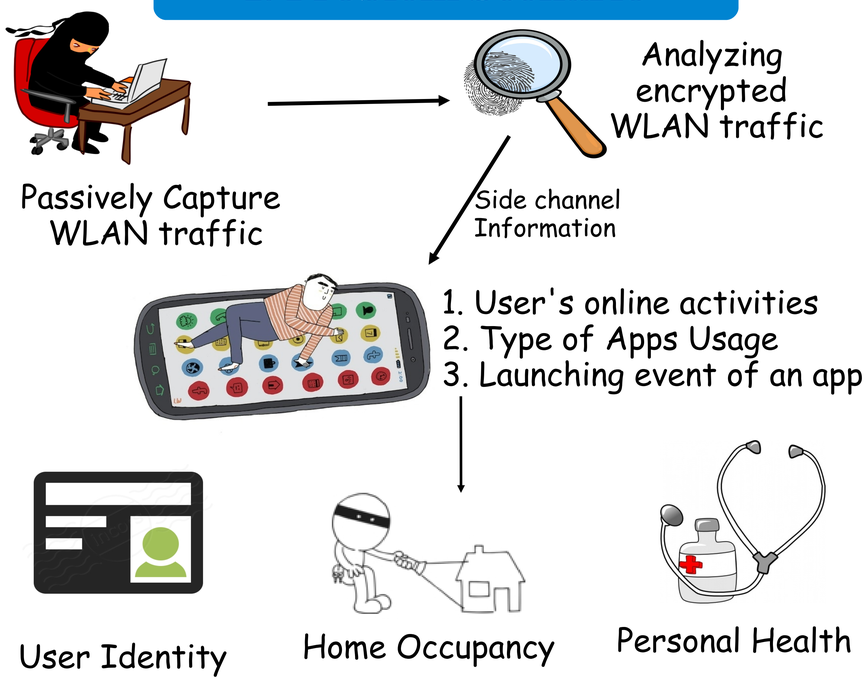
SafeWLAN: A WLAN-based SDN Approach for Securing WLAN Traffic

Mostafa Uddin, Ashish Kshirsagar and Tamer Nadeem
Old Dominion University

SwimSys Lab
http://swimsys.cs.odu.edu

1. Problem Scenario



2. Challenges & Motivation

- **Mobile app typically has a more focused objective – something with a single functionality.**
- **User runs many sensitive apps in their mobile devices.**
- **Each app has unique frame size interaction sequence – *application DNA*.**

STUDY ON HEALTH SECTOR:

1. Clinicians use 6.4 mobile devices per day on average.
2. 66% doctor use tablets for medical purpose.
3. 70% physician use smartphone to research medication at least once a week.

1. 93% physician believe mobile health app can improve patients health care.
2. 90% physician prefer patient would upload their medical data directly to EHR.
3. 89% physician would recommend an app to the patient for future use.

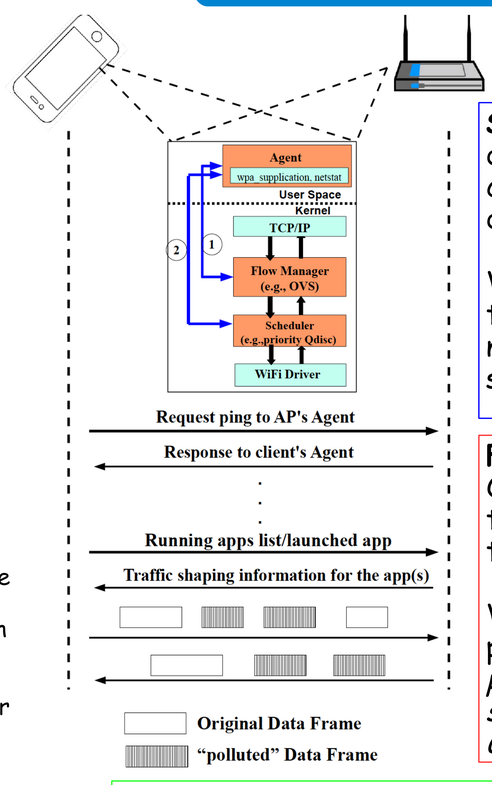
Mobile devices can be easily compromise.

Physician feel vulnerable to use mHealth apps.

3. Design Principle

- **TRANSPARENT** to the client and the server side of the apps.
- **FLEXIBLE** and **CONFIGURABLE** to the user or the network administrator.
- **COMPATIBLE** with the existing Off-the-shelf mobile devices.
- **EFFICIENTLY** hide the wireless traffic at the edge of the network.

4. System Components



Scheduler can create artificial delay of the traffic flow by controlling the start/stop dequeueing of the packet.

We develop the **scheduler** with two linux multiq qdisc, one for normal apps and other for sensitive apps.

Flow Manager is a software OpenFlow switch that direct traffic flow of sensitive apps to the proper qdisc.

We extend the **Flow Manager** to provide traffic shaping actions: *padding*, *aggregating*, and *splitting, in addition with IPsec, GRE tunnel.*

Agent is a user space software that control the Flow manager and the Scheduler.

The **Agent in mobile device** provide application awareness.

The **Agent in AP/Controller** provide per-flow/per-app privacy policies.

5. Methodology

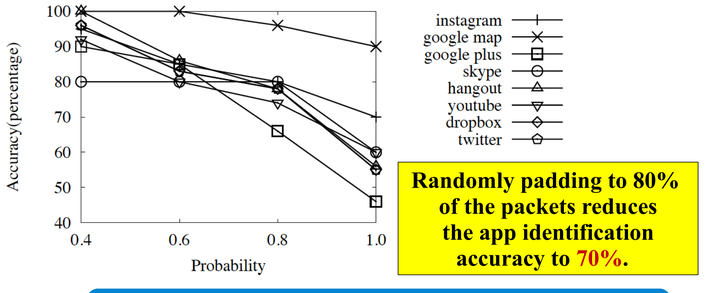
- Initially, the **hand-shaking** happens between the agents in the mobile device and the AP/controller.
  - Define per-flow/per-app traffic actions.
- The agent in the mobile device uses **OpenFlow protocol to add actions** in OVS for securing sensitive app's traffic flows.
  - Apply IPsec, GRE, traffic shaping.
- The agent in the AP/Controller apply **reverse actions in OVS** to retract the original packet.

6. Evaluation

We can identify the app with at least 90% accuracy by looking at the initial frame interaction sequence.

IMPLEMENTATION

→ We extend the OVS kernel ***Datapath*** for new action of ***padding*** **random bytes** at the end of the packet.
→ We use ***IP option*** **header** to encode the padding information in the IP packet.
→ We **randomly select** the packet of a flow for applying the padding action.



Randomly padding to 80% of the packets reduces the app identification accuracy to 70%.

7. Future Work

- Analyzing the network and computation overhead.
- Impact on user's QoE or app performance.
- Evaluation of securing the sensitive app's traffic flow and its content.
- Maintain the QoS of the traffic flow.