# Decomposition of MAC Address Structure for Granular Device Inference

Jeremy Martin*, Erik C. Rye*,
Robert Beverly+

*US Naval Academy
Annapolis, MD
+US Naval Postgraduate School
Monterey, CA

December 9, 2016

# Outline

1. Introduction
2. Methodology
3. Results
4. Conclusions

# Furious MAC

## Motivation

### Layer-2 Media Access Control (MAC) Addresses:

- Ubiquitous (Ethernet, WiFi, Bluetooth, etc)
- Uniqueness ensured via IEEE allocations
- Readily available, regardless of encryption, associated state, or user interaction

### What's in a MAC?

DE: AD: BE: EF: CA: FE

- First 3 bytes (OUI): device manufacturer
  - FuriousMAC: can we trust the first 3 bytes alone?
- FuriousMAC: what can we infer from 3 *least* significant bytes?
  - Contiguous?
  - Sequential?
  - Predictable? e.g., fine-grained make and model?

## Layer-2 Media Access Control (MAC) Addresses:

- Ubiquitous (Ethernet, WiFi, Bluetooth, etc)
- Uniqueness ensured via IEEE allocations
- Readily available, regardless of encryption, associated state, or user interaction

## What's in a MAC?

DE: AD: BE: EF: CA: FE

- First 3 bytes (OUI): device manufacturer
  - FuriousMAC: can we trust the first 3 bytes alone?
- FuriousMAC: what can we infer from 3 *least* significant bytes?
  - Contiguous?
  - Sequential?
  - Predictable? e.g., fine-grained make and **model**?

## Layer-2 Media Access Control (MAC) Addresses:

- Ubiquitous (Ethernet, WiFi, Bluetooth, etc)
- Uniqueness ensured via IEEE allocations
- Readily available, regardless of encryption, associated state, or user interaction

## What's in a MAC?

DE: AD: BE: EF: CA: FE

- First 3 bytes (OUI): device manufacturer
  - FuriousMAC: can we trust the first 3 bytes alone?
- FuriousMAC: what can we infer from 3 *least* significant bytes?
  - Contiguous?
  - Sequential?
  - Predictable? e.g., fine-grained make and **model**?

## Layer-2 Media Access Control (MAC) Addresses:

- Ubiquitous (Ethernet, WiFi, Bluetooth, etc)
- Uniqueness ensured via IEEE allocations
- Readily available, regardless of encryption, associated state, or user interaction

## What's in a MAC?

DE: AD: BE: EF: CA: FE

- First 3 bytes (OUI): device manufacturer
  - FuriousMAC: can we trust the first 3 bytes alone?
- FuriousMAC: what can we infer from 3 *least* significant bytes?
  - Contiguous?
  - Sequential?
  - Predictable? e.g., fine-grained make and **model**?

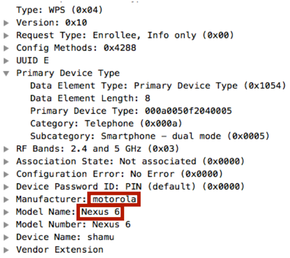
### Fine-Grained Wireless Device Fingerprinting. Why:

- Support policy-based security
- Crowd density and population diversity studies
- User profiling, tracking, and security threats
- Targeted device attacks
- Reconnaissance (e.g., IoT devices such as security cameras, thermostats, and automobiles)

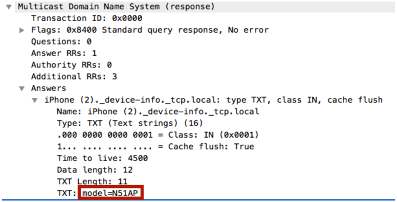
1. Introduction
2. Methodology
3. Results
4. Conclusions

## Enabling device manufacturer and model **predictions** for previously unknown MACs:

- FuriousMAC is first *trained* on MACs with known manufacturer and model
- Derive mapping of MAC address to device manufacturer model
  - Management frames containing WPS-enriched data fields
  - Discovery protocols, primarily mDNS
  - Easily extensible



```
  Type: WPS (0x04)
▶ Version: 0x10
▶ Request Type: Enrollee, Info only (0x00)
▶ Config Methods: 0x4288
▶ UUID E
▼ Primary Device Type
    Data Element Type: Primary Device Type (0x1054)
    Data Element Length: 8
    Primary Device Type: 000a0050f2040005
    Category: Telephone (0x000a)
    Subcategory: Smartphone - dual mode (0x0005)
▶ RF Bands: 2.4 and 5 GHz (0x03)
▶ Association State: Not associated (0x0000)
▶ Configuration Error: No Error (0x0000)
▶ Device Password ID: PIN (default) (0x0000)
▶ Manufacturer: motorola
▶ Model Name: Nexus 6
▶ Model Number: Nexus 6
▶ Device Name: shamu
▶ Vendor Extension
```



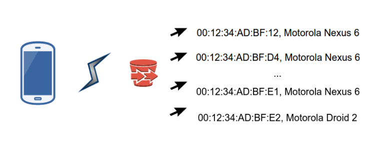
```
▼ Multicast Domain Name System (response)
    Transaction ID: 0x0000
  ▶ Flags: 0x8400 Standard query response, No error
    Questions: 0
    Answer RRs: 1
    Authority RRs: 0
    Additional RRs: 3
  ▼ Answers
    ▼ iPhone (2)._device-info._tcp.local: type TXT, class IN, cache flush
        Name: iPhone (2)._device-info._tcp.local
        Type: TXT (Text strings) (16)
        .000 0000 0000 0001 = Class: IN (0x0001)
        1... .... .... .... = Cache flush: True
        Time to live: 4500
        Data length: 12
        TXT Length: 11
        TXT: model=N51AP
```

## Derive mapping of MAC address to device manufacturer model

- **Management frames with WPS-enriched data fields**
  - Access Points (Beacons and Probe Responses), client devices (Probe Requests) `manufacturer`, `model_name`, `model_number`, `device_name`, `primary_device_type.category`, `.subcategory` and `uuid_e`
  - **Advantages:** Unencrypted, non-associated state, low data-rates, wide range of device types
  - **Disadvantage:** Not used by all devices (iOS, Ubiquiti, etc.)
- **Discovery protocols, primarily mDNS**
  - mDNS data field, `dns.txt`: reveals a model identification key-value pair, correlates to a manufacturer and model
  - **Advantages:** Fills in some high profile gaps → iOS!!
  - **Disadvantages:** Layer-2 encryption, associated state, often higher data-rate, not used by all devices

## Training

- Using 802.11 management frames and unencrypted mDNS packets, we build a model of $MAC \rightarrow (manufacturer, model)$



00:12:34:AD:BF:12, Motorola Nexus 6

00:12:34:AD:BF:D4, Motorola Nexus 6

...

00:12:34:AD:BF:E1, Motorola Nexus 6

00:12:34:AD:BF:E2, Motorola Droid 2

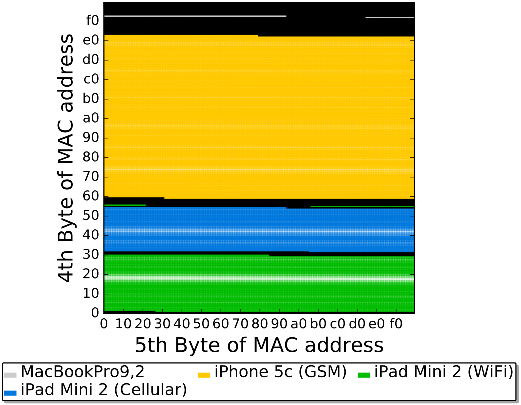
- Trained on 600GB of passively-collected 802.11 traffic:
  - Two billion frames
  - 2.8 million unique devices across a spectrum of IoT devices
  - January 2015 – May 2016
  - IRB exemption: Only examine MACs, management frames, and discovery protocols. No attempt to decrypt traffic or inspect user's communication.

### Locally assigned MAC address

- Privacy: randomized MAC addresses while in a non-associated state (Probe Requests)
- P2P: peer-to-peer connections utilize a locally assigned MAC address derived from the global MAC address
- APs and hotspots often advertise service using locally assigned MAC
- Ignored to preserve accuracy of mappings

# Methodology - Prediction

We perform a lexicographical comparison to find the manufacturer and model (Constrained such that the OUI must match)



Observed Models in `24:A2:E1` (Apple)

- Plot observed MAC addr-models by 4th and 5th bytes for all OUI
- Color between same models; color intensity relative to largest “gap”

# Outline

1. Introduction
2. Methodology
3. Results
4. Conclusions

## Results

- 802.11 Corpus Statistics
- Vendor MAC Address Allocation Strategies
- Prediction Validation

# 802.11 Corpus Statistics

Top 10 Manufacturers - Clients

| WPS | Count | % | non-WPS | Count | % |
|---|---|---|---|---|---|
| LGE | 11,184 | 22.60 | Apple | 231,214 | 44.36 |
| Ralink | 4,279 | 8.64 | Samsung | 48,617 | 9.33 |
| Motorola | 3,260 | 6.58 | Murata | 48,246 | 9.26 |
| HTC | 3,256 | 6.57 | Intel | 25,734 | 4.95 |
| Prosoft | 2,234 | 4.50 | HP | 15,287 | 2.94 |
| Amazon | 2,222 | 4.49 | Microsoft | 13,949 | 2.68 |
| Huawei | 1,905 | 3.83 | Ezurio | 12,385 | 2.38 |
| Asus | 1,659 | 3.34 | Epson | 6,839 | 1.32 |
| ZTE | 1,619 | 3.25 | Lexmark | 5,289 | 1.01 |
| Alco | 1,036 | 2.10 | Sonos | 4,542 | .09 |
| Other | 16,859 | 34.10 | Other | 109,271 | 20.96 |

Apple makes up ∼45% of the non-WPS devices, emphasizing how mDNS and WPS are complementary

# MAC Address Allocation

## OUI Complexity

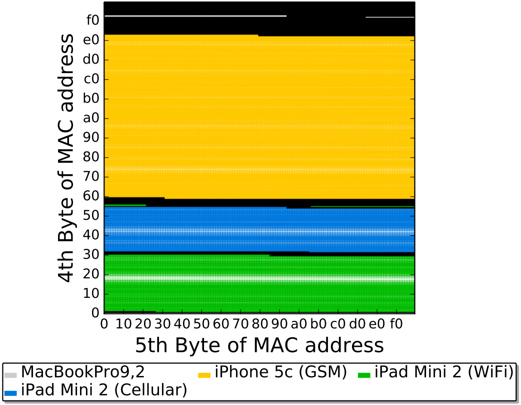
- There is no general pattern between manufacturers; some assign the entire OUI to only one model while others assign smaller ranges to dozens of distinct models
- The size and number of distinct ranges assigned to a model also follows no general rule

- 2,956 OUIs observed (WPS): $\sim$5,000 OUI to manufacturer pairings and $10,000$ OUI to model pairings
- 352 OUIs observed (Apple mDNS): 1,028 OUI to model pairings

## Visualization of Allocation Space

Next, we highlight several exemplar allocation schemes

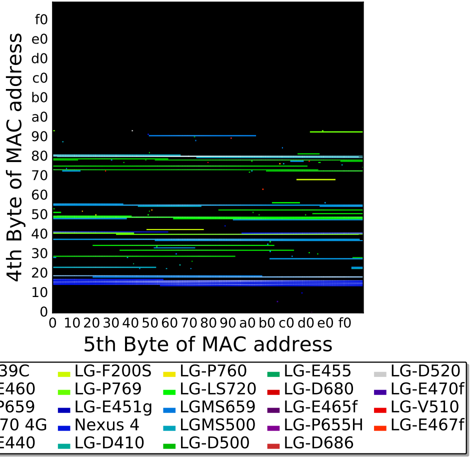
# MAC Address Allocation



Observed Models in `24:A2:E1` (Apple)

- Different generations w/in same OUI
- Different device types (phone, tablet, laptop)
- Different allocation sizes, large contiguous blocks
- Fine-grained, e.g., iPad Mini 2 WiFi vs. Cellular

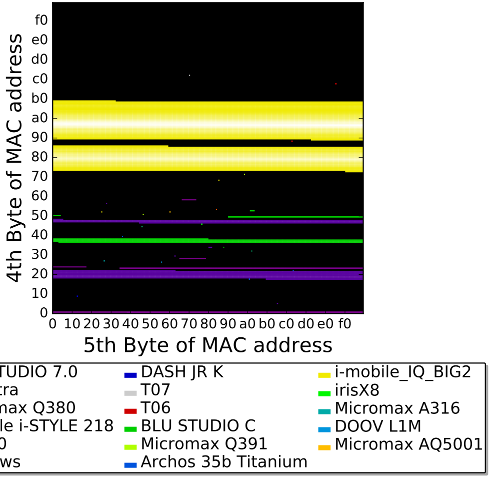
# MAC Address Allocation



Observed Models in `8C:3A:E3` (LGE)

- Micro-allocation of LGE smartphones
- Large blocks of unallocated or unobserved address space
- Fingerprinting is difficult compared to Apple

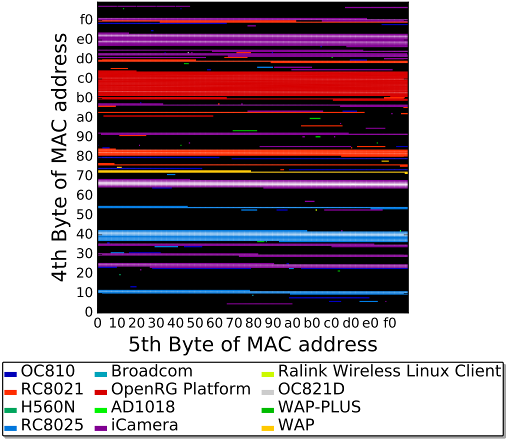
# MAC Address Allocation



Observed Models in 90:21:81 (Shanghai Huaqin)

- Diversity of Phone Manufacturers for a Single OUI
- Improves granularity of fingerprinting over OUI-based methods

# MAC Address Allocation



Observed Models in 00:0E:8F (Sercomm Corp.)

- Fine-grained model inference → 802.11-enabled cameras

# Validation - CRAWDAD dataset

## CRAWDAD Sapienza Dataset

- 11M probe requests from ∼ 160,000 unique devices
  - Captured from Italy in 2013; do not appear in our corpus
  - Anonymized data, to include MAC addresses

## Validate Against Our Corpus

- Identify CRAWDAD probe requests with distinguishing WPS-manufacturer/model fields and UUID-E
- Obtain global MAC from precomputed UUID-E lookup tables[1]
  - 1,746 global addresses recovered (test data), find closest MAC address "match" in our WPS corpus (training set)
  - If CRAWDAD manufacturer/model matches corpus closest-match manufacturer/model, inference is correct
  - Validation achieves 81.3% accuracy

[1] M. Vanhoef, C. Matte, M. Cunche, L. Cardoso, and F. Piessens. Why MAC Address Randomization is not Enough: An Analysis of Wi-Fi Network Discovery Mechanisms. In ACM AsiaCCS, 2016.

# Validation - Ground Truth

## Device Overview

- Procured 140 Apple and 139 Samsung devices
- Gamut of device types, life-cycles, and operating system versions
- Specifically evaluate the power Apple mDNS derived allocations

| Device | Precision | Recall | F-score |
|---|---|---|---|
| Apple | | | |
| - iPhone (iOS 7.0-) | .000 | .000 | 0 |
| - iPhone (iOS 8.0+) | .909 | .909 | .909 |
| - iPad/iPod (iOS 8.0+) | .857 | .900 | .877 |
| - All iOS 8.0+ Devices | .892 | .906 | .898 |
| - OS X | .771 | 1.00 | .870 |
| - Apple TV | .750 | 1.00 | .857 |
| - iOS 8.0+ and OS X | .850 | .934 | .890 |
| - All | .715 | .838 | .772 |
| Samsung | | | |
| - Galaxy S4 and prior | .684 | .892 | .774 |
| - Galaxy S5 to current | .475 | .863 | .613 |
| - Galaxy Tablets | .250 | .071 | .110 |
| - All | .598 | .761 | .670 |

# **Furious MAC** Validation - Cross Validation Test

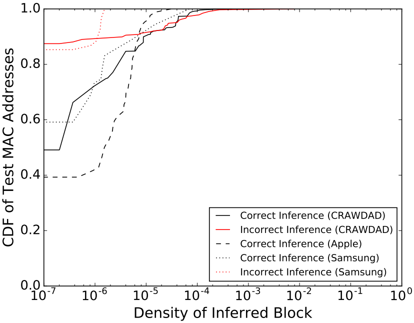
## 5-Fold Cross Validation

- Partition corpus' WPS and mDNS datasets into five random sets
- For MAC addresses in each set (test data), find the closest-matching MAC address in remaining sets (training data)
  - Compare using simple distance (48-bit integer representation) versus lexicographical distance
  - Manufacturer/model in test set compared to manufacturer/model in training set
  - Each set is used once as test data against the remaining four sets

## Validation

- Achieve average accuracy:
  - ∼90.95% (lexicographical distance) vs ∼91.16% (simple distance)
  - ∼10% improvement over the accuracy we obtain when testing against CRAWDAD dataset
  - ∼3% improvement over our validation using ground truth devices

# Furious MAC  Validation - Density vs Inference



- **Block density** – $\frac{\text{\# of device observations}}{\text{size of inferred model range}}$
- CRAWDAD density analysis
  - 55% of correct inferences within non-trivial block density
  - 85% of incorrect inferences fall outside of any block (density of 0)
  - Only 1 incorrect Apple inference falls inside a block

1 Introduction

2 Methodology

3 Results

4 Conclusions

## MAC address allocation is complex but generally non-random

- Vendors allocate contiguous blocks from their OUIs to individual device models.

**This determinism illustrates two concerns:**

  - management and discovery protocols allow significant privacy leaks
  - the allocation of MAC addresses lends itself to device fingerprinting

## Fingerprinting

- Our corpus of over two billion 802.11 frames and ~3,000 OUIs allows us to make accurate device model predictions
  - Improved granularity of MAC-based fingerprinting
  - Complexity and variety of allocation policies causes simpler fingerprinting techniques to fail
  - Resilient, other methods rely on user-configurable data