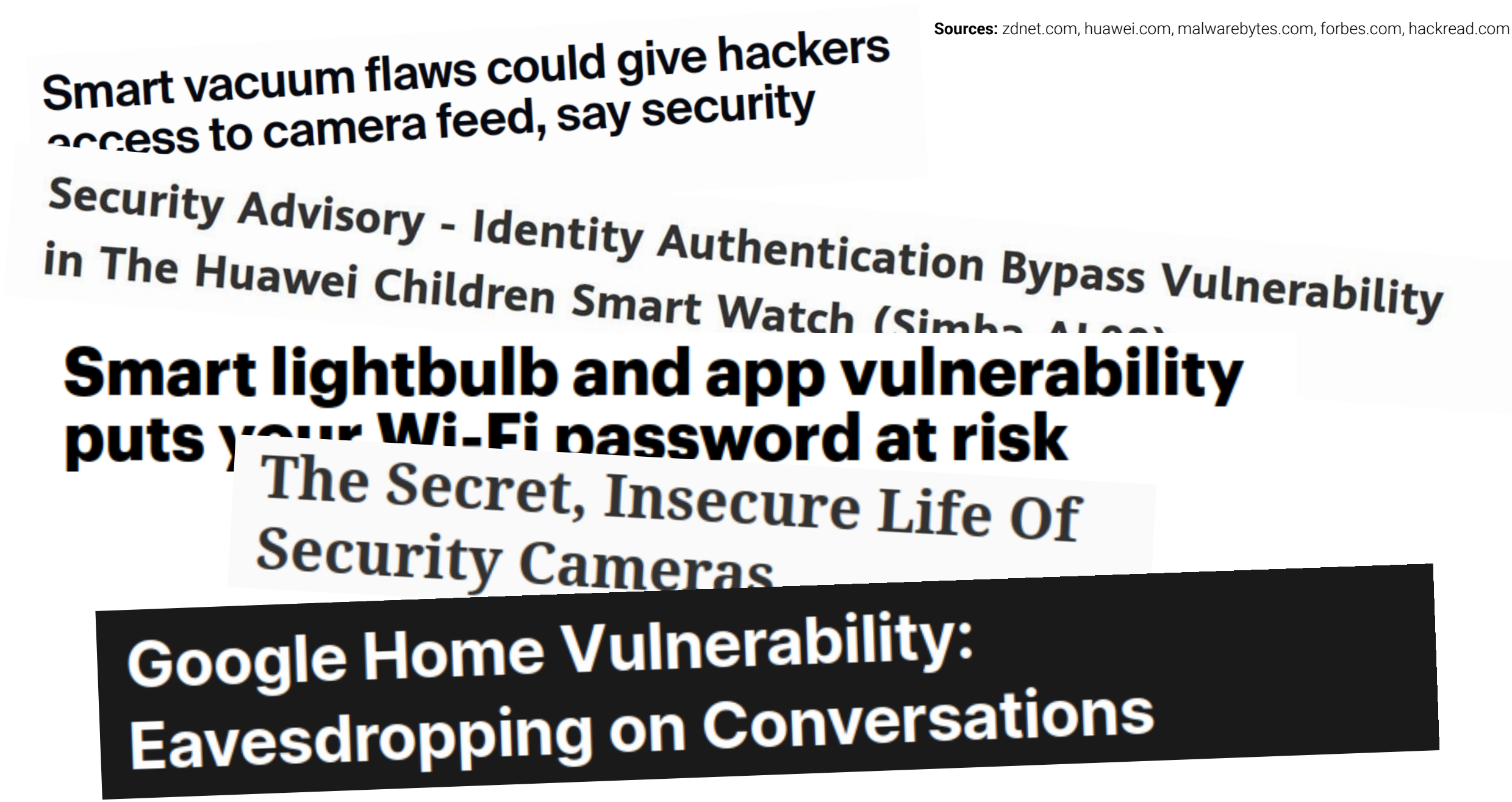
IoTFlow

Inferring IoT Device Behavior at Scale through Static Mobile Companion App Analysis

David Schmidt — TU Wien
Carlotta Tagliaro — TU Wien
Kevin Borgolte — Ruhr University Bochum
Martina Lindorfer — TU Wien

Sources: zdnet.com, huawei.com, malwarebytes.com, forbes.com, hackread.com



Sources: Zhou et al. (USENIX Security 2019), Chen et al. (NDSS 2018), Wang et al. (USENIX Security 2019)

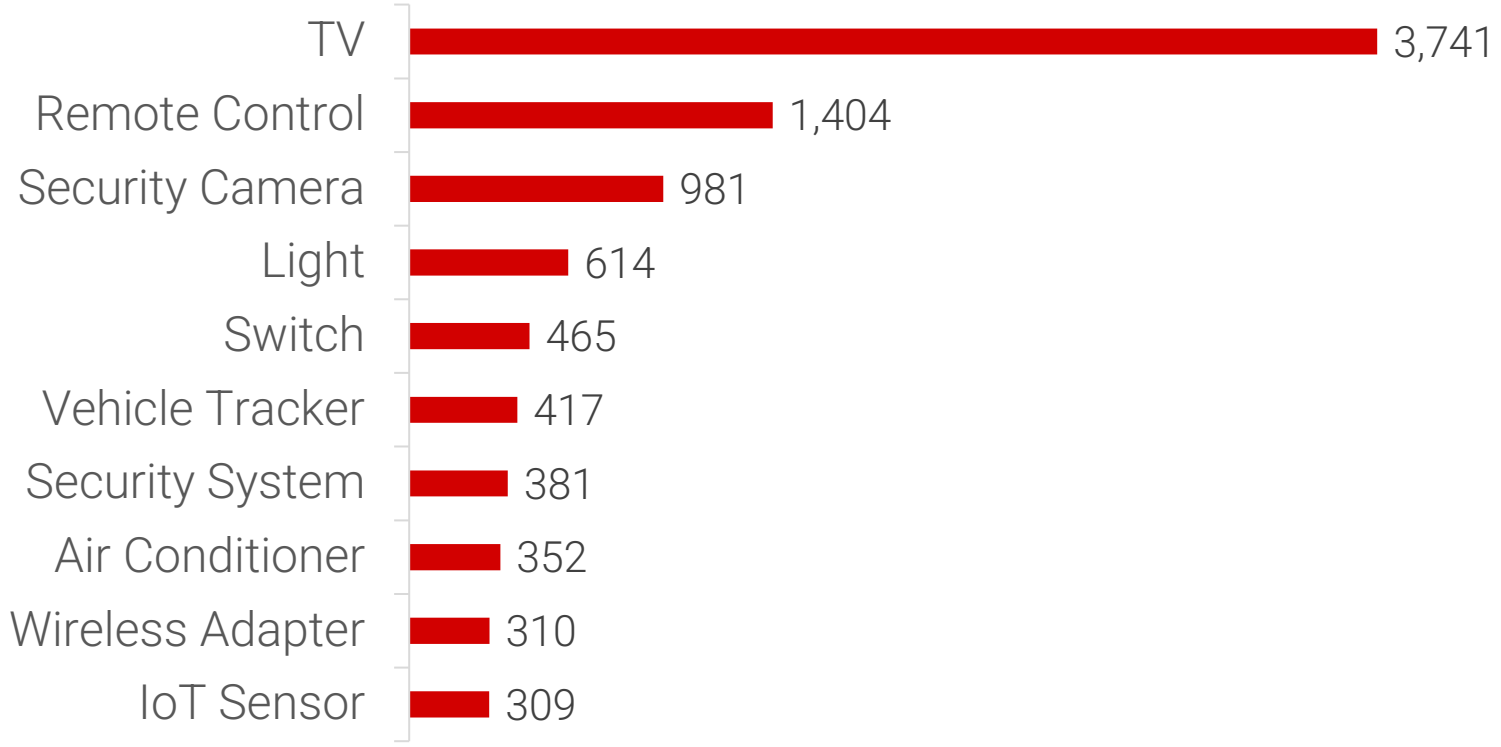
IoT Device Analysis

Figure 4: Smart Home Devices Used in Our Experiments

Fig. 4: IoT Devices Used for Our Experiments

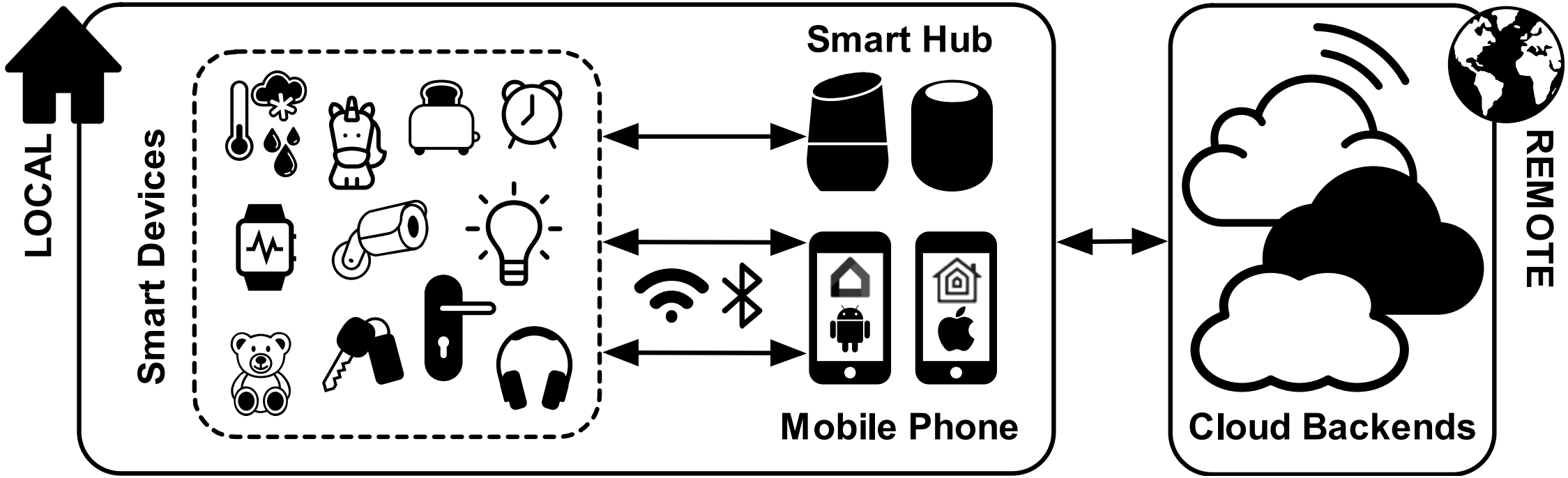
Figure 6: Smart home IoT devices for vulnerability validation

Diversity of IoT Devices

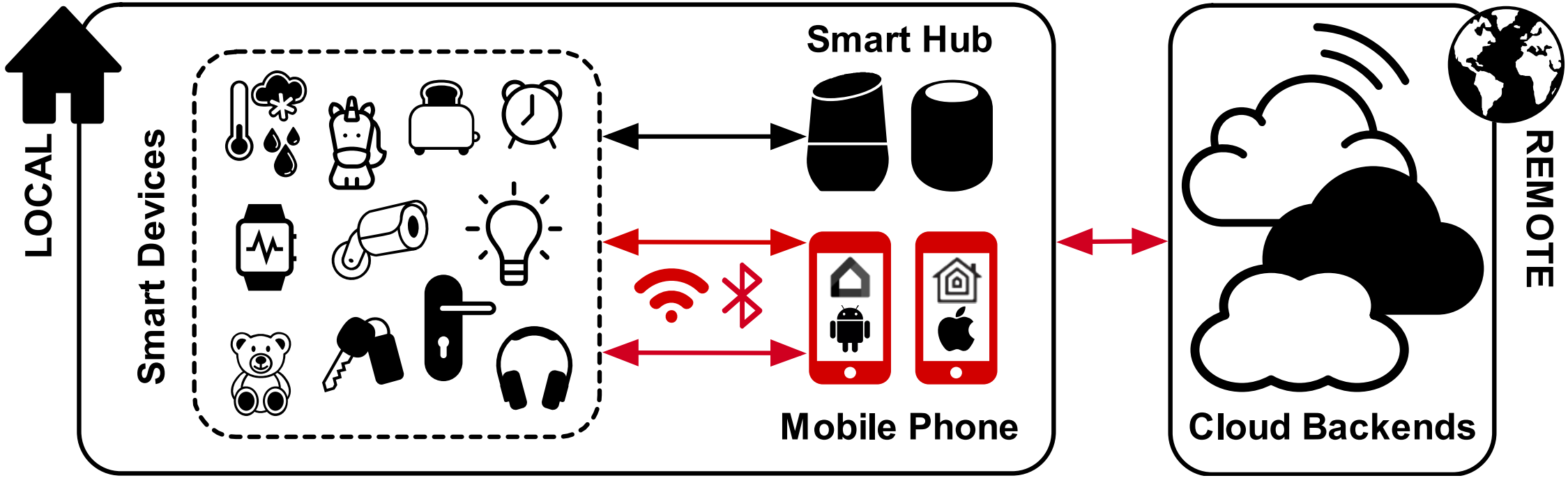


Source: Jin et al. (ACM CCS 2022)

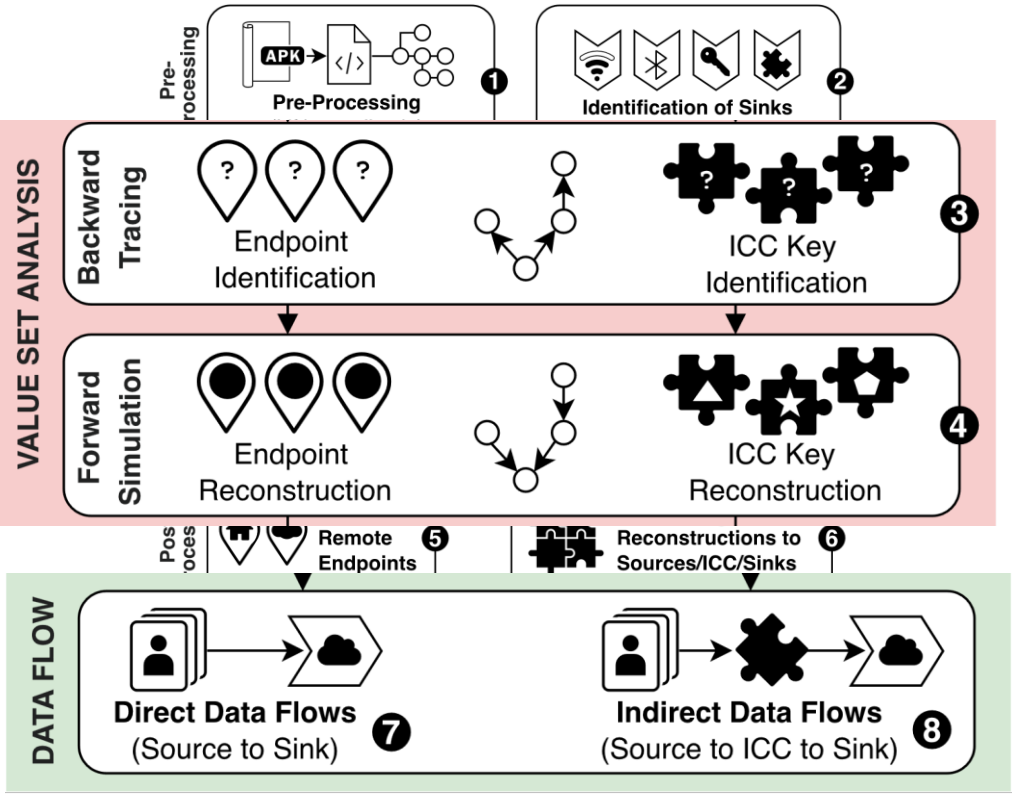
IoT Ecosystem



IoT Ecosystem

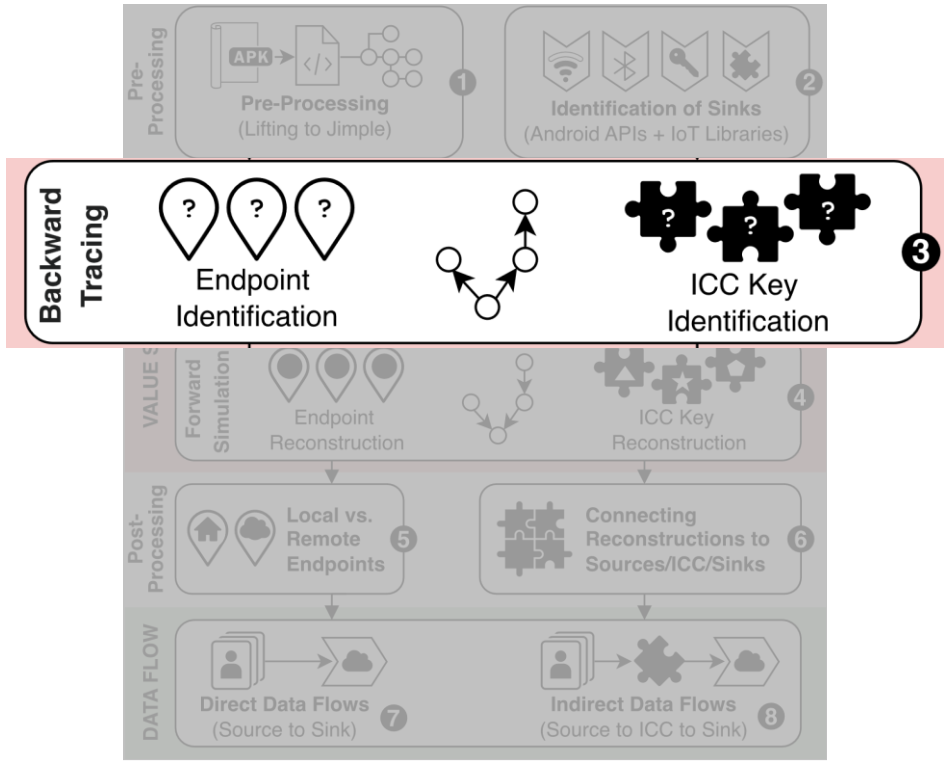


IoTFlow: Approach



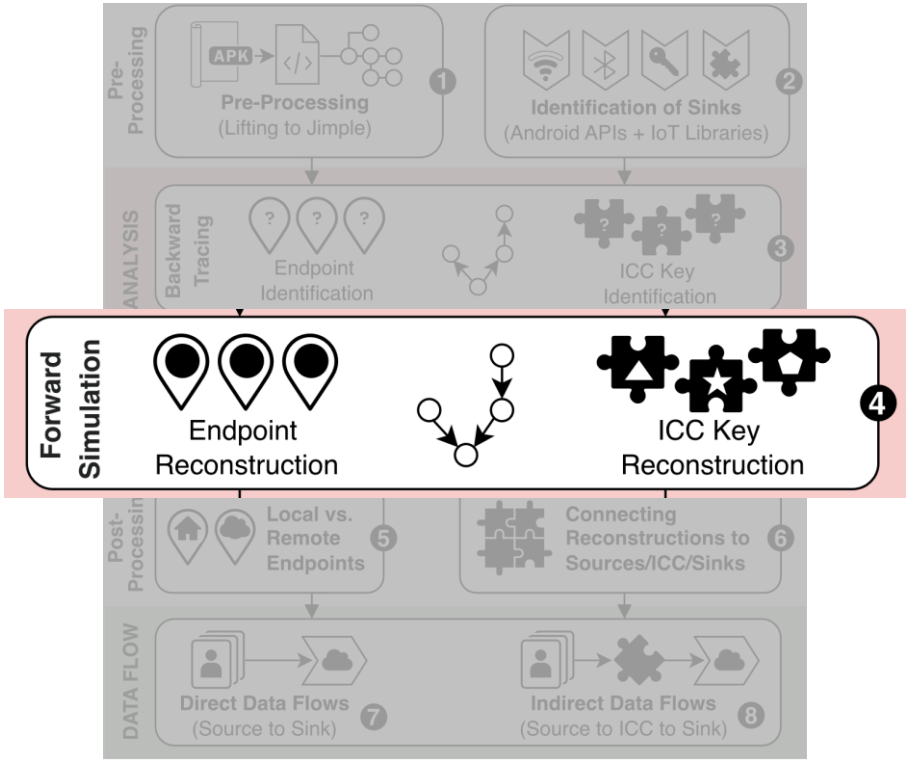
- We analyze mobile companion apps because they need to communicate with smart devices
- Value Set Analysis
  - Identify all potential values that a variable might take at a specific point in a program
- Dataflow Analysis
  - Determine where data comes from, or data is going to

IoTFlow: Backward Tracing




```
String BLE_DATA = "device";

void onCharacteristicRead(BluetoothGattCharacteristic bgc, /*...*/) {
    Intent intent = new Intent(DeviceActivity.class);
    intent.putExtra(BLE_DATA, parseData(bgc.getValue()));
}

MqttConfig config = new MqttConfig()
    .setEndpoint("example.com").setTopic("things/Wifi_device");

class DeviceActivity {

    void onCreate(Bundle bundle) {
        String data = getIntent().getStringExtra(BLE_DATA);
        mqtt.publish(new MqttMessage(data, config));
    }

    void publish(MqttMessage m) {
        MqttManager mqttManager = new MqttManager(m.config.endpoint);
        mqttManager.publishString(m.data, m.config.topic);
    }

}
```

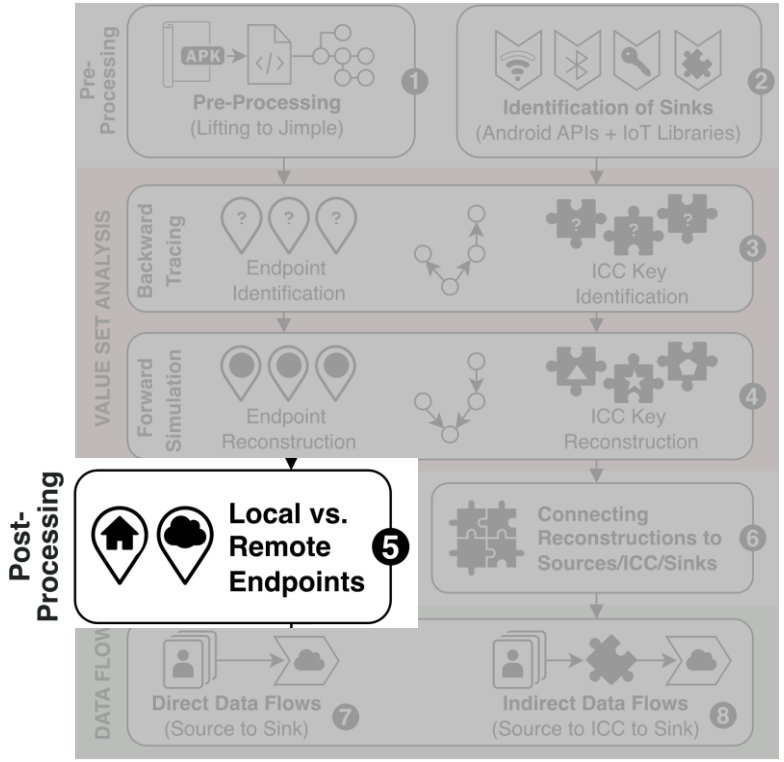
IoTFlow: Forward Simulation




```
String BLE_DATA = "device";

void onCharacteristicRead(BluetoothGattCharacteristic bgc, /*...*/) {
    Intent intent = new Intent(DeviceActivity.class);
    intent.putExtra(BLE_DATA, parseData(bgc.getValue()));
}

MqttConfig config = new MqttConfig()
    .setEndpoint("example.com").setTopic("things/Wifi_device");

class DeviceActivity {

    void onCreate(Bundle bundle) {
        String data = getIntent().getStringExtra(BLE_DATA);
        mqtt.publish(new MqttMessage(data, config));
    }

    void publish(MqttMessage m) {
        MqttManager mqttManager = new MqttManager(m.config.endpoint);
        mqttManager.publishString(m.data, m.config.topic);
    }

}
```

Current values: {"BLE_DATA": **"device"**}
"topic": "things/Wifi_device"}, "data": "Intent -> device"}

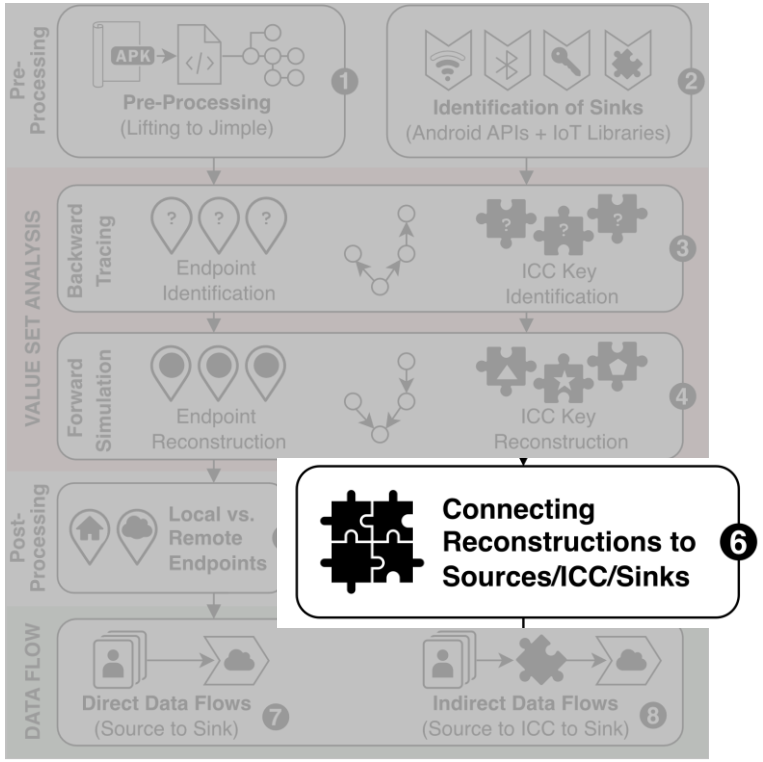
IoTFlow: Endpoint Classification



We reconstructed the MQTT endpoint: **example.com** and we now can analyze whether it is a **local/remote endpoint**

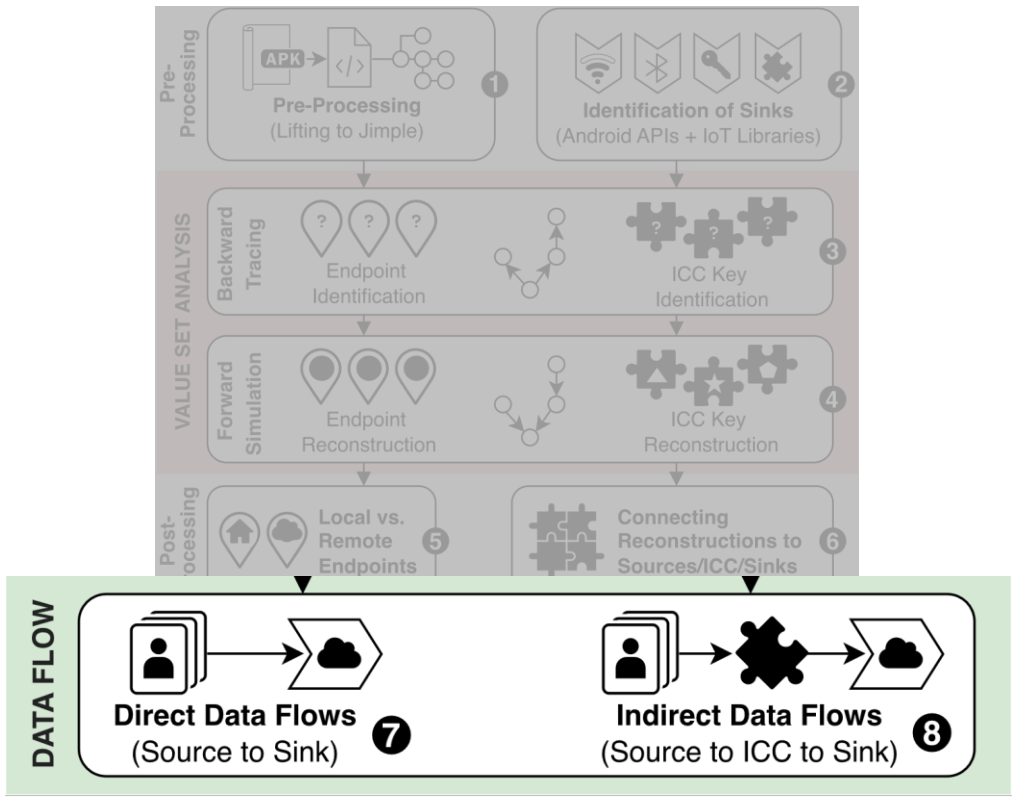
```
MqttConfig config = new MqttConfig()
    .setEndpoint("example.com").setTopic("things/Wifi_device");

class DeviceActivity {

    void onCreate(Bundle bundle) {
        String data = getIntent().getStringExtra(BLE_DATA);
        mqtt.publish(new MqttMessage(data, config));
    }

    void publish(MqttMessage m) {
        MqttManager mqttManager = new MqttManager(m.config.endpoint);
        mqttManager.publishString(m.data, m.config.topic);
    }

}
```

IoTFlow: Connection Reconstructions




```
String BLE_DATA = "device";

void onCharacteristicRead(BluetoothGattCharacteristic bgc, /*...*/) {
    Intent intent = new Intent(DeviceActivity.class);
    intent.putExtra(BLE_DATA, parseData(bgc.getValue()));
}

MqttConfig config = new MqttConfig()
    .setEndpoint("example.com").setTopic("things/Wifi_device");

class DeviceActivity {

    void onCreate(Bundle bundle) {
        String data = getIntent().getStringExtra(BLE_DATA);
        mqtt.publish(new MqttMessage(data, config));
    }

    void publish(MqttMessage m) {
        MqttManager mqttManager = new MqttManager(m.config.endpoint);
        mqttManager.publishString(m.data, m.config.topic);
    }

}
```

IoTFlow: Data Flows

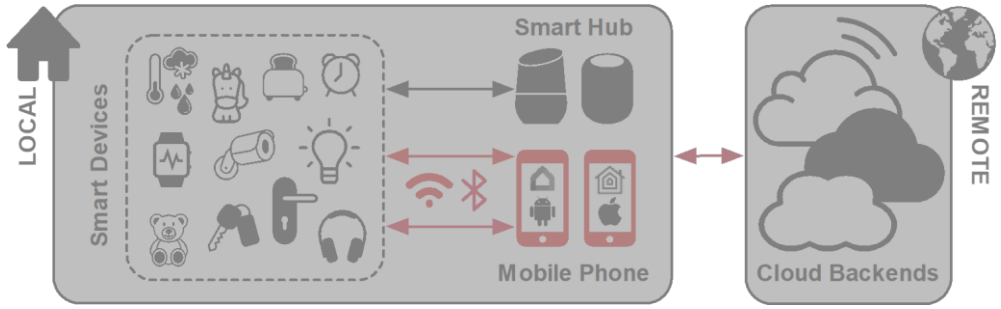



```
String BLE_DATA = "device";

void onCharacteristicRead(BluetoothGattCharacteristic bgc, /*...*/) {
    Intent intent = new Intent(DeviceActivity.class);
    intent.putExtra(BLE_DATA, parseData(bgc.getValue()));
}

MqttConfig config = new MqttConfig()
    .setEndpoint("example.com").setTopic("things/Wifi_device");

class DeviceActivity {

    void onCreate(Bundle bundle) {
        String data = getIntent().getStringExtra(BLE_DATA);
        mqtt.publish(new MqttMessage(data, config));
    }

    void publish(MqttMessage m) {
        MqttManager mqttManager = new MqttManager(m.config.endpoint);
        mqttManager.publishString(m.data, m.config.topic);
    }

}
```

Data flow from Bluetooth source to remote endpoint: **example.com**, involving an ICC

IoTFlow Analysis

- With IoTFlow, we can answer the questions:
  - RQ1: How do companion apps and devices communicate?
  - RQ2: Who are companion apps communicating with?
  - RQ3: Which data are companion apps sharing (and how)?
- Large-scale analysis
  - 9,889 companion apps
  - 947 general-purpose apps

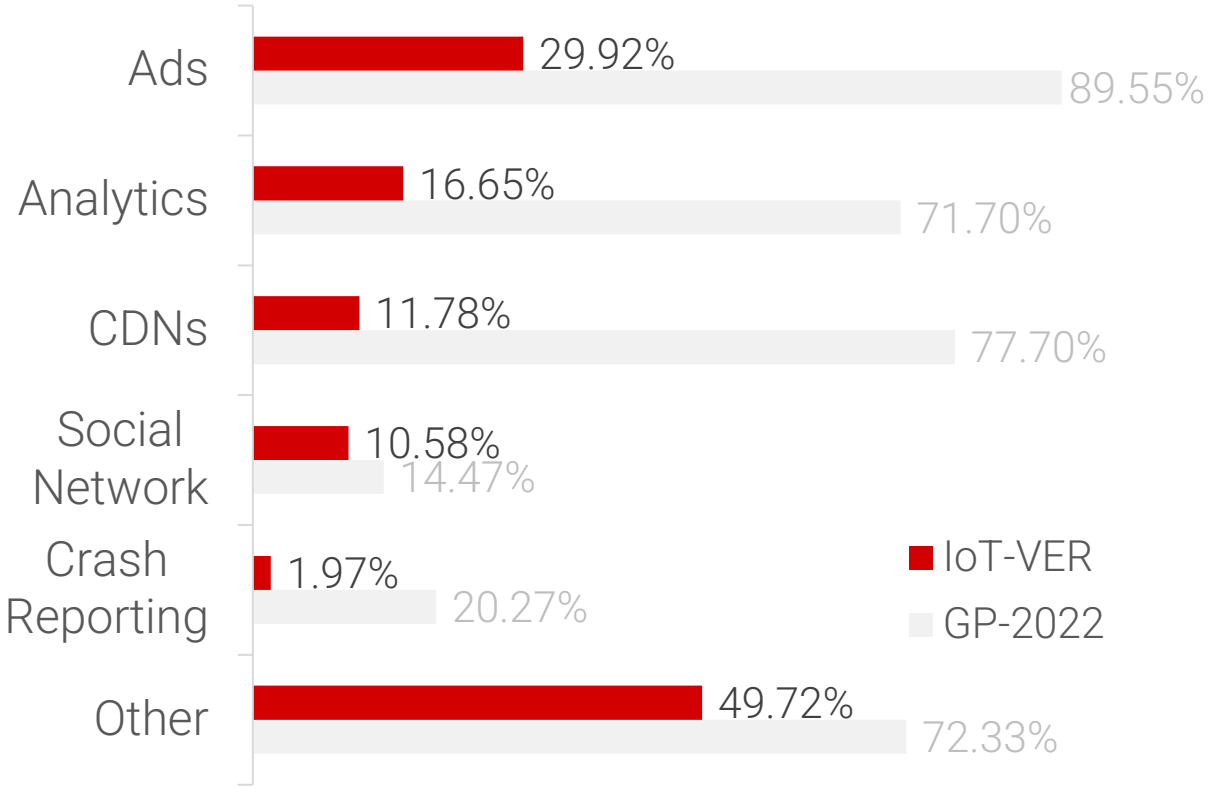


RQ1: Direct-Device Communication

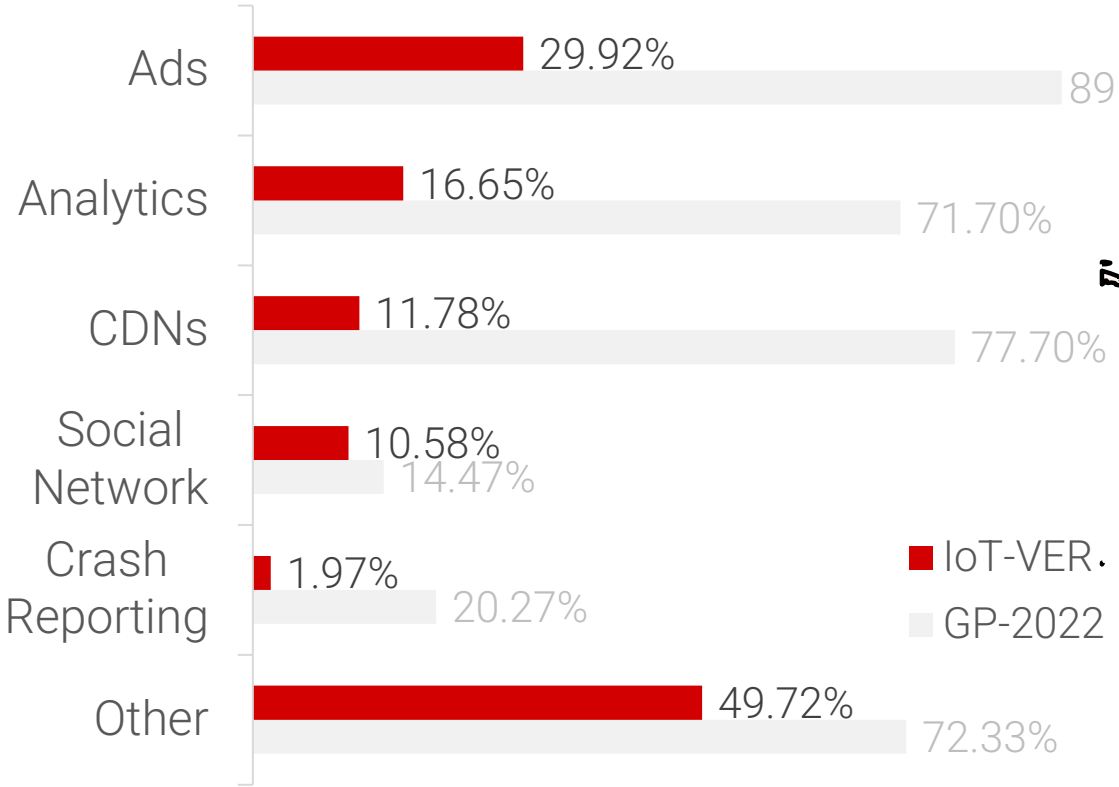
	Companion Apps	General-Purpose Apps
Bluetooth Permission	64.26%	19.01%
Local IP Address	14.99%	2.21%
Multi- and Broadcasts	4.57%	0.42%
User Input Address	1.24%	0.11%

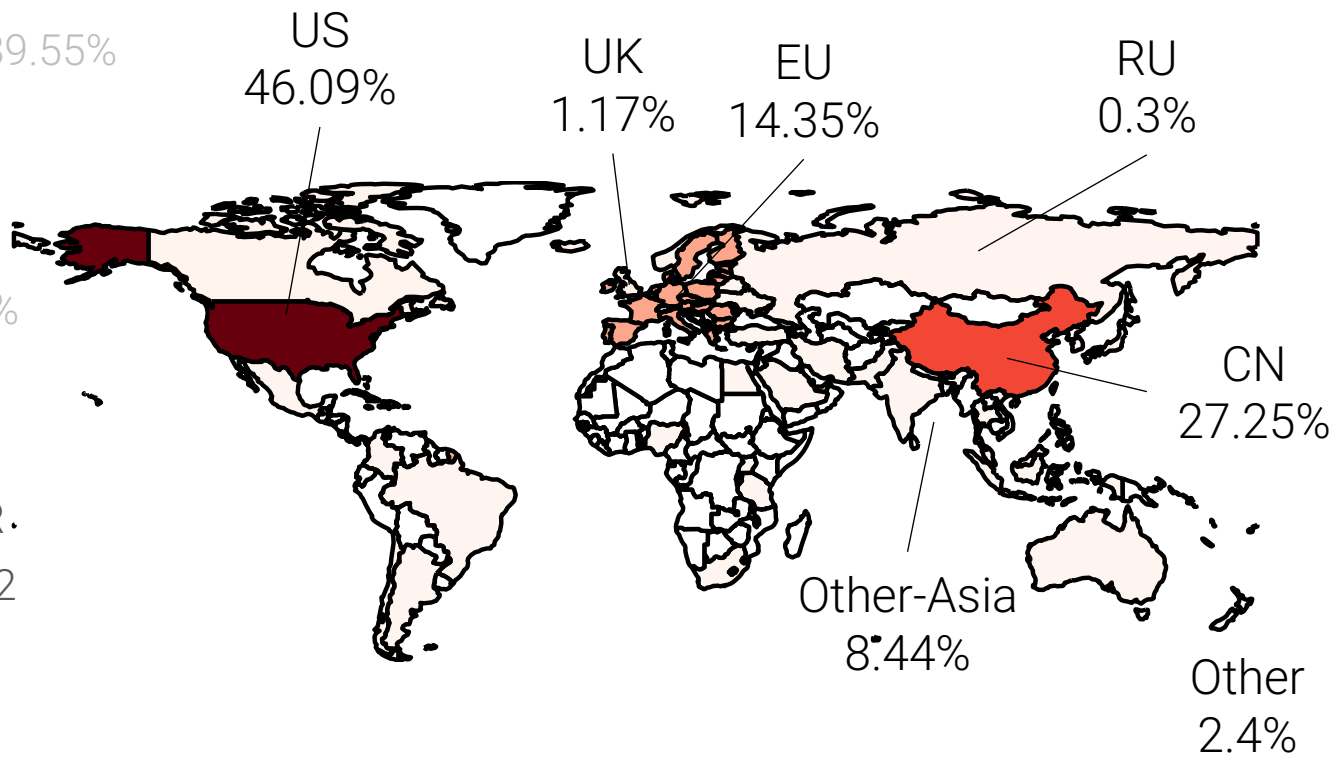
Source: com.tplink.kasa.android

RQ2: With Whom Companion Apps Communicate

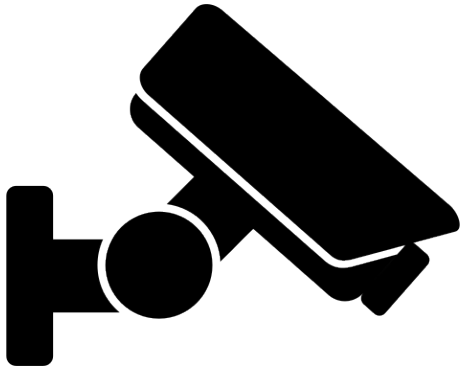


RQ2: With Whom Companion Apps Communicate





RQ3: What Data Companion Apps Share



- Case study: **Smart camera**
  - IMEI used for authentication
  - Insecure cryptography with hard-coded key
    - Password hashed with MD5
    - Username and password encrypted with 3DES
      - IoTFlow reconstructs “obfuscated” key

Takeaways

New large-scale analysis of companion apps

- Combination of value set analysis and data flow analysis

Analyzed 9,889 companion and 947 general-purpose apps

- Identified differences in their network behavior

Discovered security and privacy issues

- Broken encryption
- Send personal identifiable information

IoTFlow: Inferring IoT Device Behavior at Scale through Static Mobile Companion App Analysis

David Schmidt, Carlotta Tagliaro, Kevin Borgolte, Martina Lindorfer

TU WIEN <!seclab> RUB SOFTWARE SECURITY WWTF VIENNA SCIENCE AND TECHNOLOGY FUND DFG CASA CYBER SECURITY IN THE AGE OF LARGE-SCALE ADVERSARIES SBA Research