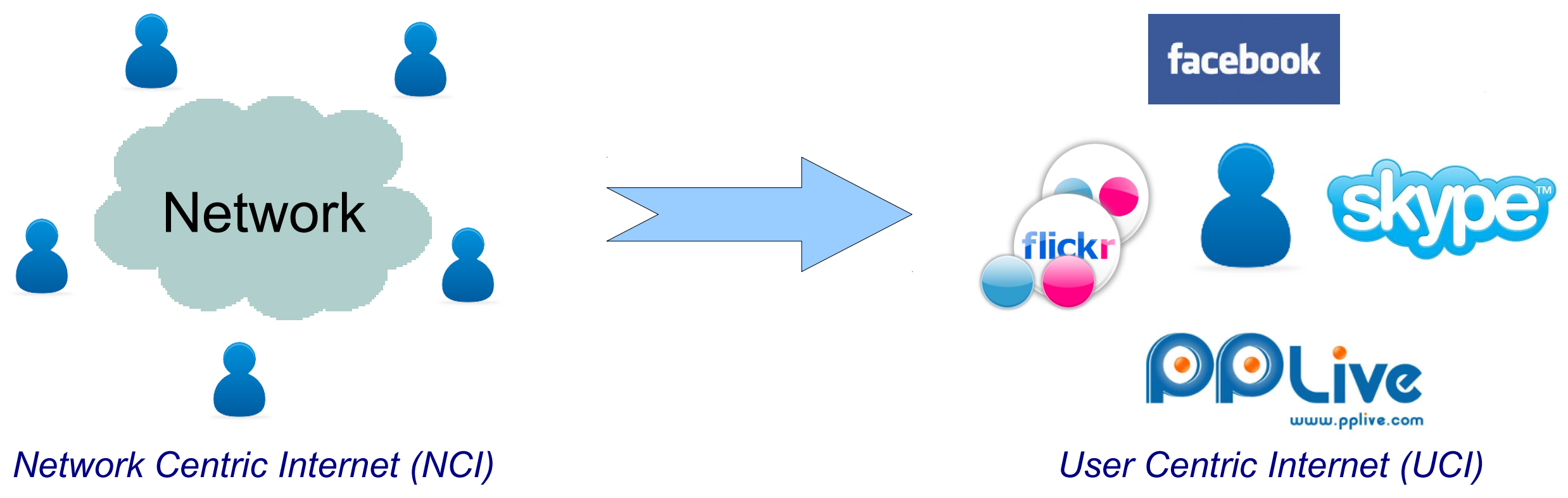
On the Characterization of Multi-Channel Applications

Walter de Donato and Antonio Pescapè
Dipartimento di Informatica e Sistemistica, University of Napoli “Federico II” (Italy)
{walter.dedonato, pescape}@unina.it

Introduction and Motivations

The Internet is evolving from the Network-Centric view to the User-Centric view. The user increasingly takes an active role in the network, promoting peer-to-peer (P2P) and many-to-many interactions, sharing his wide-band connection and providing both contents and network functionalities.



The transition to the User-Centric Internet (UCI) is fostering the development of **multi-channel applications**. Such applications **provide a single interface to perform heterogeneous activities exploiting many communication channels**.

Characterizing multi-channel applications has implications in many networking fields:

- Capacity planning and provisioning
- Traffic engineering
- Fault diagnosis
- Policy enforcement
- Intrusion and anomaly detection
- Billing
- Network neutrality

Traditional network traffic analysis methodologies are less and less effective [1]

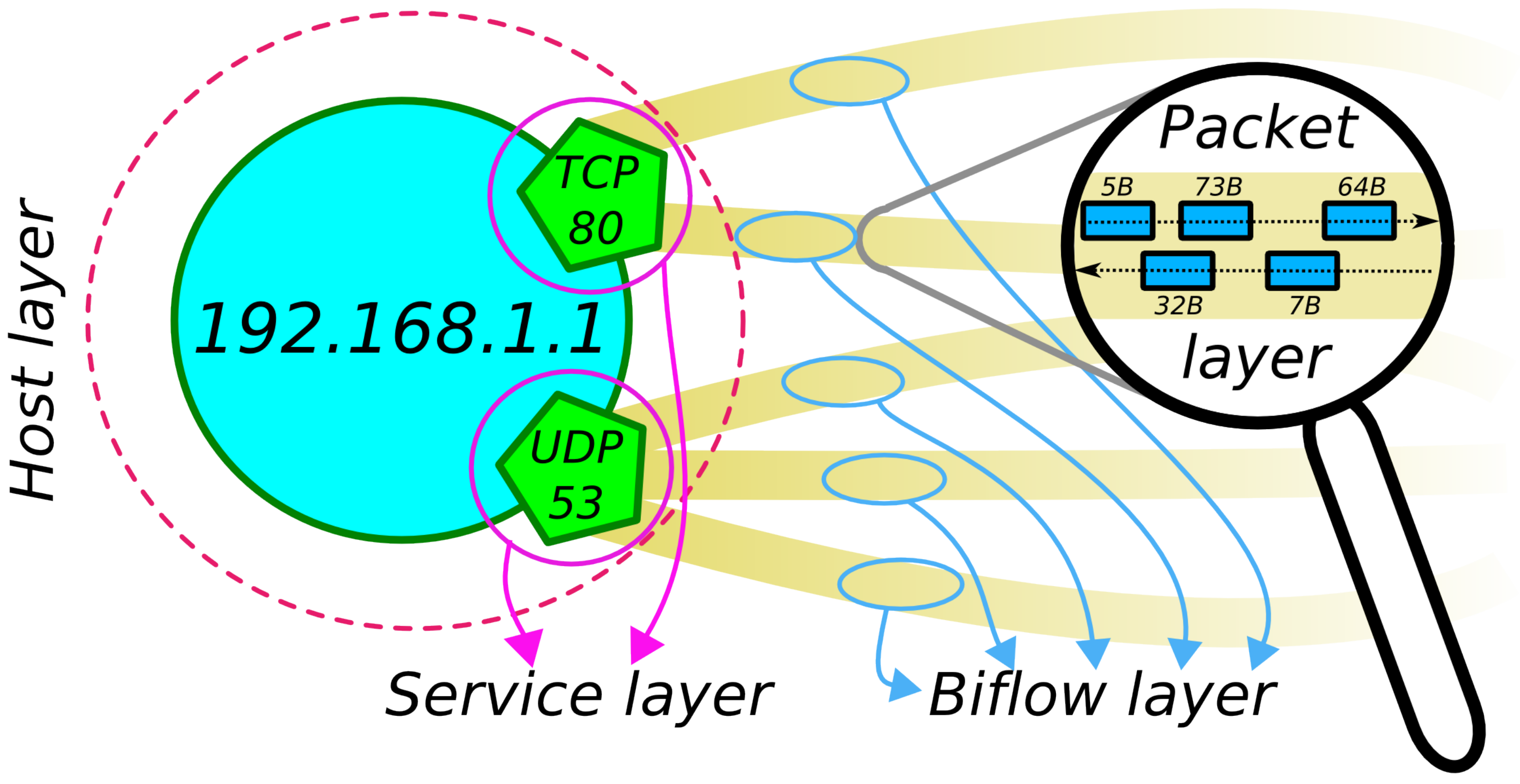
- ignore relations among nodes and among communication channels
- cannot usually deal with of obfuscation, encapsulation and encryption

Therefore, it is necessary to find new techniques and analysis methodologies purposely designed for the properties of emerging applications.

The Proposed Methodology

We propose the definition of a novel methodology for the characterization of multi-channel applications based on a multi-layer traffic inspection and a decomposition approach, as depicted in figure, counting four layers:

- **Host** - aggregates the whole traffic pertaining to a single host
- **Service** - groups together packets having the same transport protocol and IP address-port pair.
- **Biflow** - aggregates packets having the same 5-tuple, where source and destination can be swapped
- **Packet** - looks at the properties of each packet (e.g. size, inter-packet time, payload, ...)



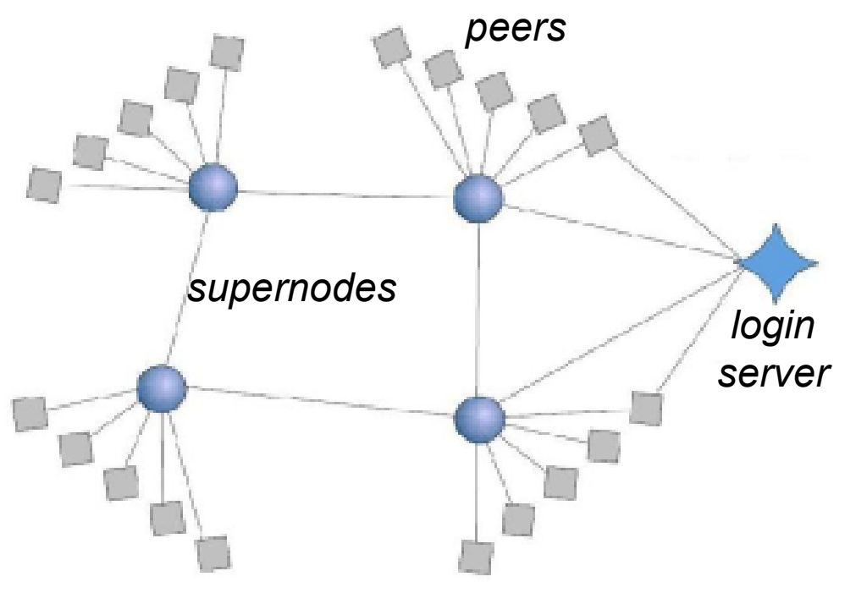
Combining information collected at these layers can reveal useful patterns otherwise not visible. For instance:

- by looking at many biflows belonging to the same application it is possible to detect the application itself
- being aware of an application running on a particular host/service can help in associating a new flow to it, and to identify the related activity

Experimental Analysis: a Proof of Concept

To validate our methodology we applied it to Skype

- multi-channel application
- widespread
- hybrid super-peer based P2P architecture
- intensive use of encryption (AES + RSA)
- proprietary protocols and algorithms



We used TIE [2] to gain knowledge of the traffic associated to each Skype communication channel (see Tab. 1), and we discovered several patterns at different layers

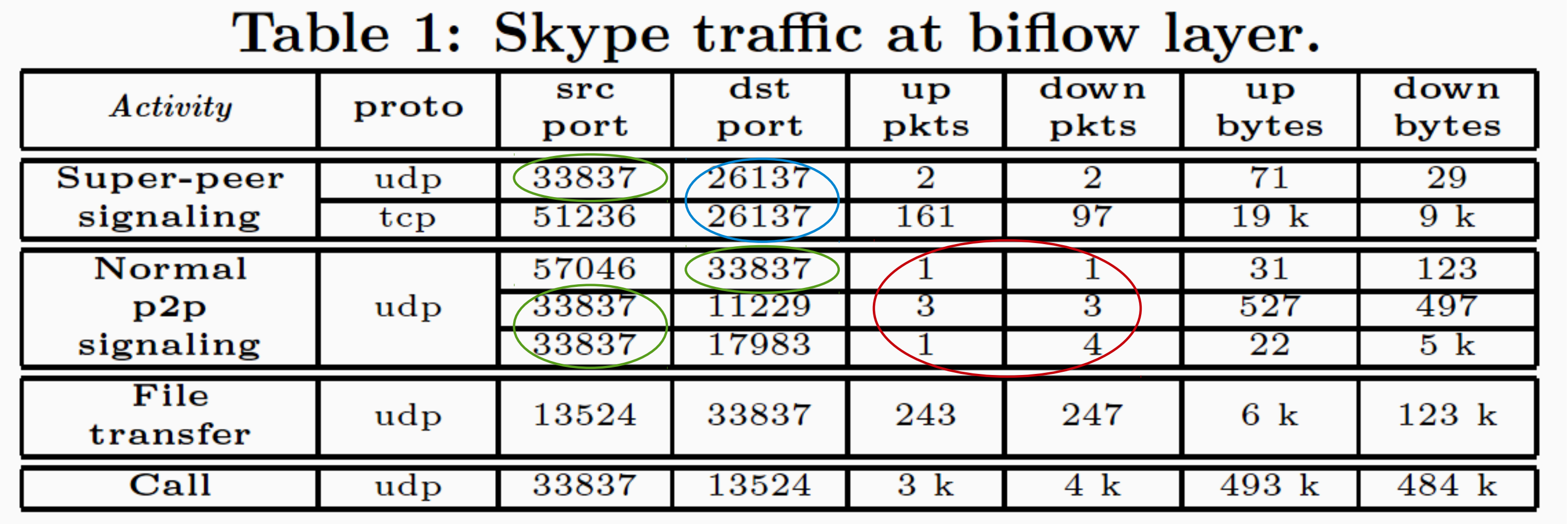


Table 1: Skype traffic at biflow layer.

Activity	proto	src port	dst port	up pkts	down pkts	up bytes	down bytes
Super-peer signaling	udp	33837	26137	2	2	71	29
	tcp	51236	26137	161	97	19 k	9 k
Normal p2p signaling	udp	57046	33837	1	1	31	123
		33837	11229	3	3	527	497
		33837	17983	1	4	22	5 k
File transfer	udp	13524	33837	243	247	6 k	123 k
Call	udp	33837	13524	3 k	4 k	493 k	484 k

Host layer

- port numbers are used more than once on a short period

→ listening services (i.e. port 33837)

Service layer

- same TCP and UDP listening port number

→ detect also TCP communications

Biflow layer

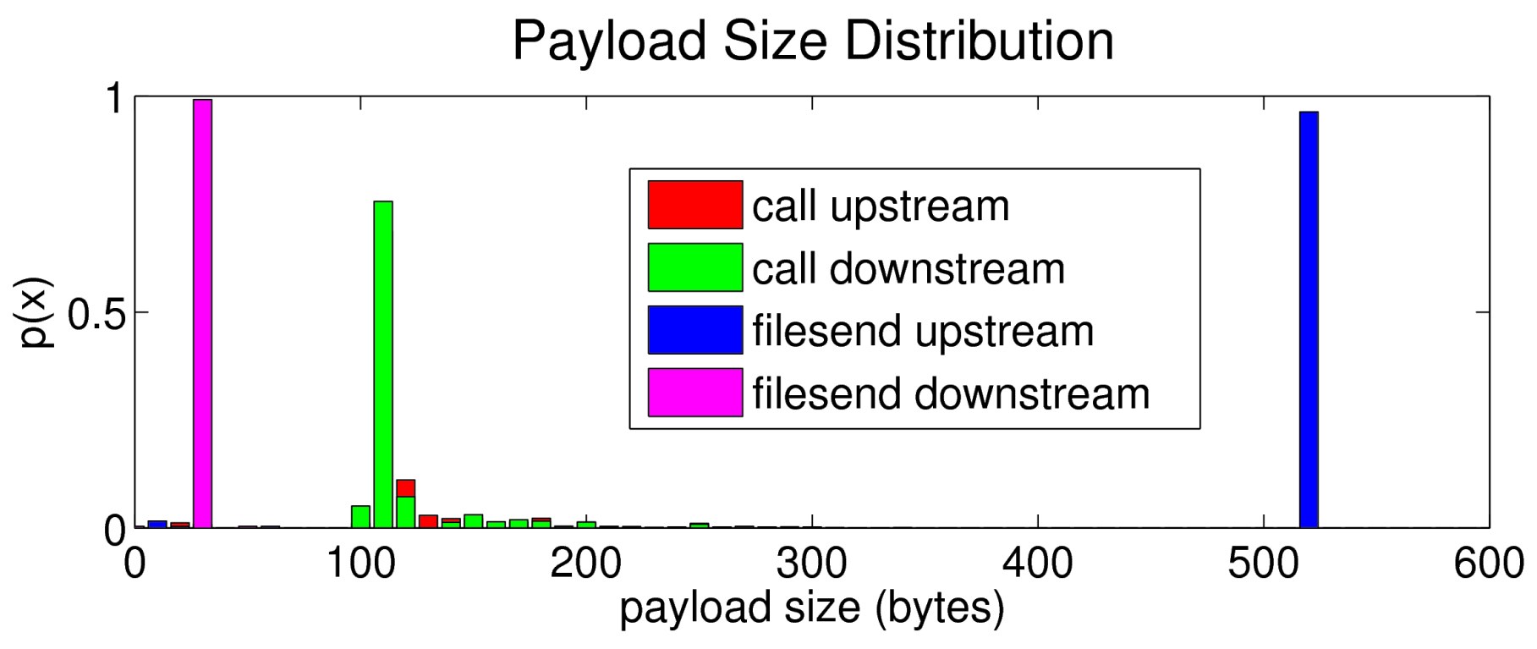
- *signaling traffic* is mostly composed by many short UDP biflows revealing a few different patterns
- *file transfers* present almost the same number of packets in both directions, but most bytes fall only in one of them
- *voice calls* reveal a symmetric pattern in transferred data

→ patterns peculiar to activities

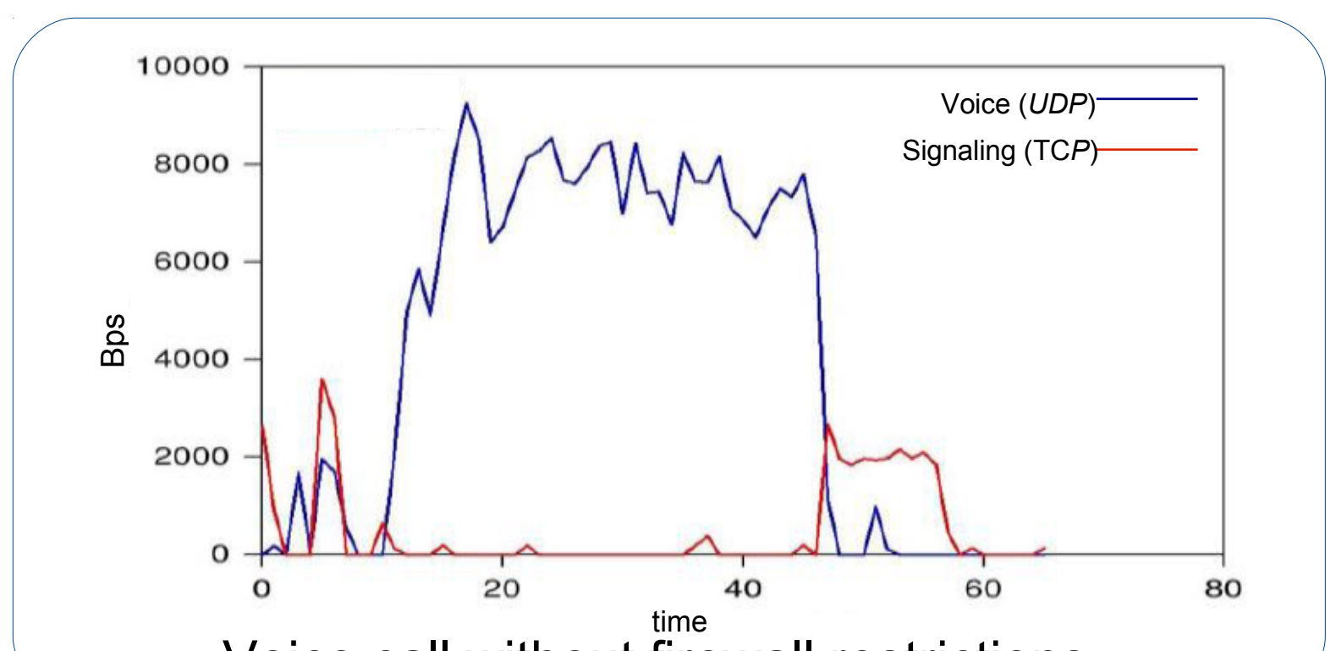
Packet layer

- activities reveal peculiar payload-sizes distribution patterns

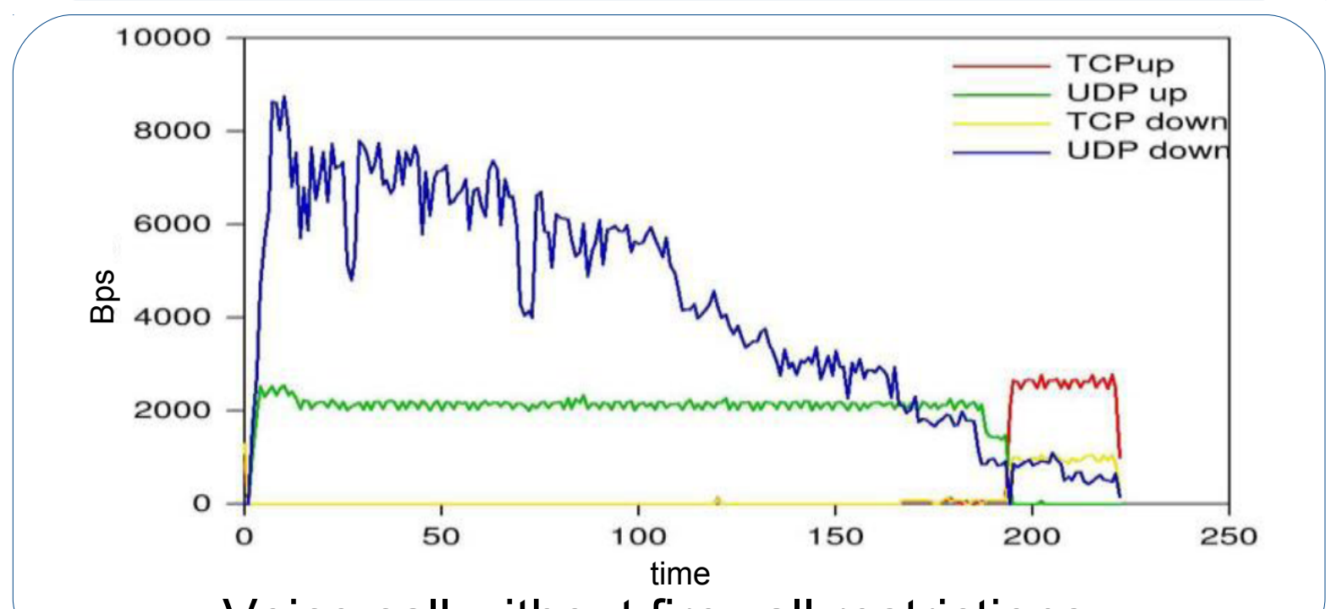
→ more patterns peculiar to activities



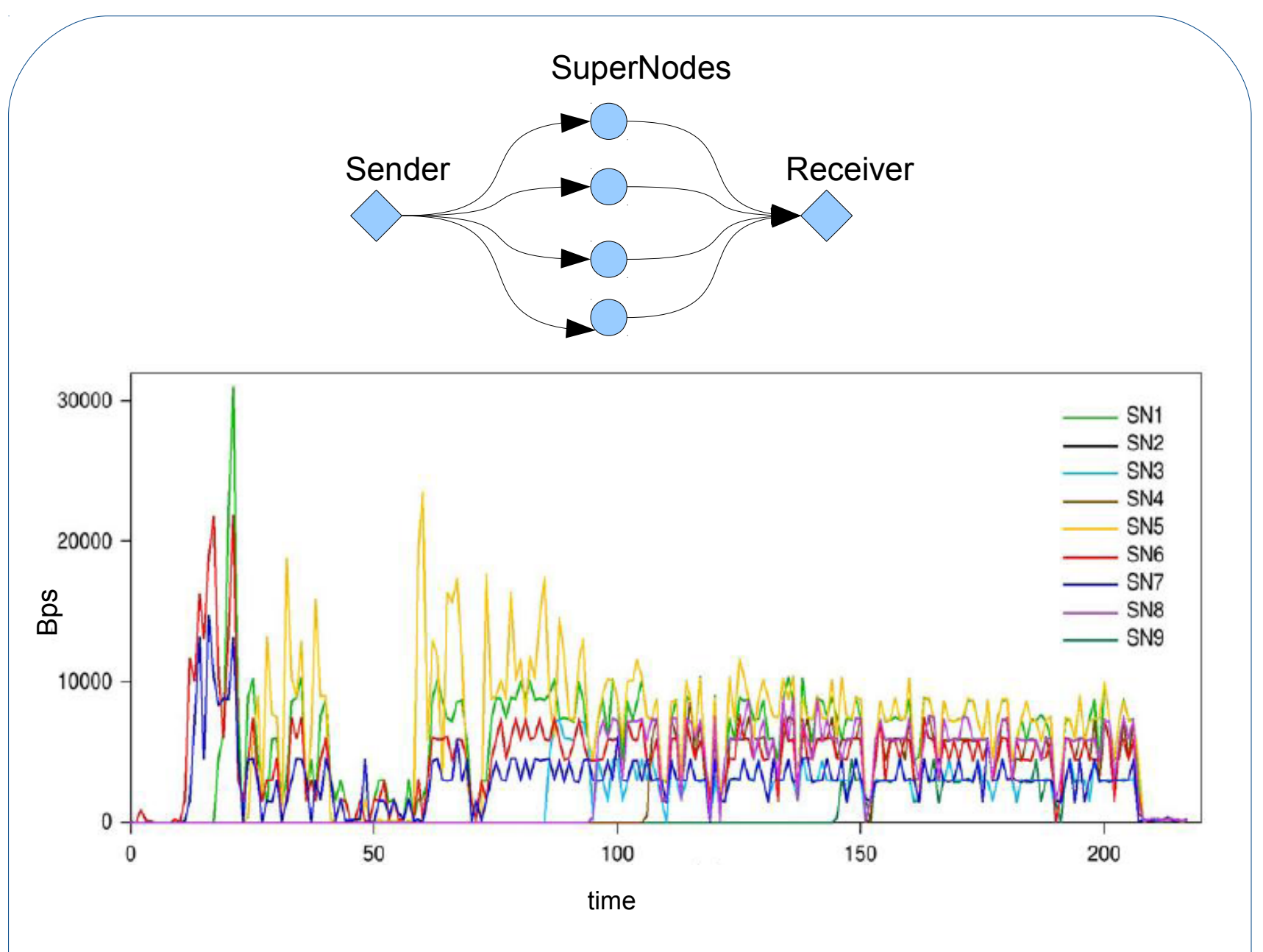
Combining the previous observations allows to identify Skype and its activities. For instance, once inferred Skype random port number, we were able to detect different communication channels involved in the same activity.



Voice call without firewall restrictions



Voice call without firewall restrictions



Distributed file transfer
(both clients behind a NAT)

References

[1] W. Li, M. Canini, A. W. Moore, and R. Bolla. Efficient application identification and the temporal and spatial stability of classification schema. Elsevier Computer Networks., 53(6):790–809, 2009.

[2] A. Dainotti, W. de Donato, and A. Pescapé. TIE: A Community-Oriented Traffic Classification Platform. In Proceedings of the First International Workshop on Traffic Monitoring and Analysis, Berlin, Heidelberg, 2009.