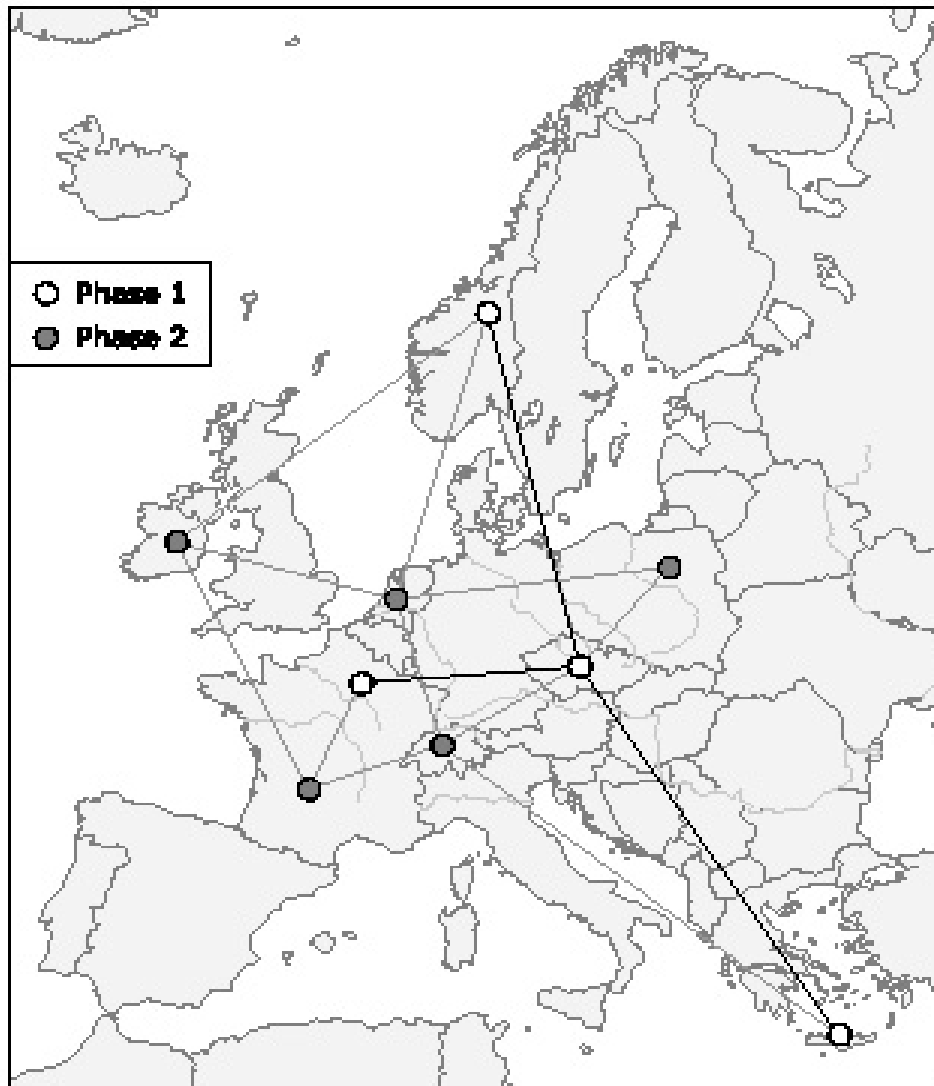
LOBSTER: Large Scale Monitoring of Broadband Internet Infrastructure

Specific Support Action (2005-2006)

- Deploy pilot passive monitoring architecture
- Add configurable anonymization
- Add support for distributed monitoring

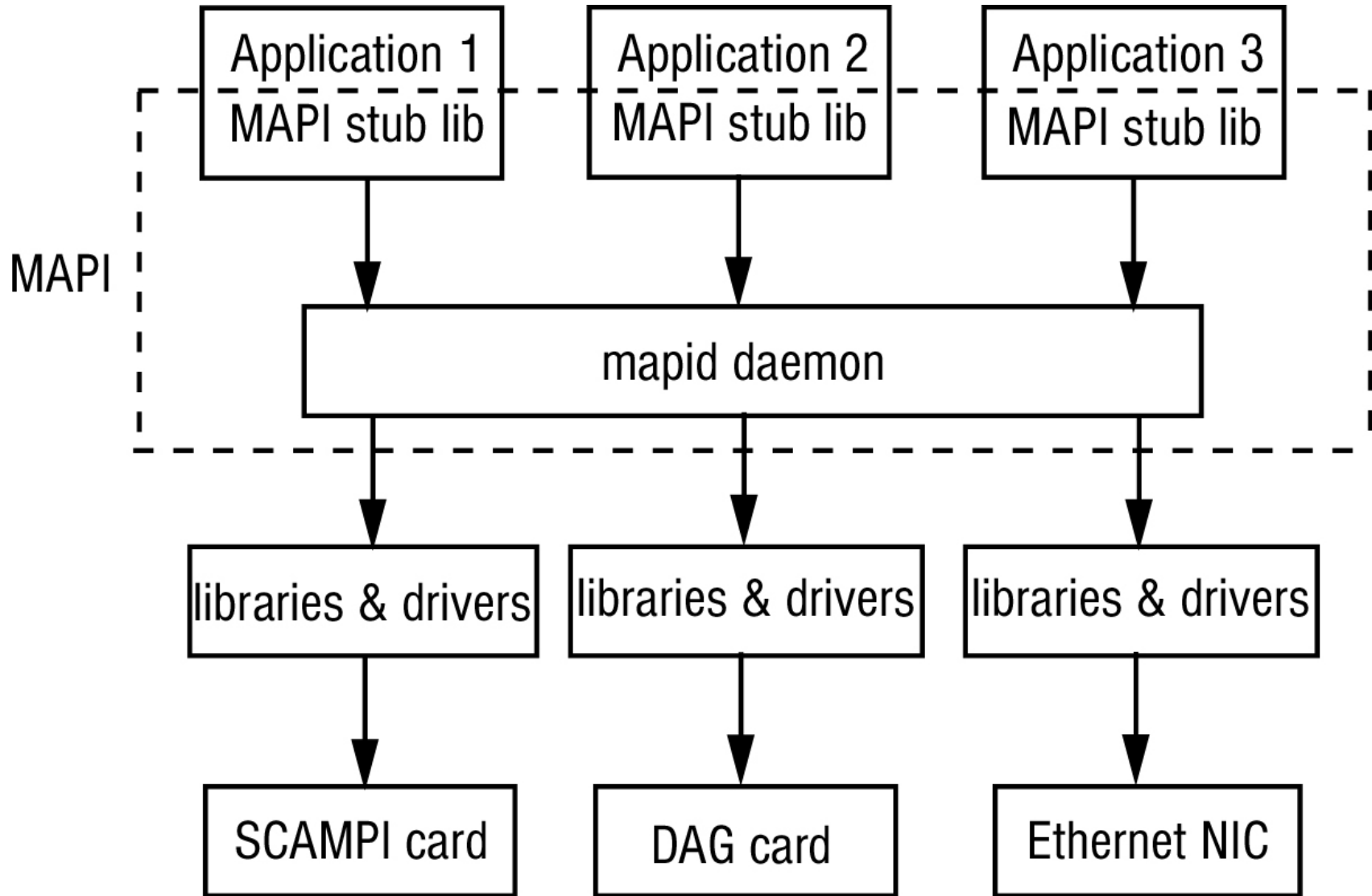


Lobster network



- Detection of security attacks
 - intrusion and DoS attacks
- Characterization of traffic using dynamic ports
 - file sharing, etc.
- Short-timescale bandwidth
 - continuous, precise, non-intrusive used/available bw.
- Packet loss rate
 - for real contracted traffic
- Providing packet trace repository
 - for researchers

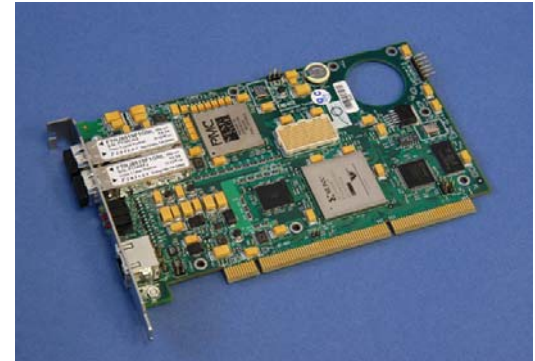
SCAMPI architecture



Hardware offloading

Scampi card, DAG card with coprocessor:

- Header filtering
- Packet sampling
- Time and length statistics
- Payload searching



Possibilities and software support differ in many details for Scampi card and DAG card

- *Flow* - all packets from one monitoring point with applied sequence of monitoring functions
- *Application* - can open multiple flows
- *mapid* - automatically decides between hardware or software implementation
- One mapid can support multiple applications

Example application

```
fd = mapi_create_flow("/dev/scampi/0");

mapi_apply_function(fd, "BPF_FILTER", "src net 192.168.1.0/24");
mapi_apply_function(fd, "STR_SEARCH", "virus");
fid = mapi_apply_function(fd, "PKT_COUNTER");

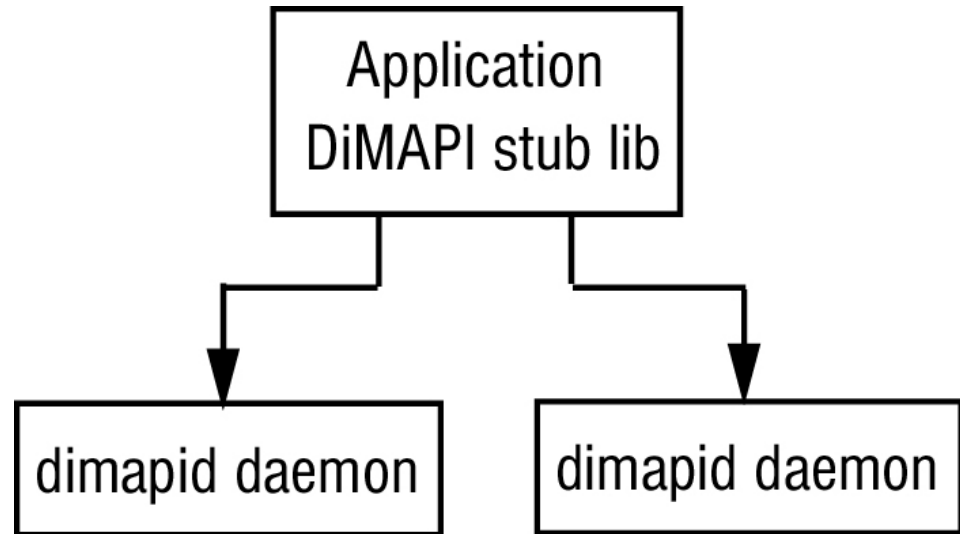
mapi_connect(fd);

while(1) {
    sleep(5);
    cnt = mapi_read_results(fd, fid, MAPI_REF);
    printf("Packets with virus: %d\n", *cnt);
}
```

MAPI functions

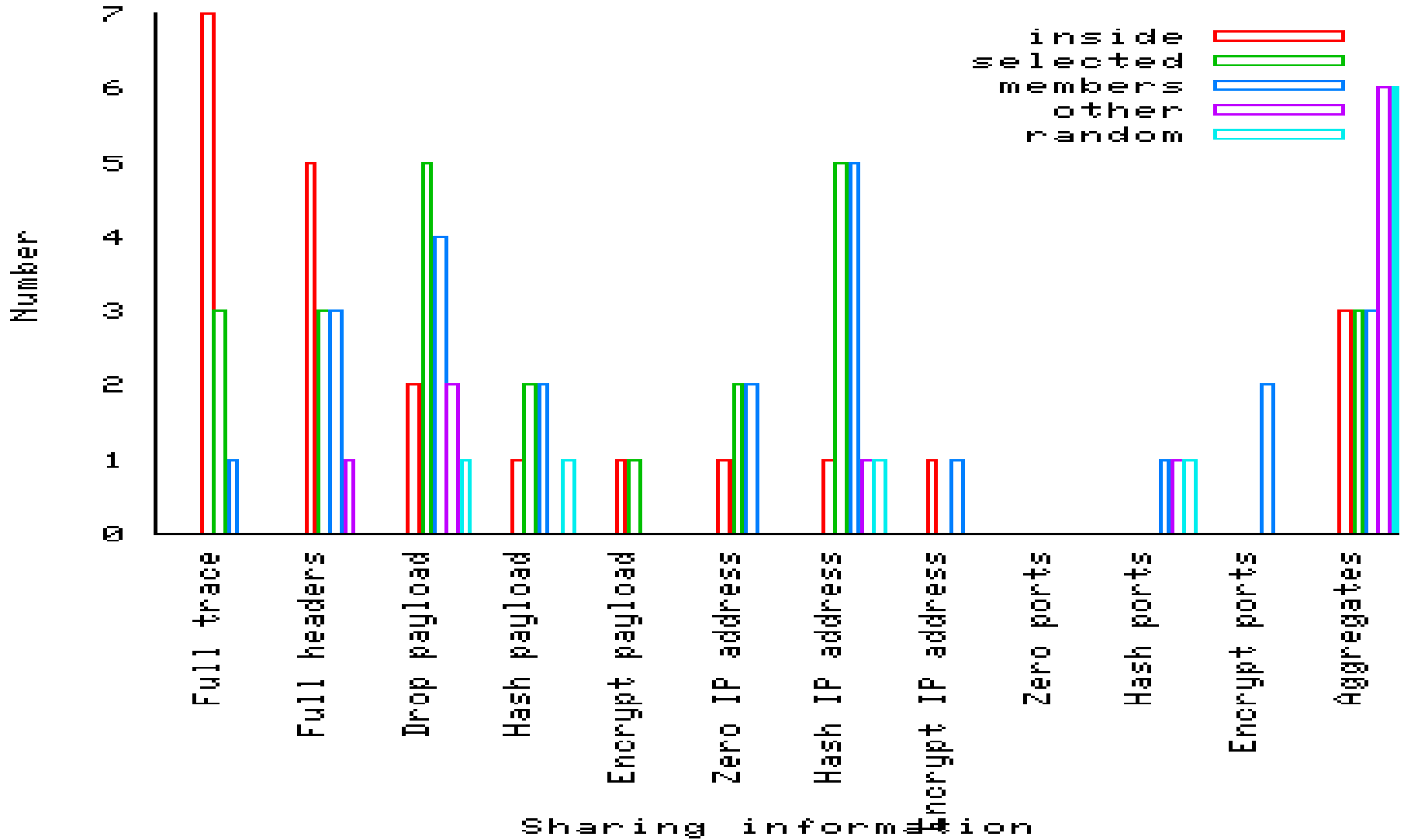
BPF_FILTER	header filtering
PKT_COUNTER	packet counter
BYTE_COUNTER	byte counter
STR_SEARCH	payload searching
TO_BUFFER	reading packets
SAMPLE	packet sampling
HASHSAMP	hash-based packet sampling
TO_FILE	storing packet trace to disk
ETHERREAL	ethereal-based filtering
HASH	packet hash computing
COOKING	TCP/IP reassembly
BUCKET	divides packets into buckets based on their timestamps
THRESHOLD	signals when a threshold is reached

- *Scope* – a set of remote monitoring sensors
- *Flow* – can span a scope



- DiMAPI monitoring functions:
 - a) Can be passed to monitoring sensors
e.g., header filtering
 - b) Can be applied to a scope as the whole
e.g., aggregation or distance-based metrics

Required anonymization



Anonymization options

```
mapi_apply_function(fd, "ANONYMIZE", protocol, field, function, args);
```

STRIP

HASHED

RANDOM

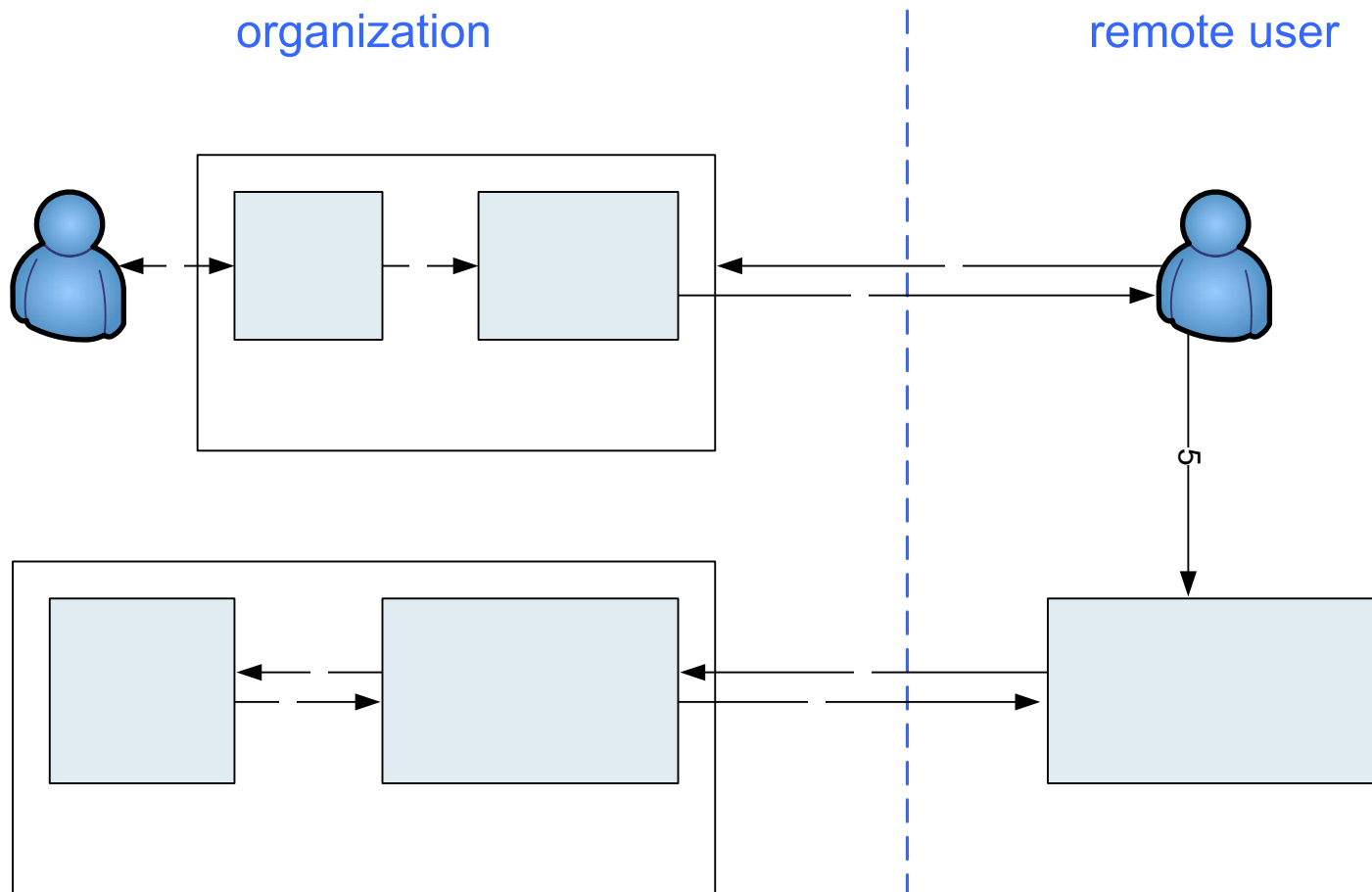
MAP

MAP_DISTRIBUTION

PREFIX_PRESERVING

- first tier anonymization (common to all flows, preferably in HW)
- second tier anonymization (flow-specific)

Anonymization, cont



Credential example

Authorizer: "RSA:abc123" # Admin's key

Licensees: "RSA:xyz999" # User's key

Conditions: ANONYMIZE == "defined" &&

ANONYMIZE.0.pos == 0 &&

ANONYMIZE.0.param.0 == IP &&

ANONYMIZE.0.param.1 == SRC_IP &&

ANONYMIZE.0.param.2 == PREFIX_PRESERVING &&

ANONYMIZE.1.pos == 1 &&

ANONYMIZE.1.param.0 == IP &&

ANONYMIZE.1.param.1 == DST_IP &&

ANONYMIZE.1.param.2 == PREFIX_PRESERVING &&

ANONYMIZE.2.pos == 0 &&

ANONYMIZE.2.param.0 == TCP &&

ANONYMIZE.2.param.1 == PAYLOAD &&

ANONYMIZE.2.param.2 == STRIP;

} anonymize
source IP

} anonymize
destination IP

} remove
payload

Signature: "RSA-SHA1:213344f9"

Lobster resources

- Project website <http://www.ist-lobster.org>
- Subscribe to lobster-news@ist-lobster.org
- Lobster tutorial on RIPE 50 meeting, Stockholm, 6 May 2005
- 1st Lobster workshop on TNC2005, Poznan, 7 June 2005

Thank you for your attention

Famous worm outbreaks:

– Summer 2001: CODE RED

worm

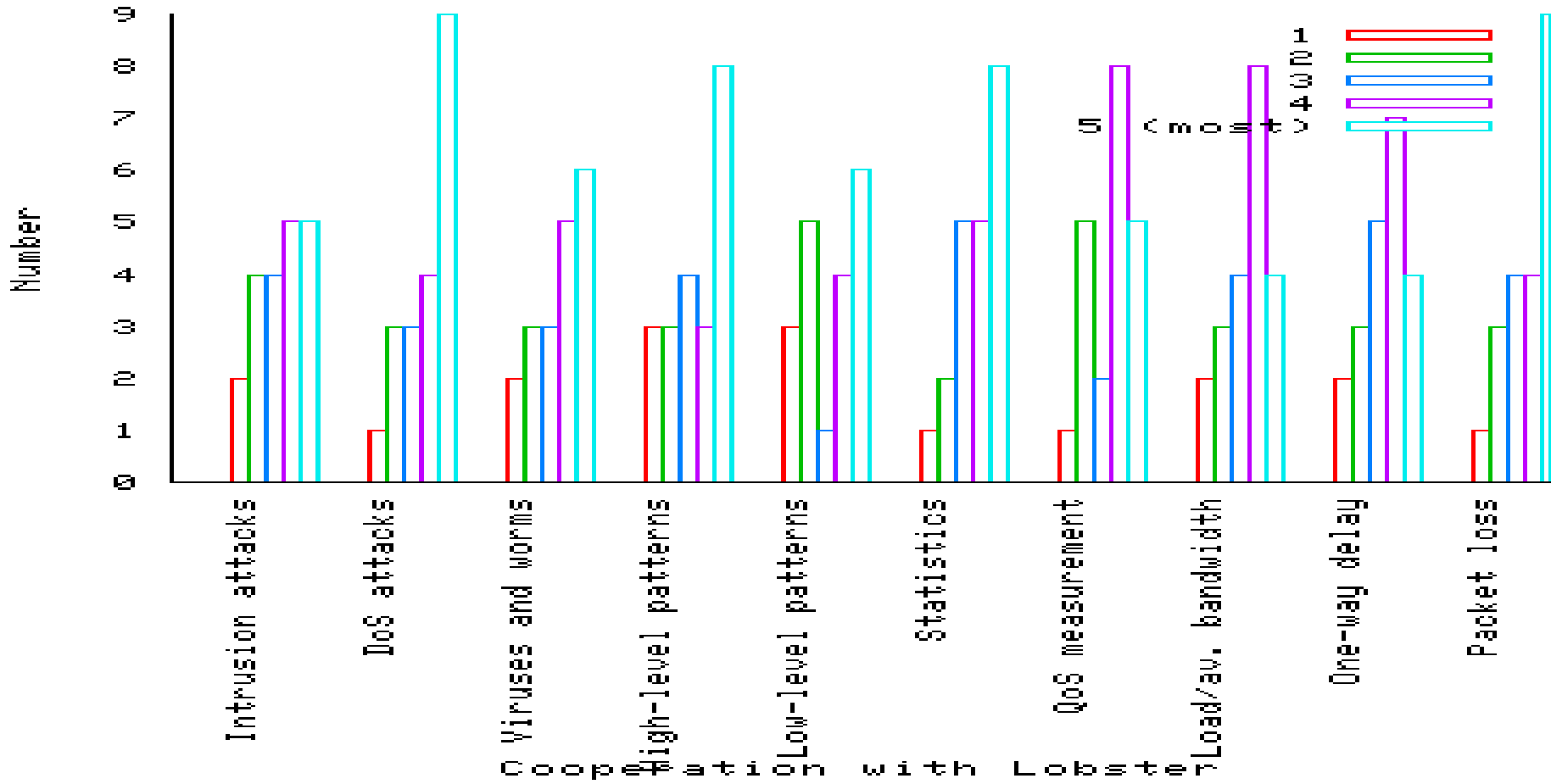
- Infected 350,000 computers in 24 hours

– January 2003: Sapphire/Slammer worm

- Infected 75,000 computers in 30 minutes

– March 2004: Witty Worm

- Infected 20,000 computers in 60 minutes



- Research organizations
 - ICS-FORTH, Greece
 - Vrije University, The Netherlands
 - TNO Telecom, The Netherlands
- NRNs/ISPs, associations
 - CESNET, Czech Republic
 - UNINETT, Norway
 - FORTHNET, Greece
 - TERENA, The Netherlands
- Industrial partners
 - ALCATEL, France
 - Endace, UK

Can we start response BEFORE all computers have

been infected? Yes! But we need:

- Smart Internet traffic monitoring sensors

- Capable of detecting new worms

- Distributed infrastructure of Internet traffic sensors

- More sensitive to attacks

- pinpoint attacks as soon as they emerge

- Spread information about new worms fast

69% of the traffic is
unaccounted-for