Identifying Traffic Differentiation on Cellular Data Networks

Abbas Razaghpanah§, Arash Molavi Kakhki‡, Rajesh Golani§, David Choffnes‡, Phillipa Gill§, Alan Mislove‡

Stony Brook University §, Northeastern University ‡

Introduction

Cellular data networks today

- **Provide Fast data** access for applications
- **Enable a range of services**

Since bandwidth is expensive, a mobile data provider might opt to **treat data traffic for some of these services differently**, for reasons such as:

- **Managing Network Load**
- **Degrading Performance of Competing Services**

Measuring and identifying this on mobile is particularly challenging since:

- **Mobile platforms have limited resources**
- Mobile **apps and services are always expanding** and designing specifically for each of them would be infeasible

Our approach uses a **novel method for detecting differentiation using Meddle VPN**[1].

Previous work addressed fixed-line environments, but were limited in the following ways:

	Switzerland [2]	Glasnost[3]	Our Approach
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

- **Identify and Expose Differentiation**
- **Measure How Differentiation Affects Performance**
  - **Delaying**
  - **Throughput Throttling**
  - **Jitter**
  - **Packet Dropping**
- **Classify Differentiation for Popular Applications**
  - **VoIP**
  - **Media Streaming**
  - **File Sharing**
  - **Cloud Storage**

Methodology

1. Record

The application that is going to be tested is run while `tcpdump` **records all the packets on Meddle VPN server.** A `pcap` packet trace file is the result.

2. Create Transcript

The recorded packet trace is parsed and a **transcript of messages is generated.**

3.Replay

- **Client and server exchange packets according to transcript**
- **repeat over unencrypted channel**
- **repeat over encrypted channel (using Meddle VPN)**

4. Compute statistics

- **➔Transfer over time**
- **➔Round-trip times** (of connection with the largest amount transfer)
- **➔Throughput**
- **➔Jitter**
- **➔Packet Loss**

Challenges

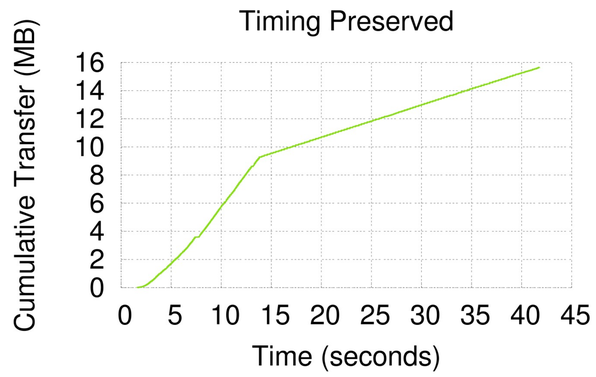
The key challenge is to replay in such a way that all of the elements that could trigger the differentiation are preserved.

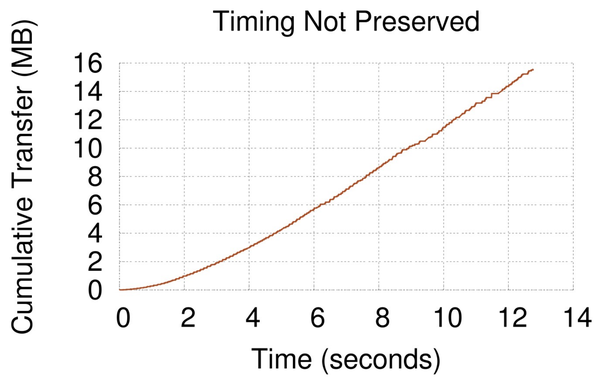
➔ Challenge: Whether to preserve inter-packet timing for TCP

We chose **not to preserve the timing since it might prevent the differentiation** from being triggered in a network with higher bandwidth. (the timing is always enabled for UDP)

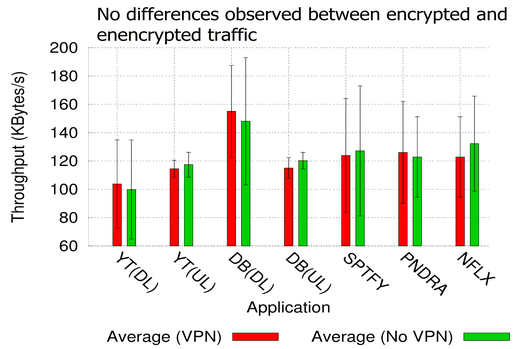
➔ Challenge: Whether to include "noise traffic" in replay

We decided to **keep network traffic from background applications** in the replay trace because we **did not want to omit important traffic that could trigger differentiation.**



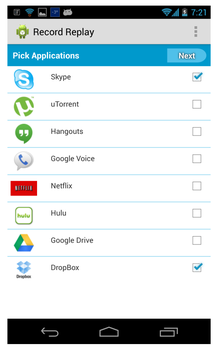


Proof of Concept



Using our controlled experiments, we can rigorously determine whether there is differentiation — or prove that there is none at all.
Results of a controlled test of **AT&T's network** shows **no major difference in metrics** between encrypted and unencrypted replays, meaning that there is **no differentiation in place** for the tested application.

Future Work



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

Our future goals are:

- **Create a mobile application** to enable wider adoption of the test.
- **Run on networks where we know there is differentiation** to validate our techniques.
- **Allow users to record and replay** their own traces (longer term).
- **Create a *Differentiation Watch* website** and blog to report the results for different networks and blog about newly detected 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