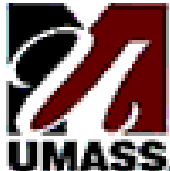


# Characterizing and Detecting Skype-Relayed Traffic

Kyoungwon Suh, Daniel R. Figueiredo,  
Jim Kurose, and Don Towsley  
presented by Kyoungwon Suh

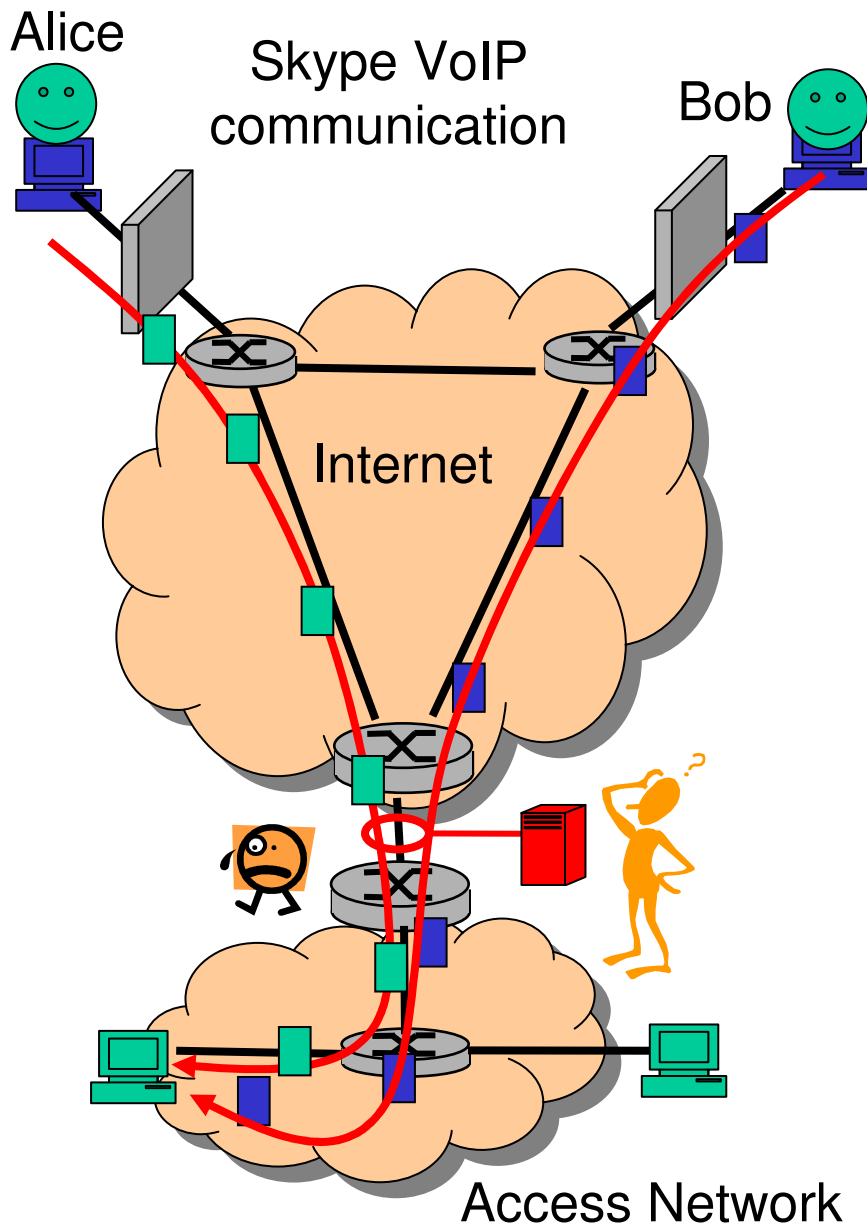


University of Massachusetts  
Department of Computer Science

# Outline

- ❑ Motivation
- ❑ Previous work
- ❑ Characterization of Skype-relays
- ❑ Four statistical metrics for detection
- ❑ Evaluation of detection accuracy
- ❑ Summary, future work

# Motivation



- ❑ Skype uses relay nodes
  - ❖ NATed/Firewalled hosts
  - ❖ Detouring routing
- ❑ Why need to detect?
  - ❖ Network resource wastage
  - ❖ Security hole
- ❑ Hard to detect at access link
  - ❖ Match incoming with outgoing traffic
  - ❖ Aggregate traffic
  - ❖ Proprietary protocol
  - ❖ Encryption
- ❑ Why focus on "Skype" relays?
  - ❖ Most prevalent; case study

# Related work on traffic-source identification

## ❑ Stepping-stone detection

- ❖ mainly dealing with *interactive* applications (Zhang00, Dohono02, Blum04)

## ❑ Flow correlation in MIX networks

- ❖ focusing on single host's traffic; non-streaming traffic (Levine04, Zhu04)

## ❑ DoS attack source identification

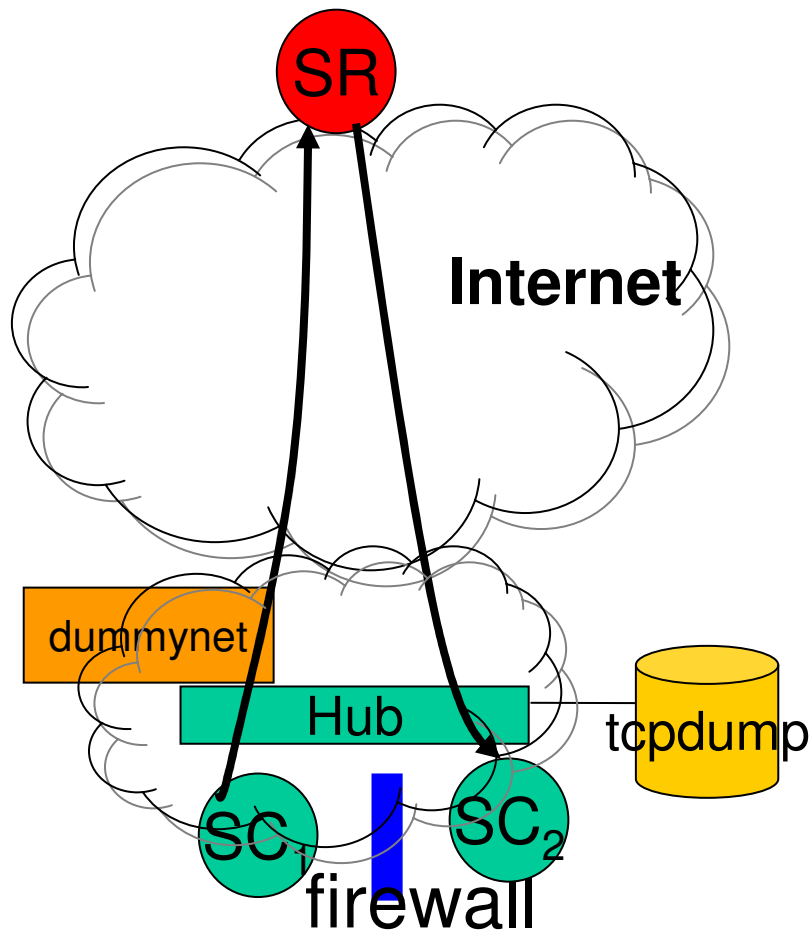
- ❖ use of host-specific information (Sekar04)

## ❑ P2P traffic identification

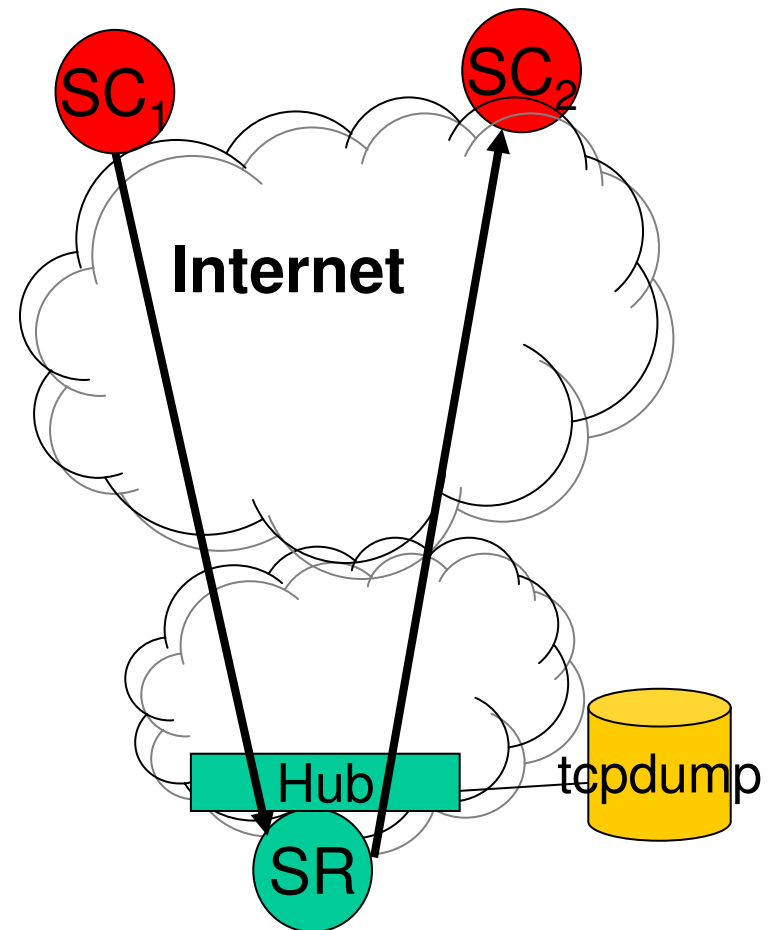
- ❖ use of application or protocol-specific information (Karagiannis05, Moore05)

No previous work on detection schemes for multimedia *streaming relayed* traffic

# Two Controlled experiments



Load: 1000 skype calls  
(serially)

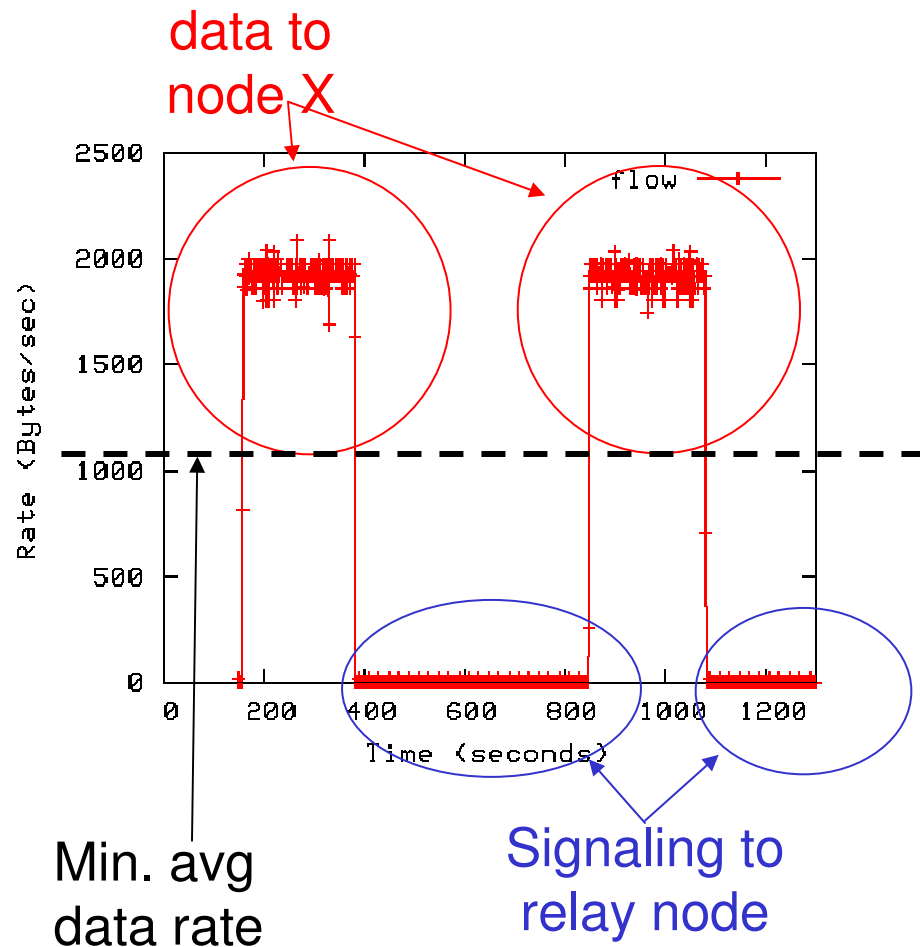


“honeypot”: hundreds of  
Skype relayed traffic (serially)

# Initial insights from controlled experiments

- ❑ Traffic usually bi-directional
- ❑ Many protocol transformations
  - ❖ UDP->UDP; UDP->TCP; TCP->UDP; TCP->TCP
  - ❖ fragmentation/defragmentation: TCP tends to use larger payload size than UDP (2 UDP packets -> 1 TCP packet)
- ❑ Bitrate: multiple codecs
  - ❖ minimum bit rate
  - ❖ max bit rate varies with CODEC
- ❑ Signaling traffic not relayed
  - ❖ traffic sent to relay but not to destination
  - ❖ need to separate data/signaling in relay study
- ❑ "Similar" volume between incoming and relayed traffic
  - ❖ no complex app-level transcoding

# Traffic Bursts in Skype

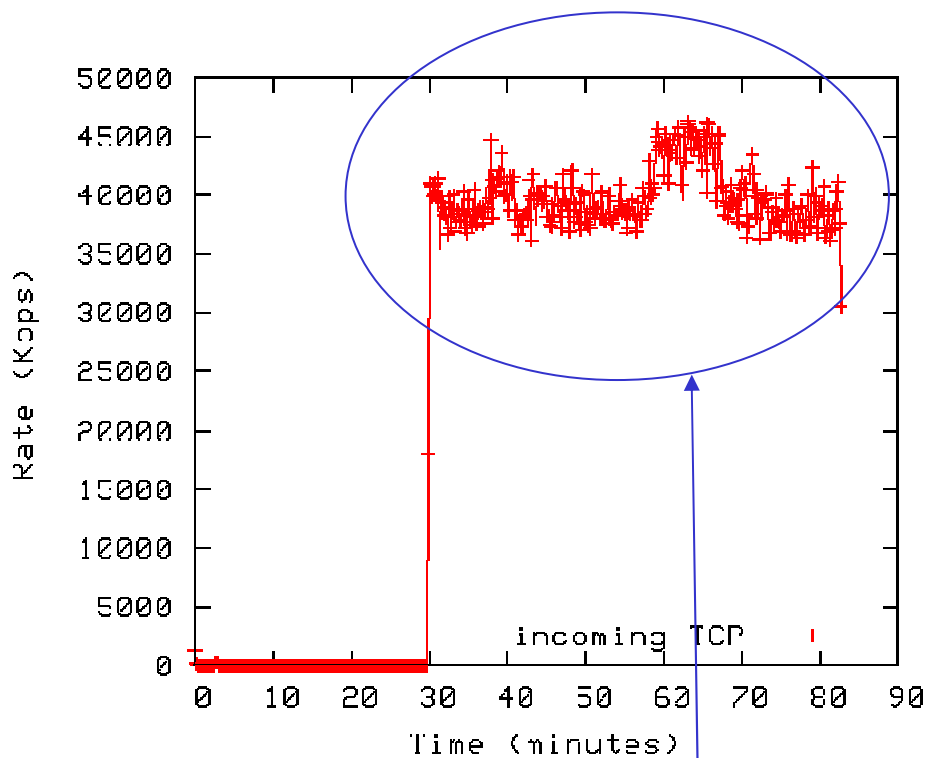


Min. avg  
data rate

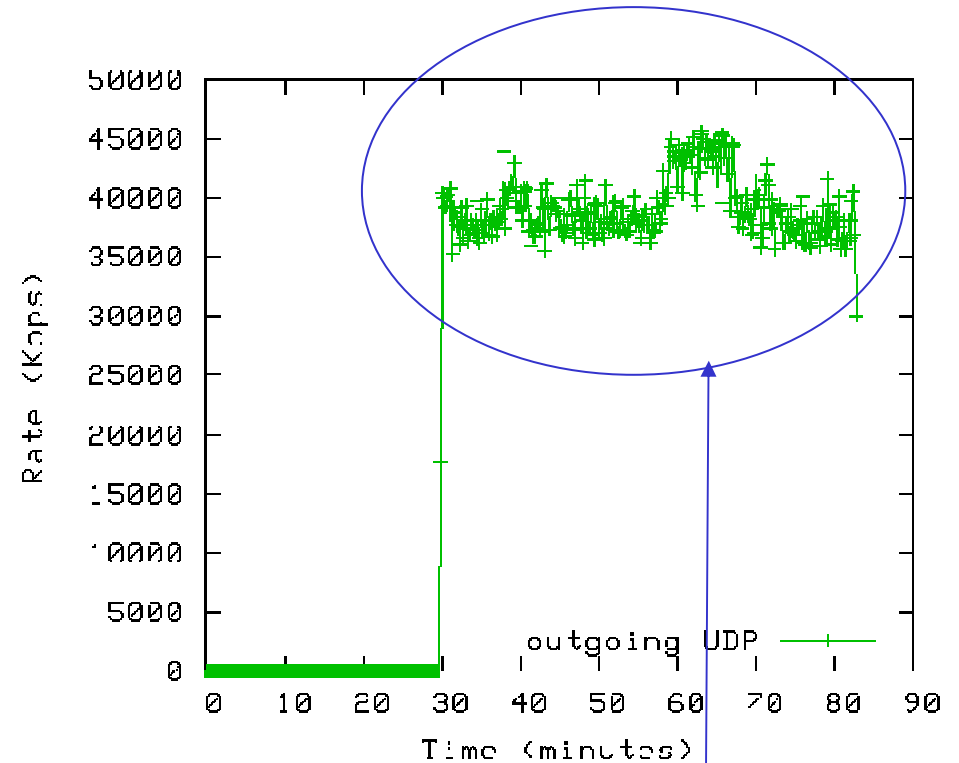
incoming flow: time series

- ❑ Burst: contiguous piece of flow
  - ❖ 5 tuple ; uni-directional
    - Src/dst addr
    - Protocol
    - Src/dst ports
  - ❖ min. avg data rate: 10kbps
  - ❖ min. duration: 30secs
- ❑ Change detection algorithm
  - ❖ EWMA: track data rate
- ❑ Using bursts for detection
  - ❖ identifies multiple voice calls within same flow
  - ❖ separates signaling traffic from data traffic
  - ❖ facilitates detection of relayed traffic

# Example of incoming and outgoing bursts (relayed)

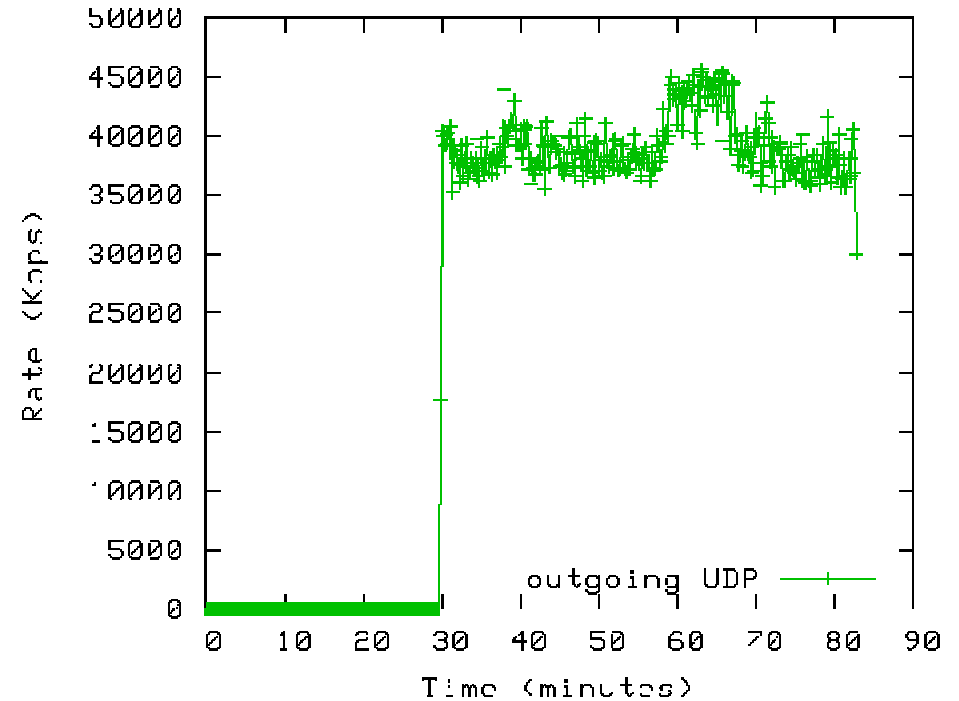
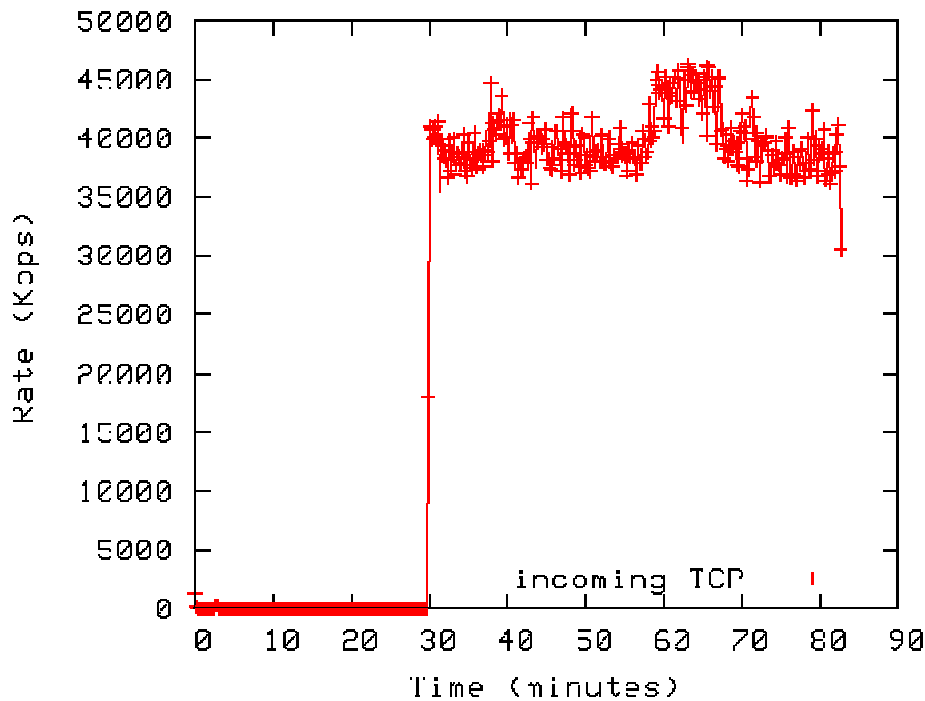


Data to  
relay node

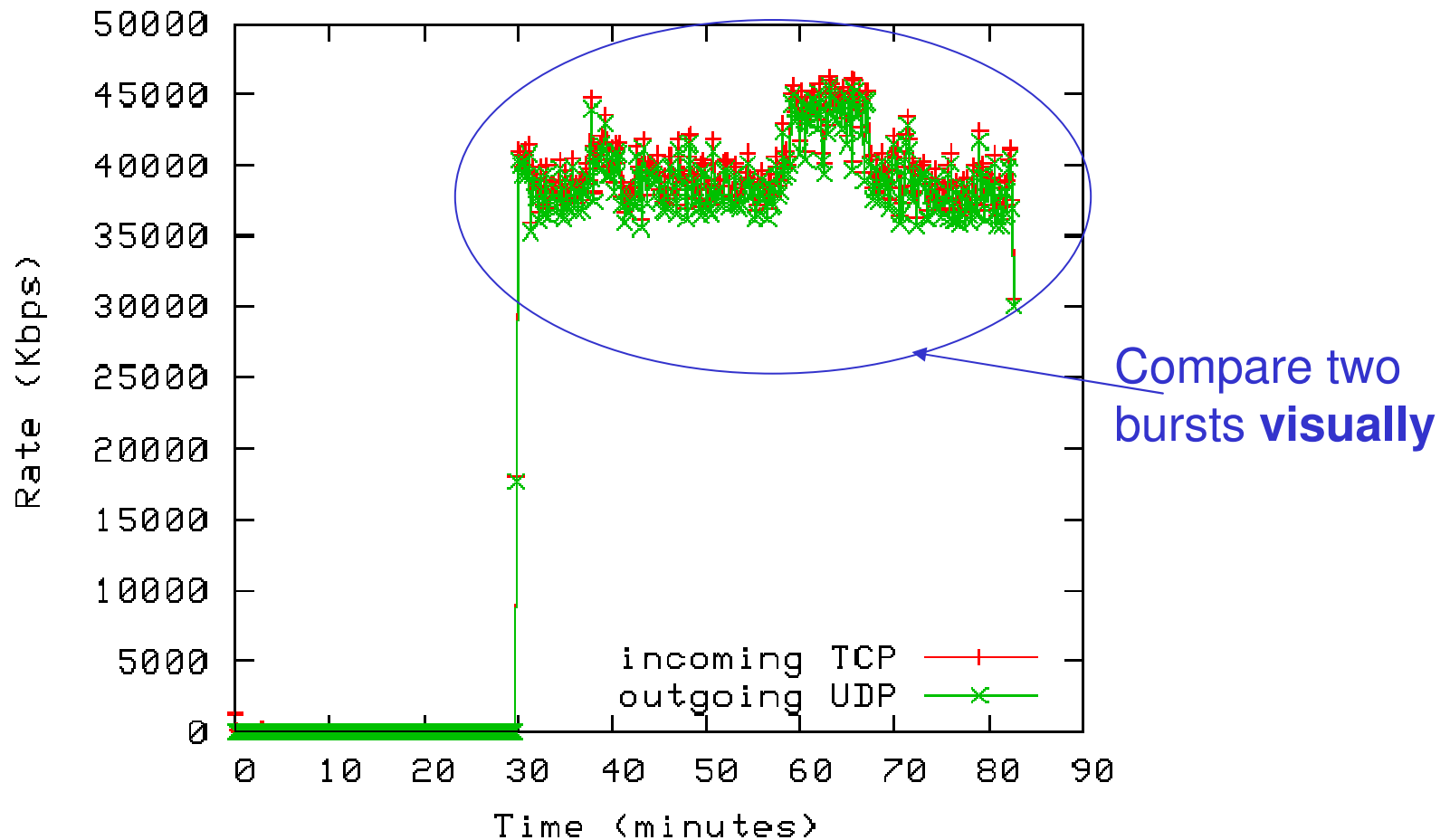


Data from  
relay node

# Example of incoming and outgoing bursts (relayed)



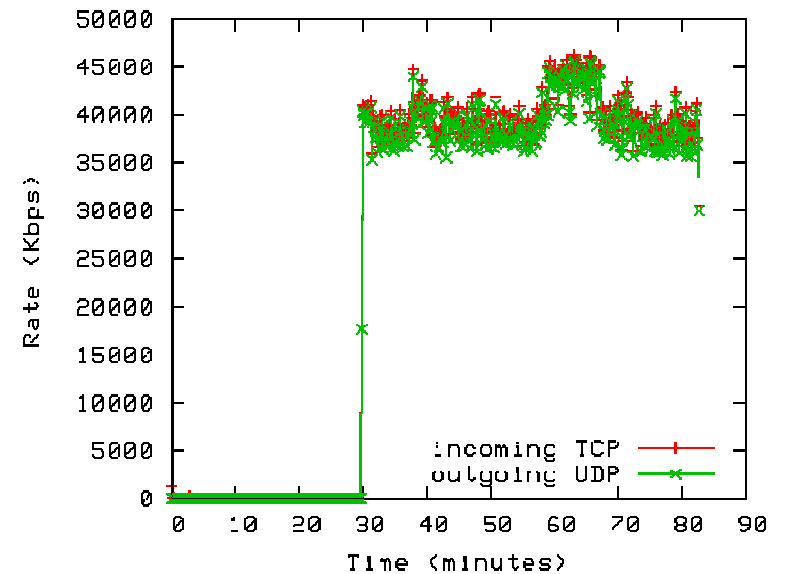
# Example of incoming and outgoing bursts (relayed)



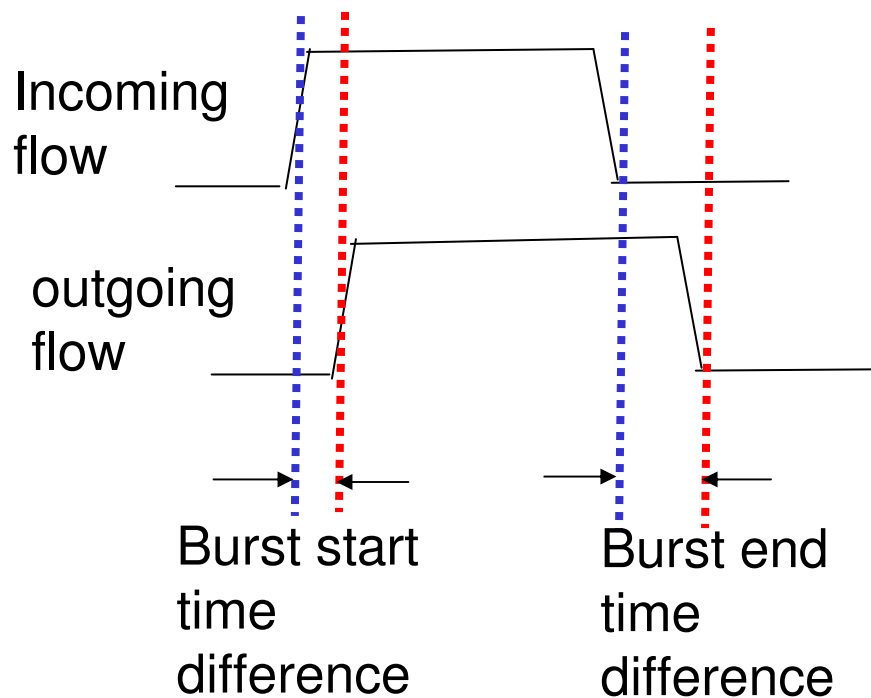
Compare incoming burst against outgoing burst!

# Skype relay detection - Main idea

- compare "input burst" against a set of "output bursts", to identify input-output relay pairs
- no application-specific information
  - ❖ use only traffic characteristics



# Four statistical metrics for Skype-relay detection



## □ Metrics

- ❖ Burst start time diff
- ❖ Burst end time diff
- ❖ Burst size ratio
- ❖ Maximum cross correlation
  - between time series of binned byte counts

# Other metrics (tried unsuccessfully)

❑ **packet count**, doesn't work well because..

- ❖ Protocol change

- Example: incoming TCP-outgoing UDP
- Use larger payload size for TCP

- ❖ Fragmentation and defragmentation

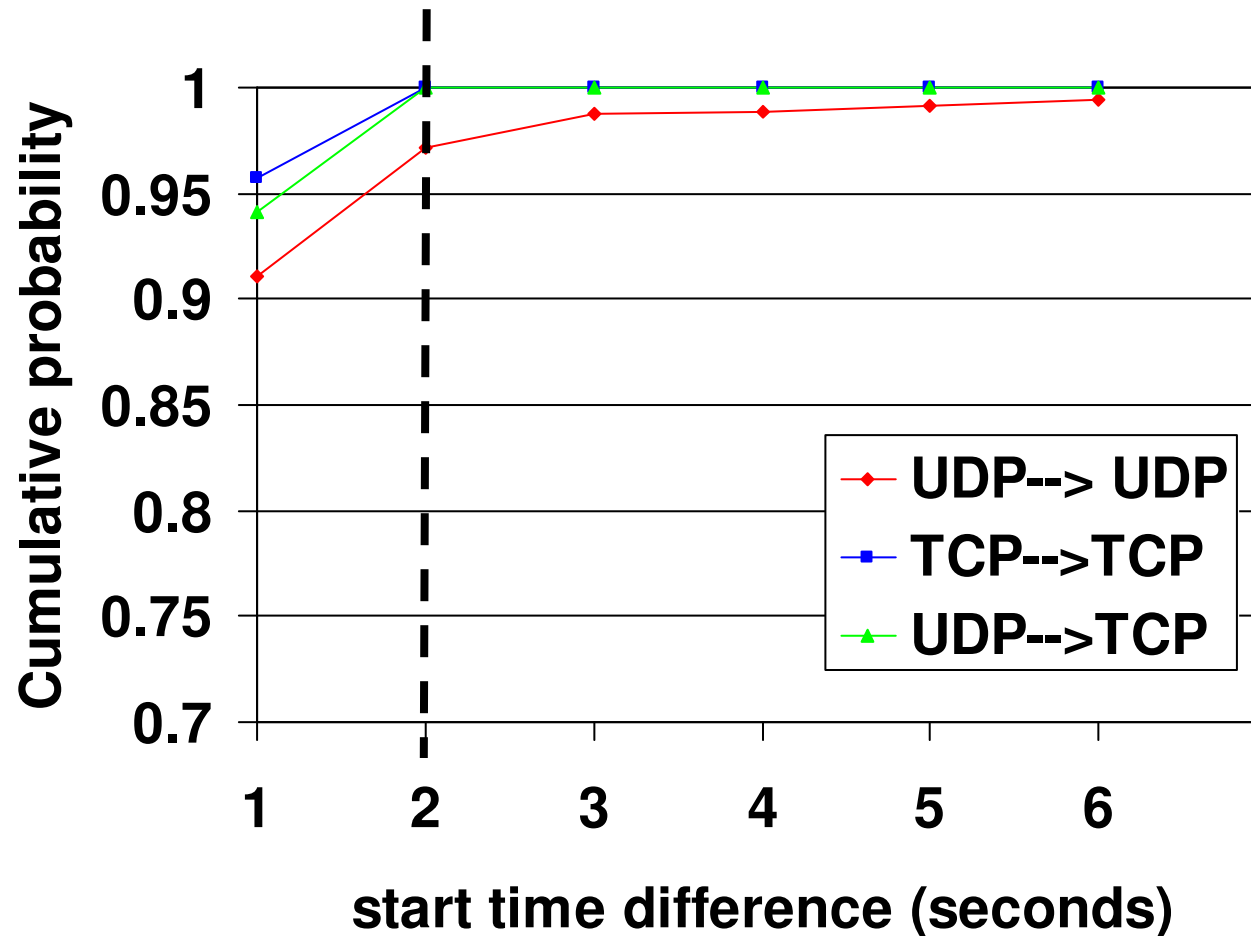
- ❖ TCP retransmission

❑ **Byte count**, doesn't work well because..

- ❖ Different App-level header size (unknown)

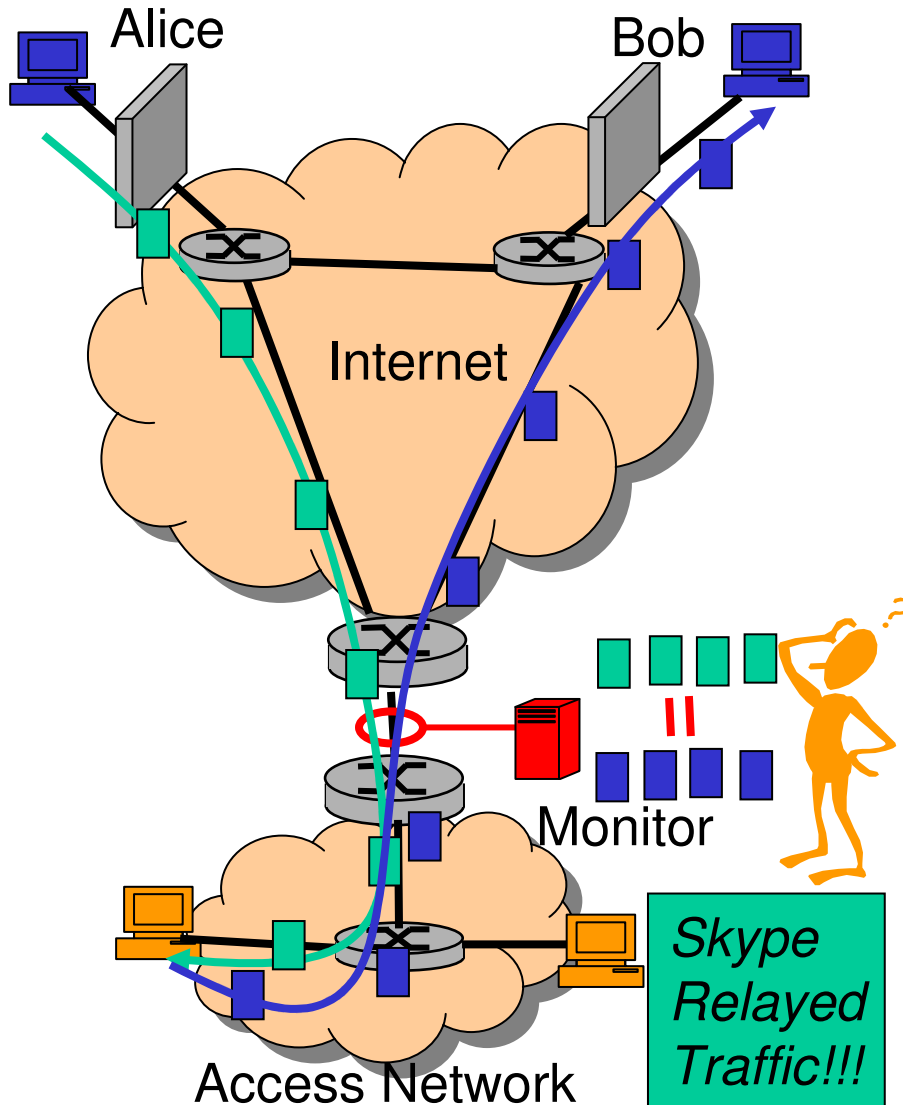
- ❖ TCP retransmission

# Empirical cumulative distribution of burst start time difference



97% of relays have less than 2 seconds as start time difference

# Trace collection setup for experiment

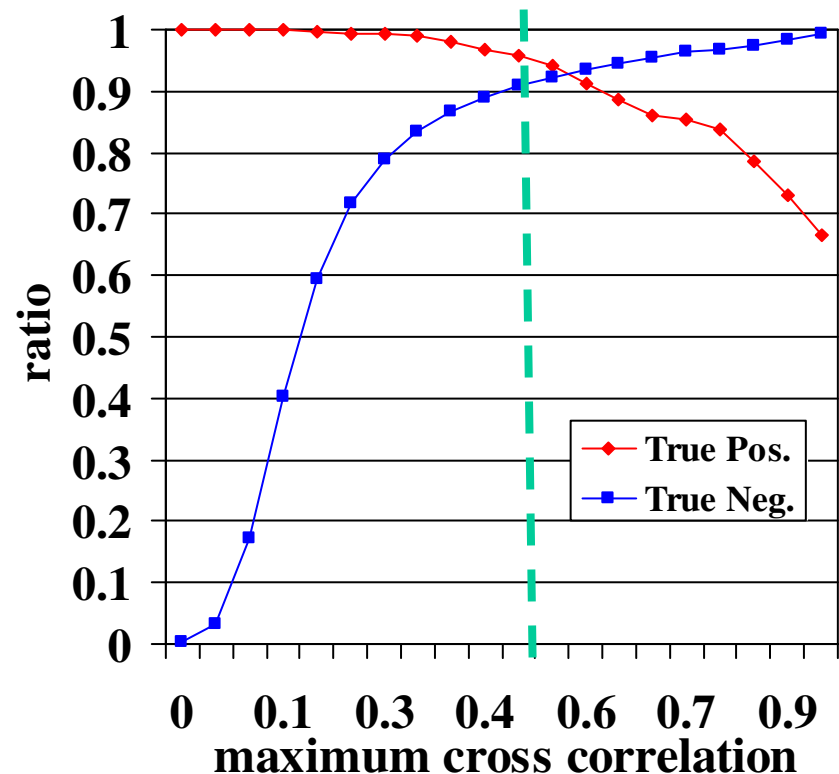
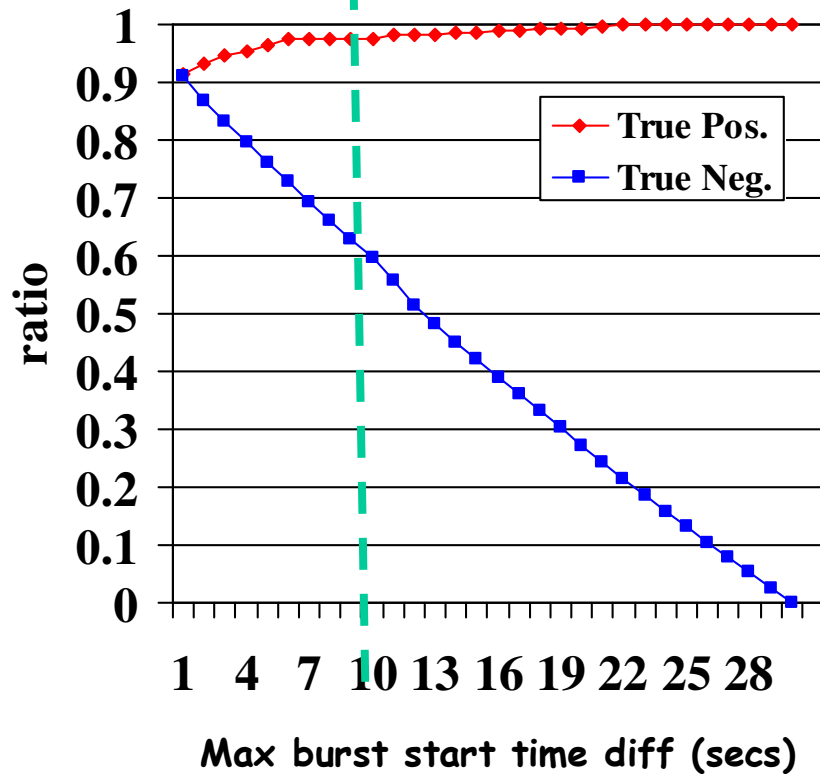


- ❑ **Passive traffic Monitor**
  - ❖ Bidirectional Gigabit link
  - ❖ IP header collection
    - No IP payload for privacy
  - ❖ 17 hours; 414G bytes
- ❑ **Traffic analysis tool**
  - ❖ Flow analysis
  - ❖ Burst detection
  - ❖ Skype relay detection
    - Statistical correlation between two bursts

# Evaluation methodology

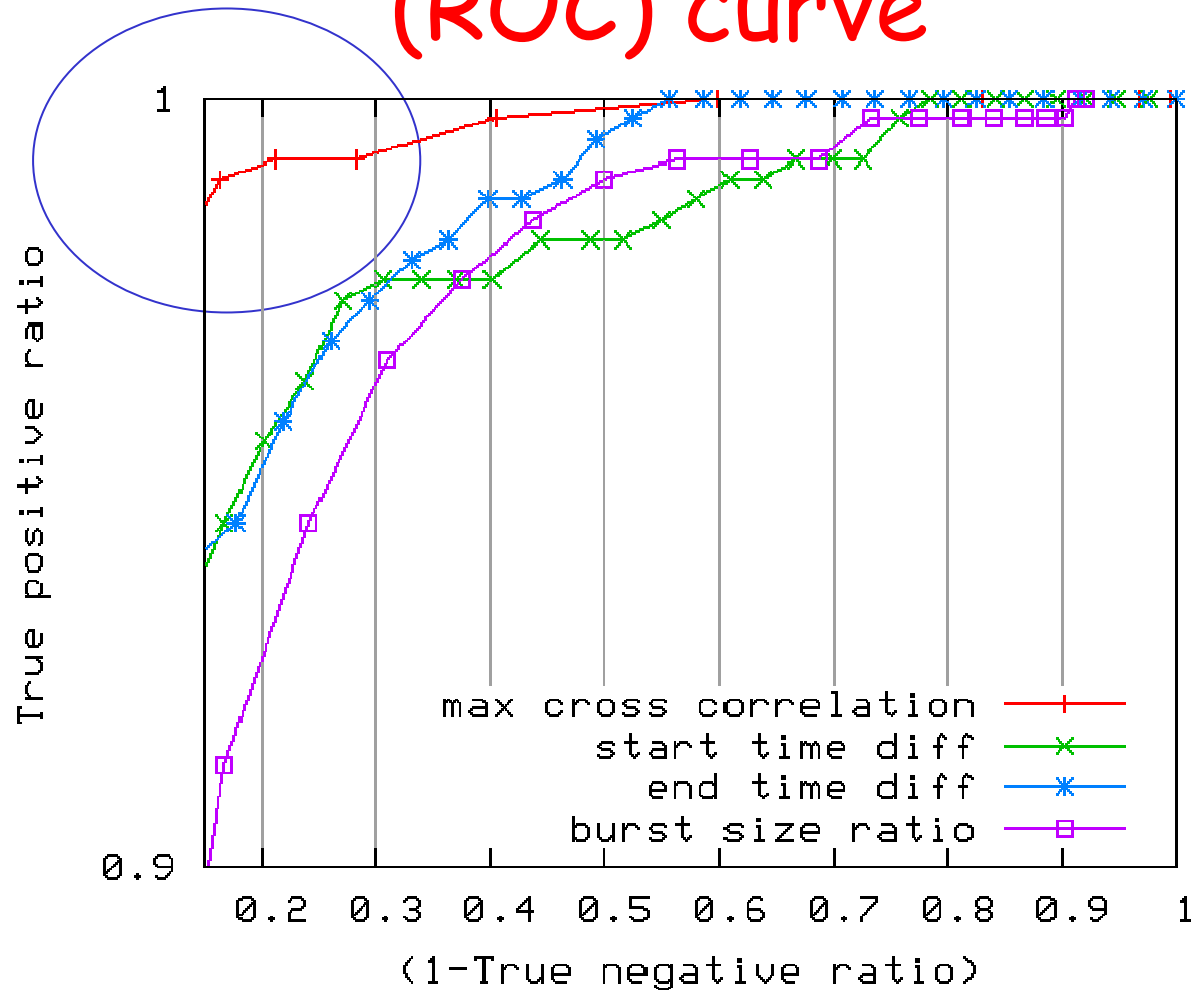
- Definitions of true pos. and false pos. ratios
  - ❖ True positive
    - $(\text{Num. of true skype relays classified as skype relays}) / (\text{num. of true skype relays})$
  - ❖ True negative
    - $(\text{Num. of false skype relays classified as Skype relays}) / \text{num. of false skype relays}$
- Use of threshold as a classifier parameter
  - ❖ Example: taking 2 sec as threshold for **max** start time difference, 97% taken into true positive

# Start time difference / max cross correlation



Varying threshold changes TP and TN

# Receiver operator characteristic (ROC) curve



By combining different metrics can improve accuracy

- .96 TP and .96 TN

# Summary and future work

- ❑ Characterization of Skype-relayed traffic using controlled experiments
- ❑ Methodology to detect Skype-relayed traffic
- ❑ Empirical validation using aggregate traffic from large network
- ❑ Future and on-going work
  - ❖ Handling skewed cumulative byte difference
  - ❖ Characterizing more complex relays (1 flow in;  $k$  flows out)

The End