Identifying Operating System Using Flow-based Traffic Fingerprinting

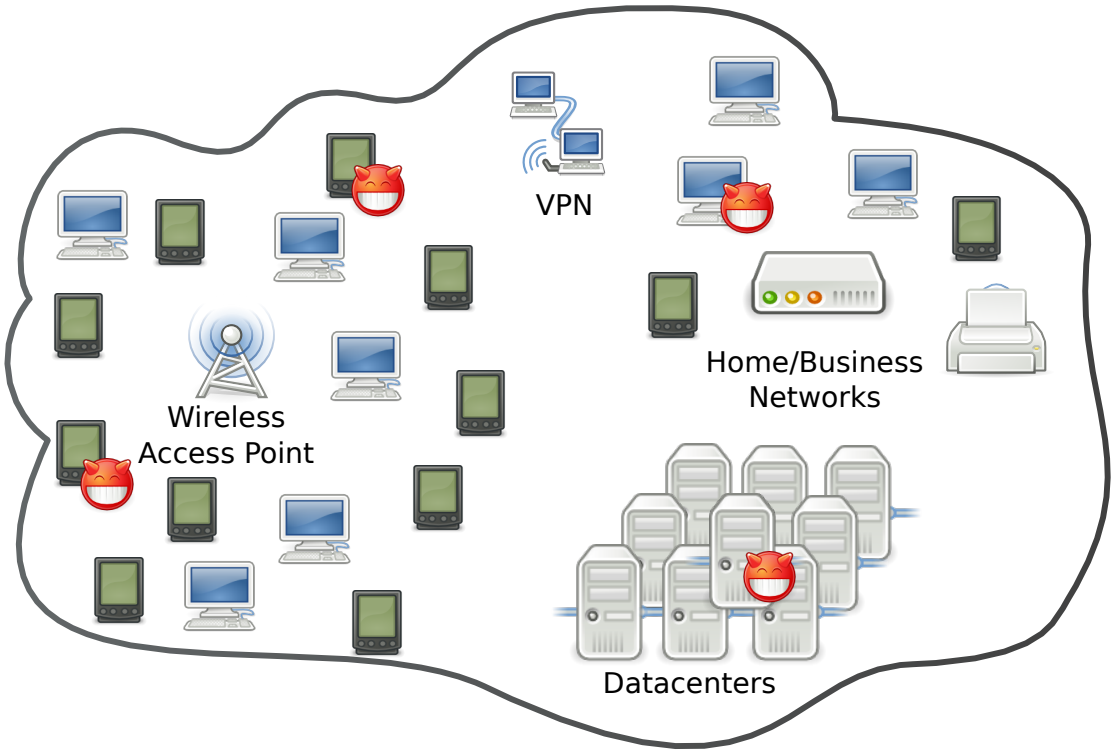
Tomáš Jirsík, Pavel Čeleda

{jirsik|celeda}@ics.muni.cz
Institute of Computer Science, Masaryk University

EUNICE 2014
September, 1. – 5., Rennes, France

Motivation

- Increasing number of devices
- Management of devices
- Security issues



State of the Art

Active Approach

- Higher precision of detection
- Inserts other traffic
- Needs to scan each host

Passive Approach

- Lower detection precision
- Transparency
- Large-scale network detection

Detection Methods

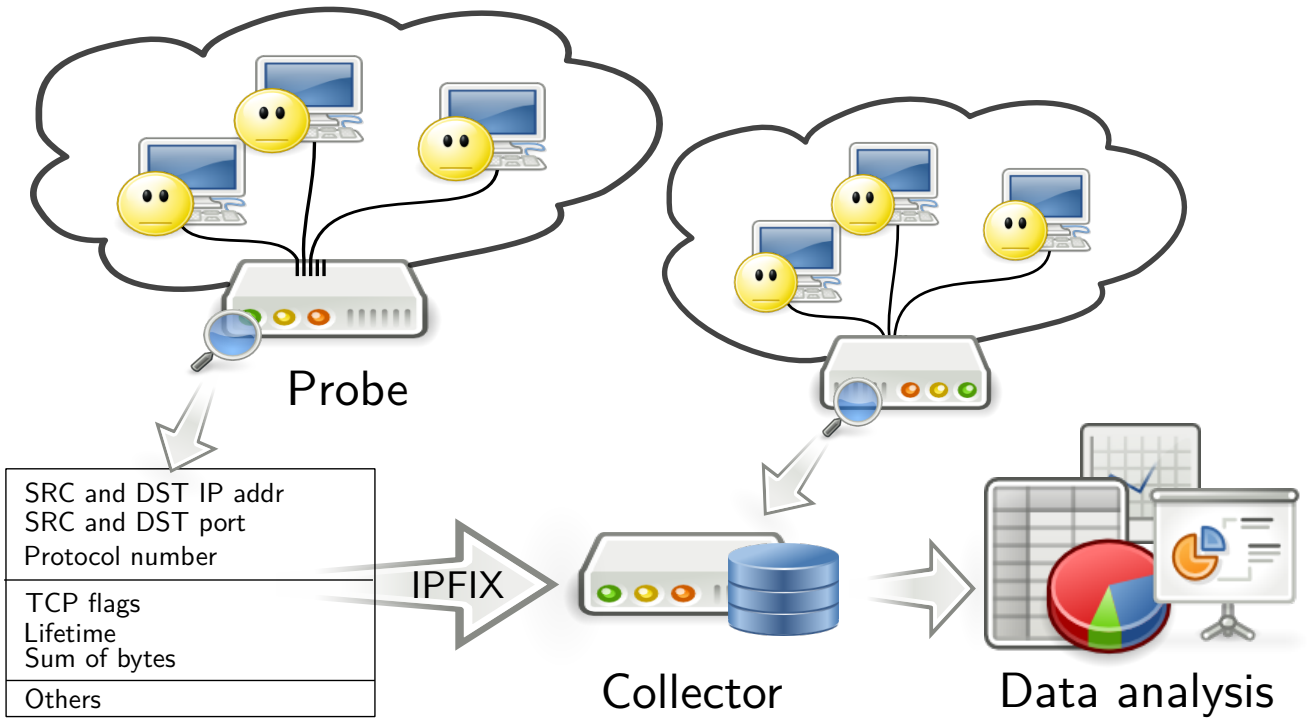
L3 - L4: Network and Transport Layer

OS	TTL	SYN packet size	TCP window size
Windows XP	128	48	65535
Windows 7	128	52	8192
Ubuntu	64	60	29200
Mac OS X	64	64	65535

L7: Application Layer

OS	Browser	User-Agent
Windows 7	Chrome	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.102 Safari/537.36
Ubuntu	Firefox	Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:29.0) Gecko/20100101 Firefox/29.0
Mac OS X	Safari	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_5) AppleWebKit/537.73.11 (KHTML, like Gecko) Version/6.1.1 Safari/537.73.11

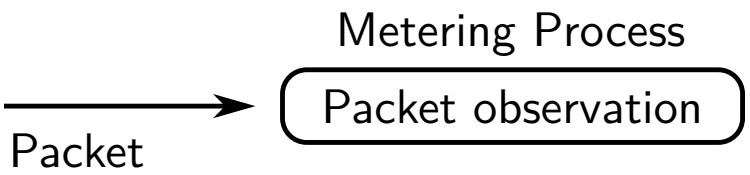
Network Flows



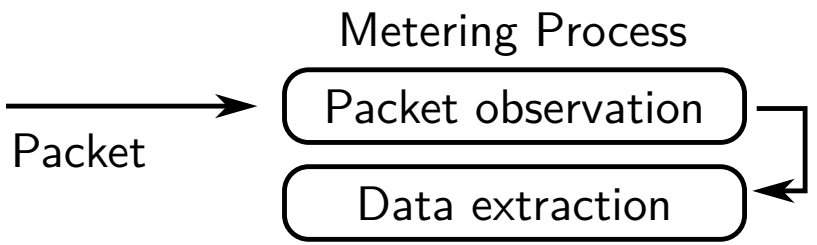
Flow start	Duration	Proto	Src IP Addr:Port		Dst IP Addr:Port	Flags	Packets	Bytes
09:41:21.763	0.101	TCP	172.16.96.48:15094	->	209.85.135.147:80	.AP.SF	4	715
09:41:21.893	0.031	TCP	209.85.135.147:80	->	172.16.96.48:15094	.AP.SF	4	1594

Architecture Design

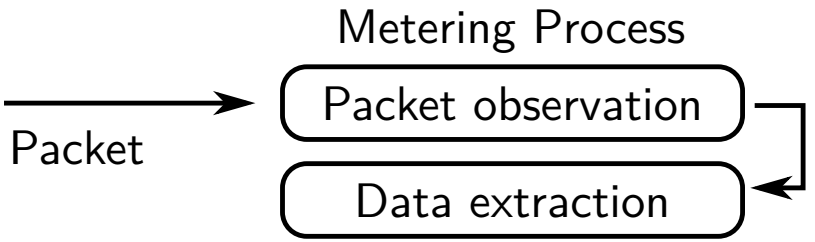
Architecture Design



Architecture Design

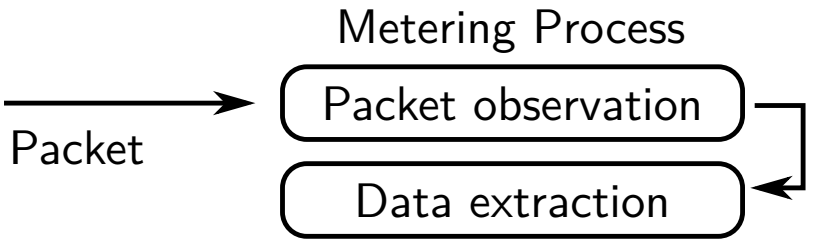


Architecture Design



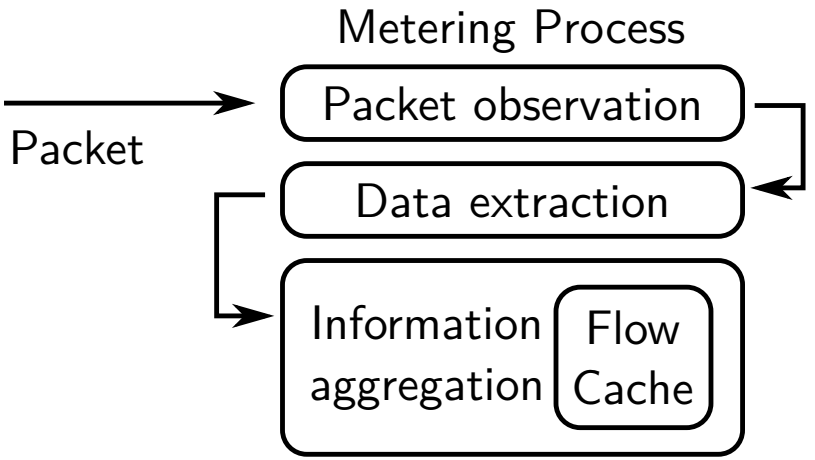
$$F = (IP_{src}, IP_{dst}, P_{src}, P_{dst}, Proto,$$

Architecture Design



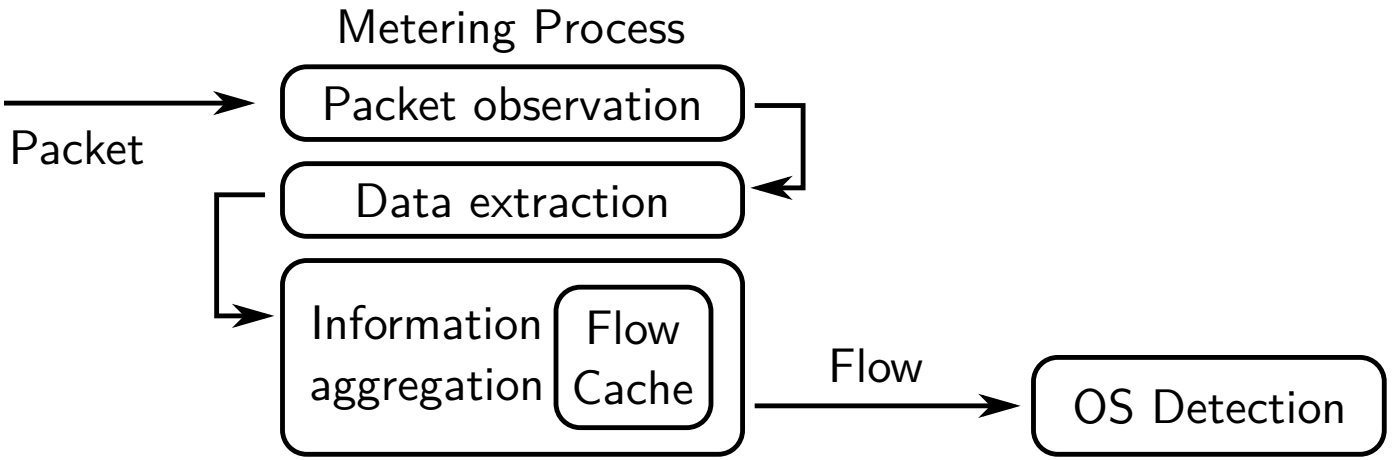
$$F = (IP_{src}, IP_{dst}, P_{src}, P_{dst}, Proto, TTL, Size_{SYN}, Size_{WIN}, UA)$$

Architecture Design



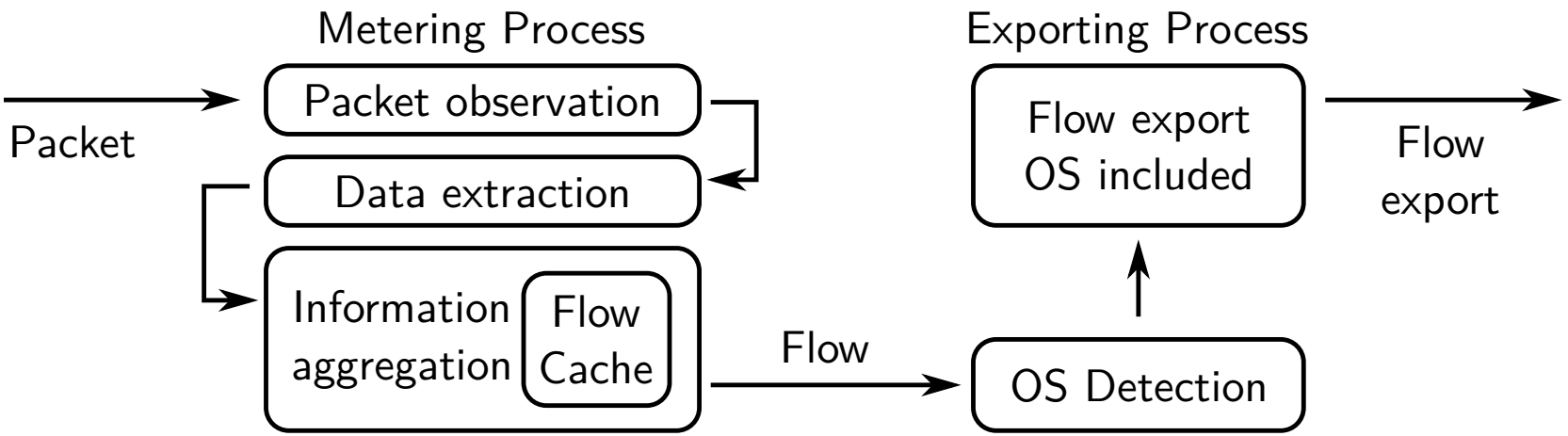
$$F = (IP_{src}, IP_{dst}, P_{src}, P_{dst}, Proto, TTL, Size_{SYN}, Size_{WIN}, UA)$$

Architecture Design



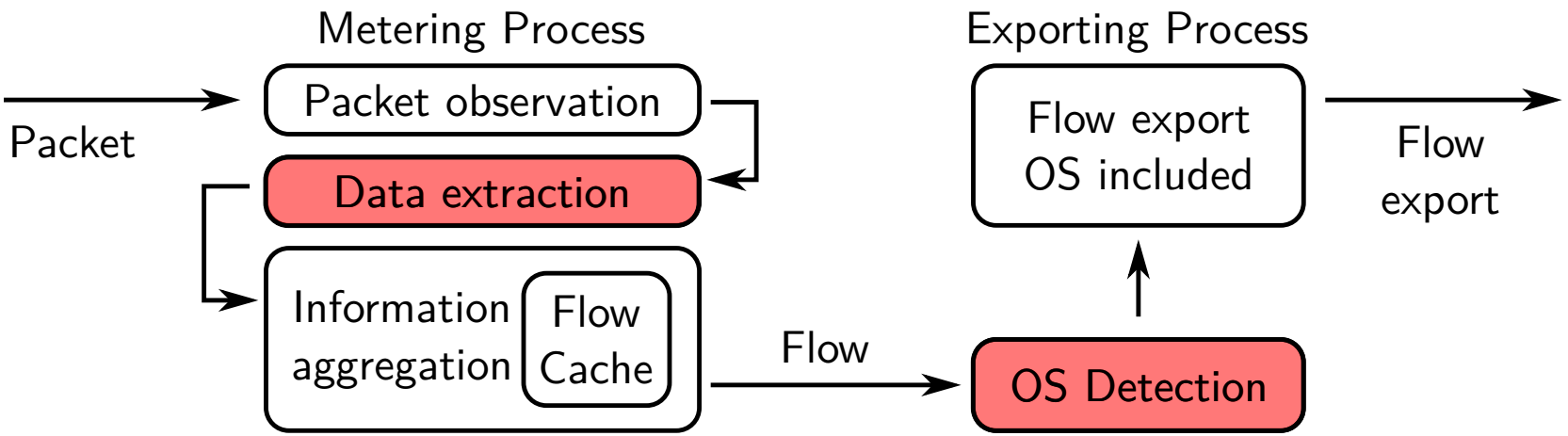
$$F = (IP_{src}, IP_{dst}, P_{src}, P_{dst}, Proto, TTL, Size_{SYN}, Size_{WIN}, UA)$$

Architecture Design



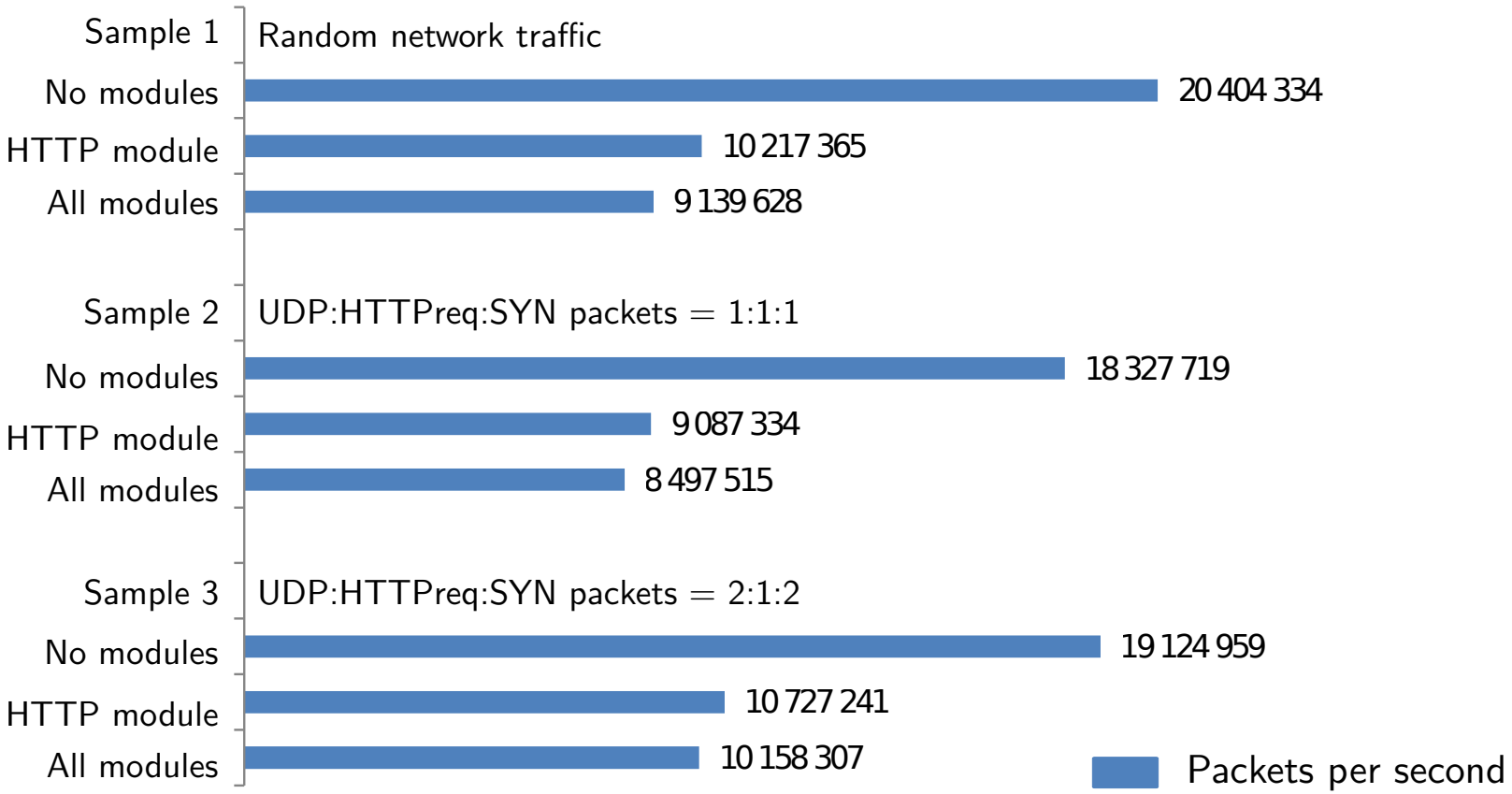
$$F = (IP_{src}, IP_{dst}, P_{src}, P_{dst}, Proto, TTL, Size_{SYN}, Size_{WIN}, UA)$$

Architecture Design



$$F = (IP_{src}, IP_{dst}, P_{src}, P_{dst}, Proto, TTL, Size_{SYN}, Size_{WIN}, UA)$$

Benchmark


Sample 1
Random network traffic
No modules
20 404 334
HTTP module
10 217 365
All modules
9 139 628
Sample 2
UDP:HTTPreq:SYN packets = 1:1:1
No modules
18 327 719
HTTP module
9 087 334
All modules
8 497 515
Sample 3
UDP:HTTPreq:SYN packets = 2:1:2
No modules
19 124 959
HTTP module
10 727 241
All modules
10 158 307
Packets per second

Deployment

Data set: 2 hours of traffic, 10.221 M flows, 12 897 hosts

# of unique OS	# of IP in A	% of all A	# of IP in B	% of all B
1	7898	87.059	3996	95.989
2	1071	11.806	159	3.819
3	80	0.882	7	0.168
>3	23	0.253	1	0.024
Total	9072	100 %	4163	100 %

Number of unique OS detected at one IP:
A - whole network, B - dynamically addressed subnets removed

Summary

- Large scale detection
- Flow based OS detection framework
- High performance

Future Work

- Deep analysis of User-Agents
- Fingerprint correlation
- Fingerprint database improvement

Thank you for your attention!

Tomáš Jirsík, Pavel Čeleda

{jirsik|celeda}@ics.muni.cz