1. INTRODUCTION
Mobile devices play an essential role in the Internet today, and there is an increasing interest in using them as a vantage point for network measurement from the edge. At the same time, these devices store personal, sensitive information, and there is a growing number of applications that leak it. We propose AntMonitor – the first system of its kind that supports (i) collection of large-scale, semantic-rich network traffic in a way that respects users’ privacy preferences and (ii) detection and prevention of leakage of private information in real time. The first property makes AntMonitor a powerful tool for network researchers who want to collect and analyze large-scale yet fine-grained mobile measurements. The second is an incentive for using AntMonitor and contributing data.

2. SYSTEM OVERVIEW
The AntMonitor system consists of three components: an Android app, AntClient, and two servers, called AntServer and LogServer for routing and collecting packets, respectively. A detailed description of the system appears in the SIGCOMM Workshop [1]. Here, we provide an overview of the functionalities of AntMonitor.

**Data Collection.** AntClient establishes a VPN service on the device to intercept all network traffic, log packets, and upload them to LogServer at a later time. LogServer receives crowdsourced data from a large number of devices, which enables global analysis. In our pilot deployment, we collected and analyzed 20 GB of data from 151 applications, and were able to classify network flows to a specific app with F1-score of 70.1% using a Linear SVM [1].

**Privacy Control.** AntClient provides users with the flexibility of choosing which apps to log, as shown in Fig. 1(b). In addition, AntClient allows users to protect data of two types: (i) sensitive information, such as, IMEI, and phone number, and (ii) custom strings. If protection is enabled, then AntClient inspects every outgoing packet for any of the protected strings, before sending it out.

3. DEMONSTRATION
Our demo will show how users can contribute data and how AntMonitor detects and prevents privacy leaks in real time. A video of the demo can be found on the project website [2].

We will start by opening AntClient on an Android phone and connecting it to the AntServer. Then we will select some apps whose traffic will be logged, as shown in Fig. 1(b). Next, we will continue to use the phone as usual, e.g., check weather, email, and play a game. Afterward, we will ask AntClient to upload data to LogServer. We will observe the log files arriving at LogServer by showing the LogServer’s database on a laptop screen. We will also demonstrate real-time measurements that show the high performance (throughput) and low cost (CPU usage) of AntClient.

In the second part of the demo, we will navigate to AntClient’s privacy screen and select several strings to monitor. We will then use various apps known to leak private information. When a leak occurs, AntClient will generate a notification, as shown in Fig. 1(c). We can then either allow the leak, replace the leaking string, or block the leak (packet) completely. Lastly, we will navigate to AntClient’s leak history screen to review the number of leaks from the apps we used and the actions (allow, replace, block) we took.

4. REFERENCES