



# Identifying Traffic Differentiation on Cellular Data Networks

Abbas Razaghpanah<sup>†</sup>, Arash Molavi Kakhki<sup>§</sup>, Rajesh Golani<sup>†</sup>, David Choffnes<sup>§</sup>, Phillipa Gill<sup>†</sup>, Alan Mislove<sup>§</sup>  
Northeastern University<sup>§</sup>, Stony Brook University<sup>†</sup>

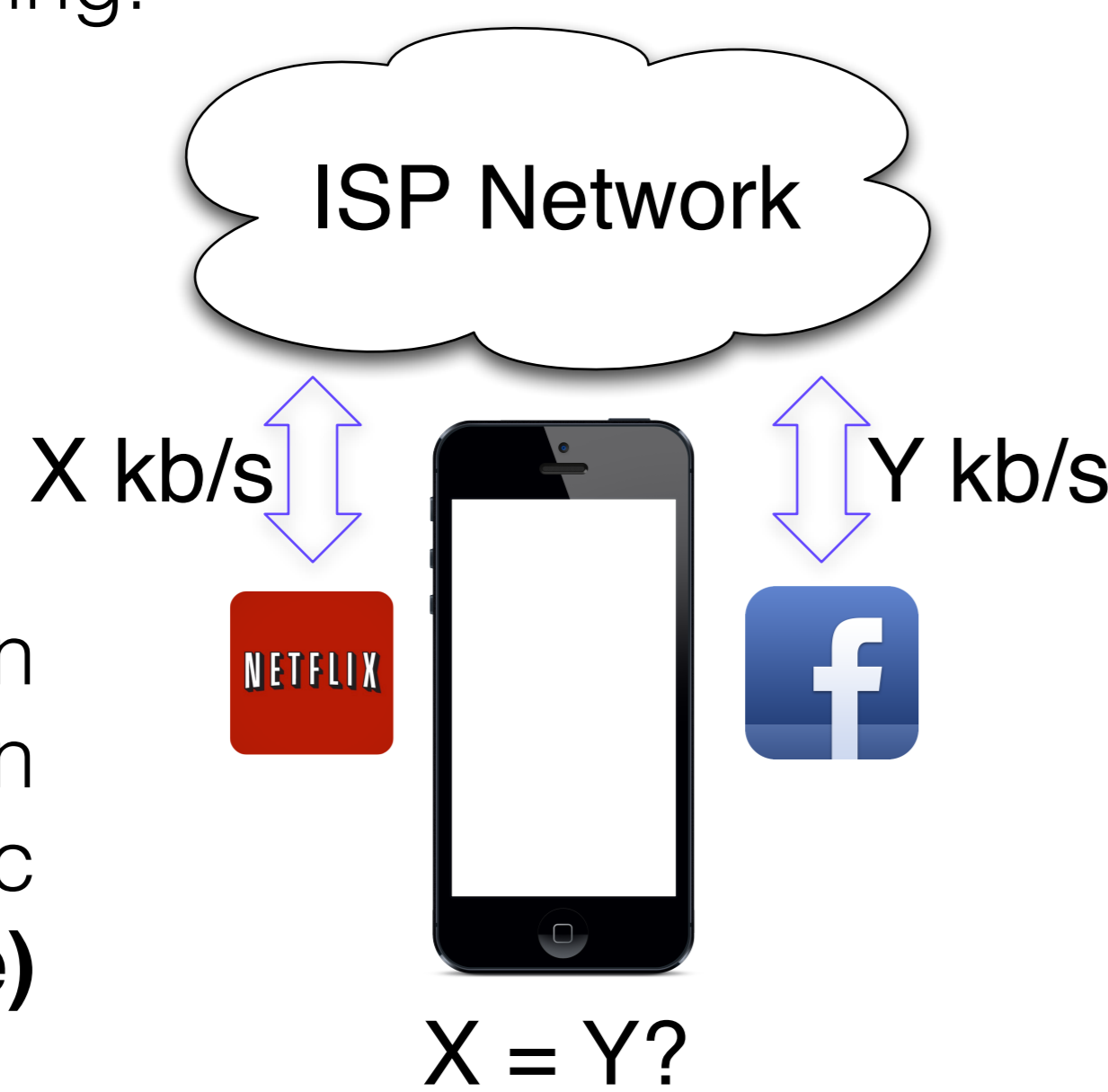


## Introduction

**Traffic differentiation:** any attempt to change the performance of network traffic traversing an ISP's boundaries.

ISPs may implement differentiation policies for a number of **reasons**, including:

- load balancing
- bandwidth management
- business reasons.



Specifically, we focus on detecting whether certain types of network traffic **receive better (or worse) performance.**

## Related work

Previous work explored this problem in limited environments.

	Switzerland	Glasnost	Us
Applications Tested	BitTorrent Only	P2P and Video	Any Application
Desktop App	Yes	Browser Plugin	Yes
Customized Tests	No	No	Yes
Smartphone App	No	No	Yes

## Objectives

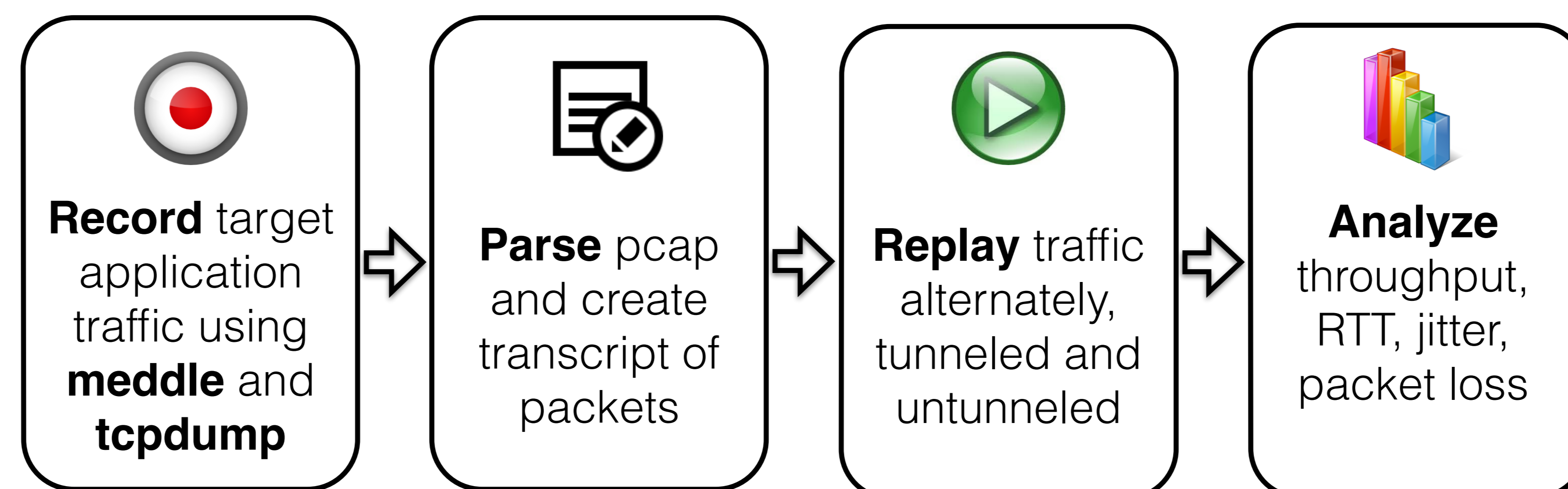
1. **Identify and expose** differentiation
2. **Measure** how differentiation affects performance  
Delaying, throughput throttling, jitter, packet dropping
3. **Classify** differentiation for popular applications  
VoIP, Media Streaming, File Sharing, Cloud Storage



## Assumptions

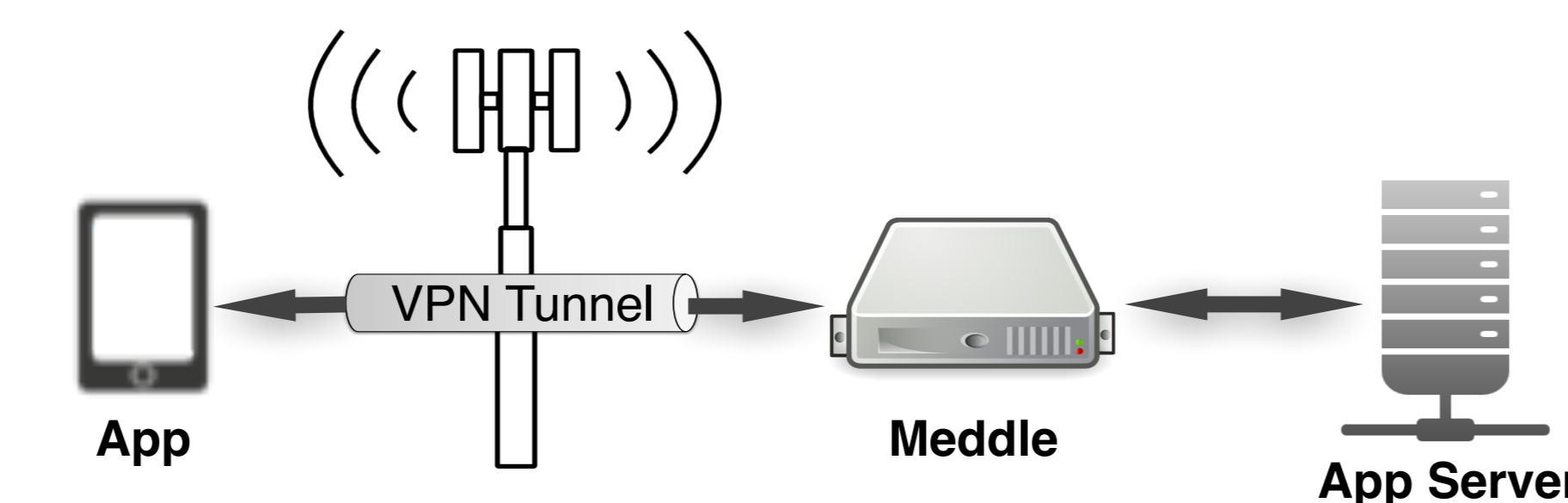
We assume that ISPs will differentiate traffic based on properties such as **hostname**, **IP addresses**, **ports**, total **number of connections**, **payload signatures**, total **bandwidth** and **time of day.**

## Methodology and challenges

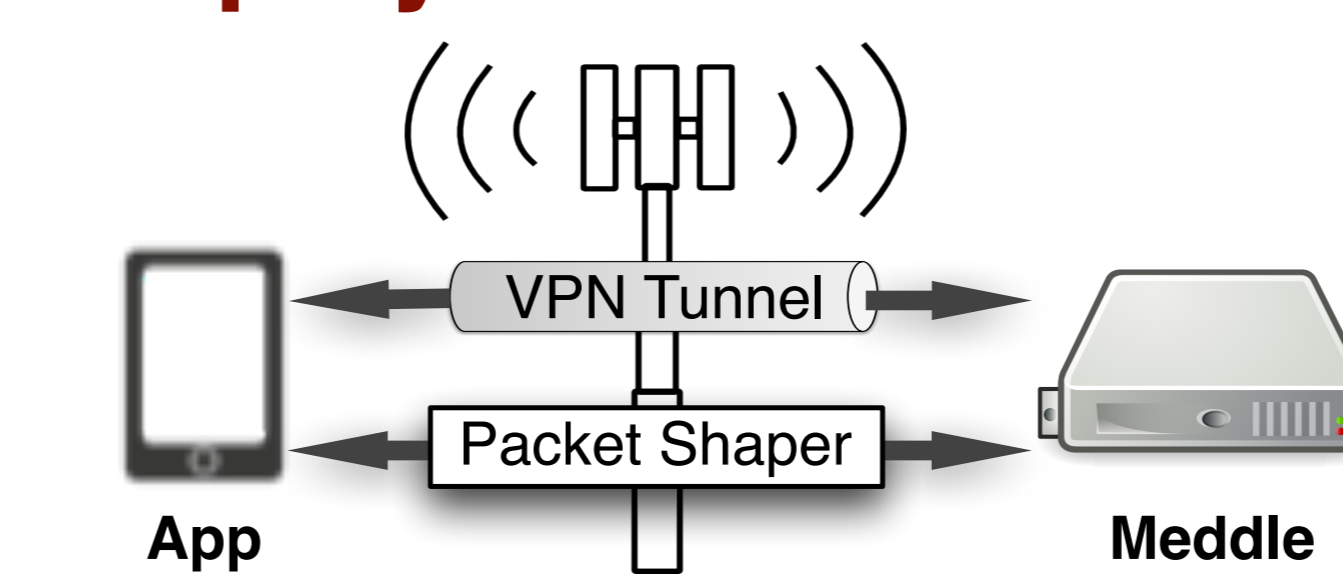


- How to **capture and replay** the salient features of application traffic such that it will be subject to differentiation from middleboxes?
- How to establish **ground truth**?
- How to **quantify** differentiation?
- Implementation challenges: e.g. users behind **NAT**, **tcpdump** on client side, **noise and unrelated traffic.**

### Record:

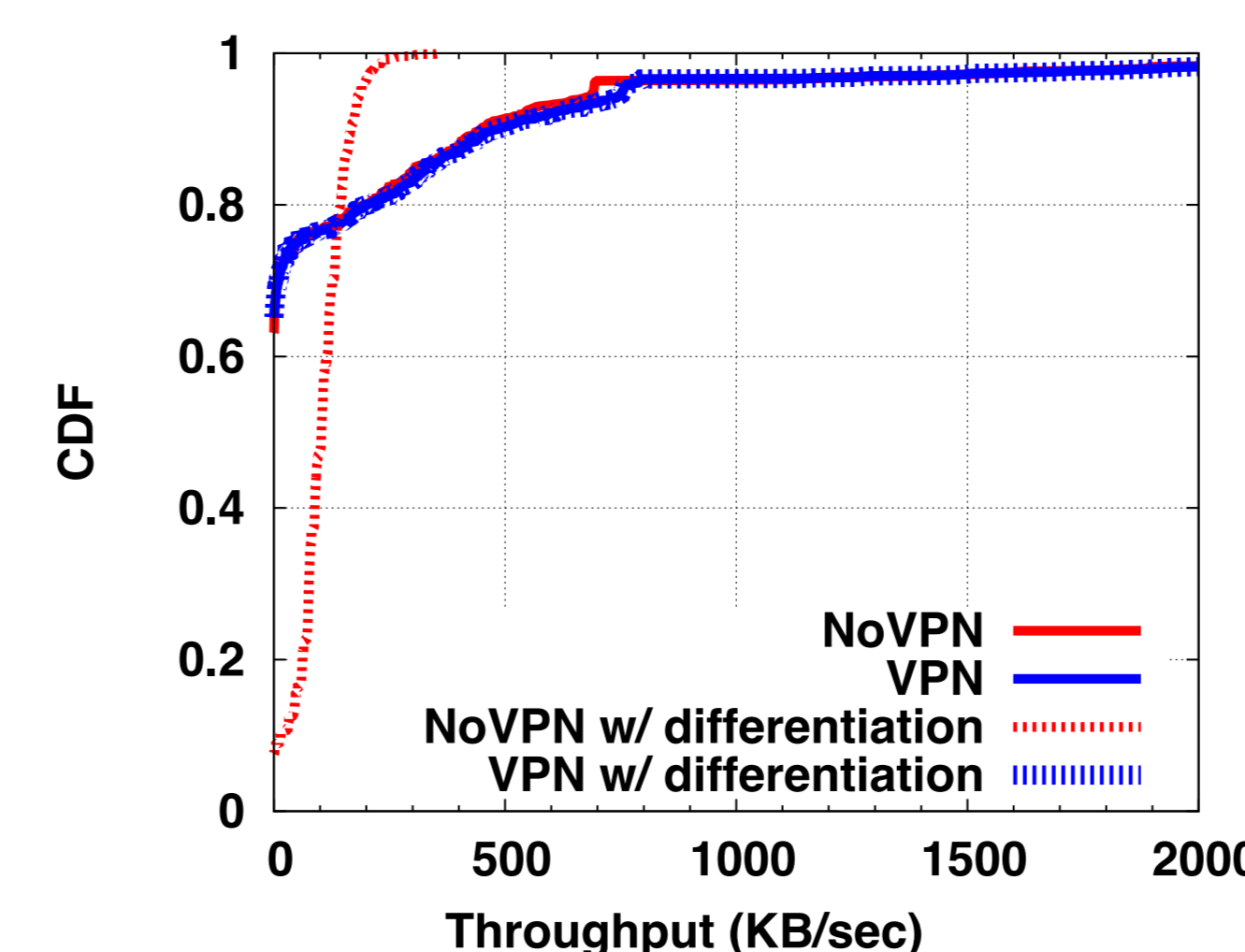
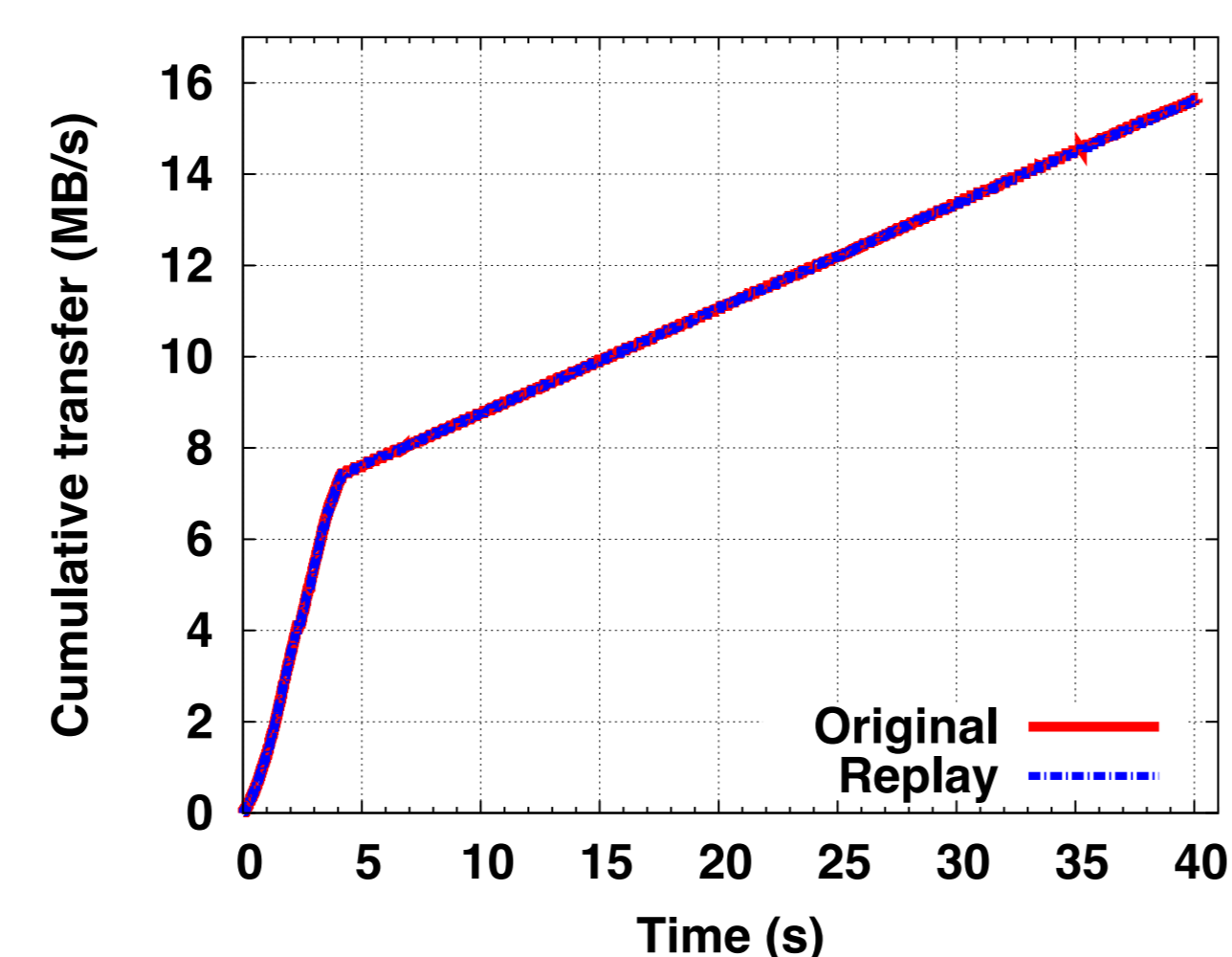


### Replay:



## Feasibility

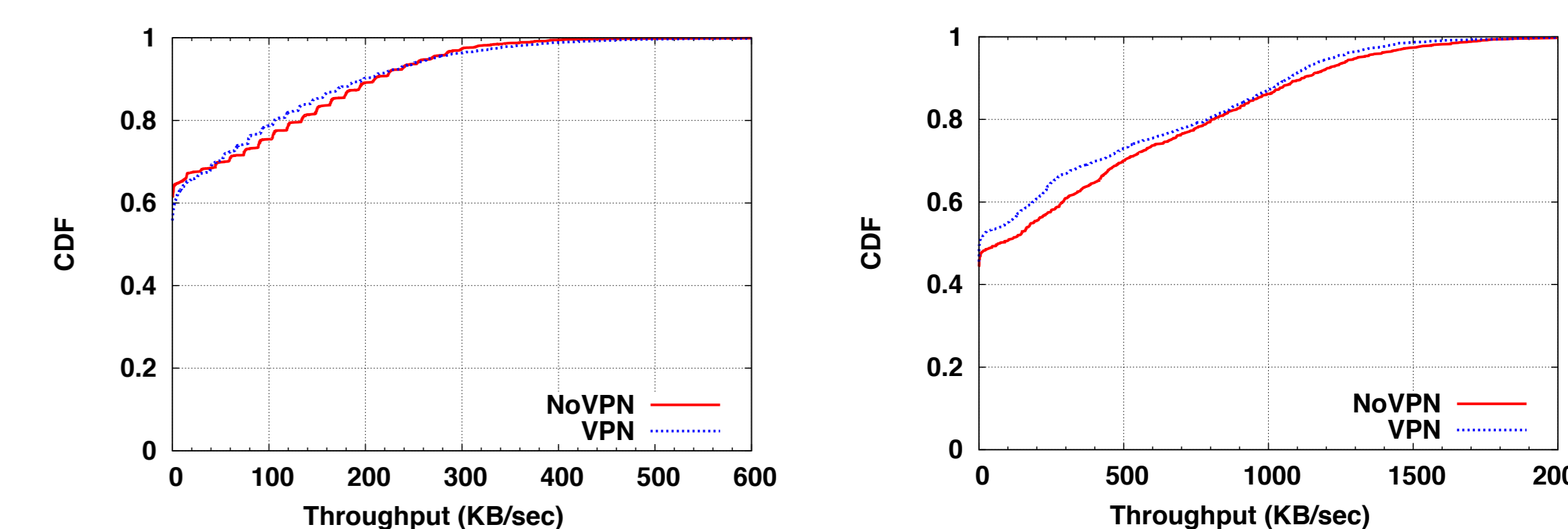
- Our replay produces traffic **nearly identical** to the original traffic (YouTube traffic recorded on Verizon).
- **Detects differentiation** in a controlled environment (%3 packet loss and 10ms delay).



## Initial results

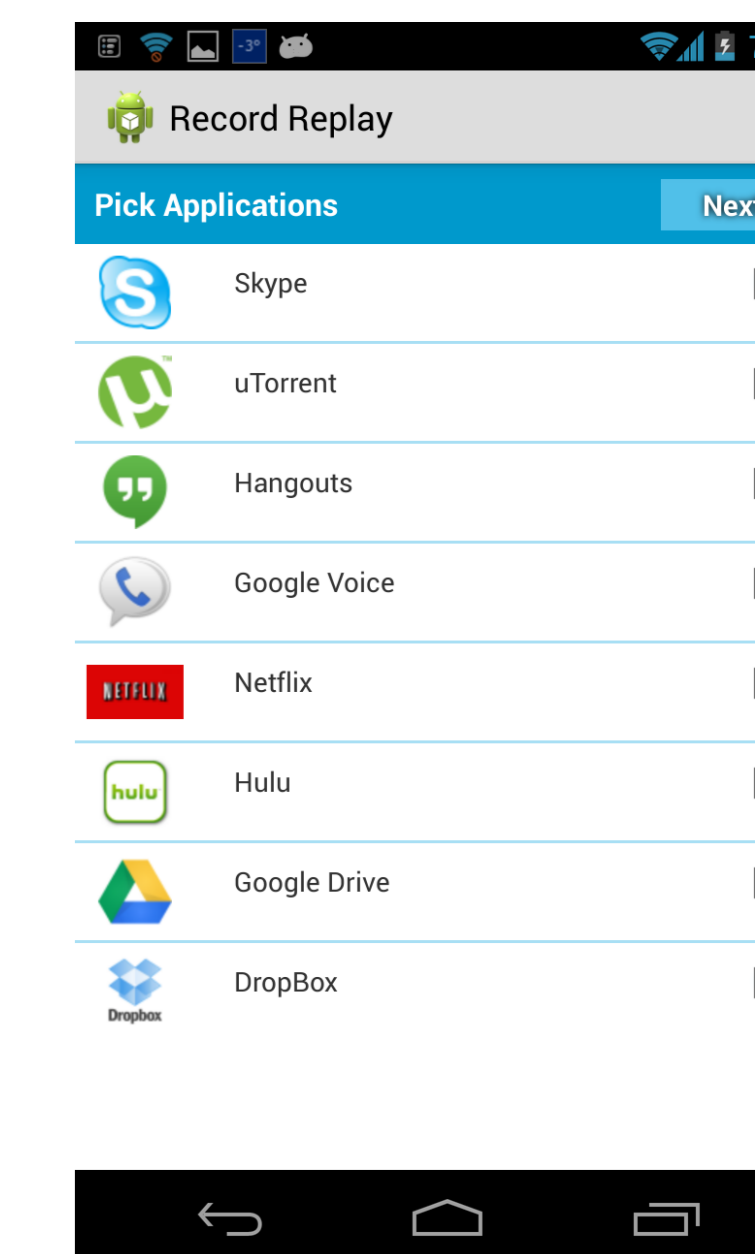
Currently tests are performed by researchers using PCs tethered to mobile devices.

Spotify (left) and Netflix (left) Verizon in Boston. **No differentiation detected.**



## Future work

- Create a **mobile app** to enable wider adoption of the test (in progress).  
Run on networks we know there is differentiation to **validate** our techniques.
- Allow users to **record and replay custom traces** (longer term).
- Create a **Differentiation Watch** website and blog to report the results for different networks and blog about newly detected differentiation.
- **Source-spoofing** to detect destination IP-based differentiation.



## References

- [1] <http://www.meddle.mobi>
- [2] <https://www.eff.org/pages/switzerland-network-testing-tool>
- [3] M. Dischinger, M. Marcon, S. Guha, K.P. Gummadi, R. Mahajan, S. Saroiu, Glasnost: enabling end users to detect traffic differentiation, in: USENIX NSDI'10, 2010