Revisiting Urban War Nibbling: Mobile Passive Discovery of Classic Bluetooth Devices Using Ubertooth One

Maxim Chernyshev, Craig Valli, and Michael Johnstone

***Abstract*— The ubiquitous nature of Bluetooth technology presents opportunities for intelligence gathering based on historical and real-time device presence data. This information can be of value to law enforcement agencies, intelligence organizations, and industry. Despite the introduction of the Bluetooth Low Energy standard that incorporates anonymity preservation mechanisms, the presence of devices that support Classic Bluetooth that uses unique and persistent device identifiers is expected to remain significant for a number of years. The common approach to finding discoverable Classic Bluetooth devices relies on a standard inquiry process that is not truly passive. Furthermore, this approach fails to detect devices that remain undiscoverable. Ubertooth One, a low-cost open source Bluetooth development platform, can assist with overcoming this limitation in a truly passive manner, making it an attractive digital forensic instrument. Using vehicle-based sensors and parallel multi-method device discovery, we conduct a practical evaluation of Ubertooth One for passive discovery and contrast its discovery rate to the standard method. Based on 83 comparative field experiments, we show that Ubertooth One can produce forensically sound observations while able to discover up to ten times as many devices. We also show that this method can identify repeat device presence, as we observe 2370 instances of repeat observations on different days in single and multiple location scenarios. We conclude that this passive technique can complement the standard method and has the potential be used as a viable alternative.**

***Index Terms*— Bluetooth, Ubertooth, passive discovery, surveillance, forensics.**

I. Introduction

BLUETOOTH also known as IEEE 802.15.1 is a wireless personal area network (WPAN) standard for short-range communications. The standard is in use by a variety of device types including personal computers, smartphones, cars, wearables, consumer electronics and home automation products. The number of annual Bluetooth device shipments is predicted to exceed 4.6 billion by 2020 mainly driven by smartphone proliferation, but also due to the expansion of the home automation and consumer robotics industries [1].

Widespread adoption of Bluetooth especially in the context of wearables and automotive technologies presents a number of opportunities in the areas of network forensics and targeted intelligence gathering. Previous studies focusing on Bluetooth device discovery and tracking outline significant implications of the original Bluetooth Classic specification stemming from lack of privacy preservation mechanisms. Investigations into early Bluetooth surveillance systems highlighted their potential for increased physical security, device location determination and consequent human action attribution capabilities. The work also uncovered the potential for privacy compromises through the disclosure of sensitive information in reported device attributes [2], [3]. The Bluetooth threat taxonomy of Dunning highlights a number of critical security issues and links published exploits to open-source attack tools [4]. In addition to device movement capture, Bluetooth device discovery techniques can be used to identify target devices for subsequent attack and compromise [5].

Some of the limitations of the original Bluetooth standard were addressed with the introduction of the Bluetooth Core Specification version 4.0 that provided a new protocol stack known as Bluetooth Smart or Bluetooth Low Energy (BLE) aimed at low power applications [6]. The improved stack offers increased privacy through device address randomization and has been under continuous development.[1] Whilst BLE offers significant improvements over Bluetooth Classic, at the time of writing, it has not fully superseded its predecessor and is not expected to do so for a number of years. By 2021, the number of BLE devices is set to account for 27% of all Bluetooth shipments, with the rest still comprising transitional dual-mode and Classic devices.

The requirement to maintain backwards compatibility especially across peripheral devices is expected to be relevant and various Bluetooth chips will continue to ship with dual-mode support.[2] As the number of these devices continues to grow, the potential for using wireless traces generated by these devices for forensic purposes is also expected to increase. For instance, law enforcement solutions for detecting and locating stolen Wi-Fi devices have already been devised.[3] Searching for stolen Bluetooth devices can also be accomplished using the same approach. From this aspect, Classic Bluetooth devices may be considered more attractive than BLE given their proliferation in more expensive consumer electronics as opposed to lower-cost BLE-based sensors and wearables.

Manuscript received June 30, 2016; revised November 25, 2016 and February 19, 2017; accepted February 22, 2017. Date of publication March 6, 2017; date of current version April 13, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Guofei Gu.

The authors are with the Security Research Institute, Edith Cowan University, Perth, WA 6027, Australia (e-mail: m.chernyshev@ecu.edu.au; c.valli@ecu.edu.au; m.johnstone@ecu.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2017.2678463

[1] http://blog.bluetooth.com/bluetooth-technology-protecting-your-privacy/

[2] http://toshiba.semicon-storage.com/eu/product/wireless-communication/bluetooth/tc35661.html

[3] http://www.l8ntwifi.com/

Therefore, the goal of this paper is to revisit the concept of Bluetooth *War-nibbling* in a modern urban setting in the context of a realistic vehicle-based data collection scenario. Akin to Wi-Fi-based *Wardriving*, the concept is centered around spatiotemporal discovery of Bluetooth wireless devices primarily using mobile sensors [7]. In light of the discussed device shipment projections and possible forensic applications, we limit the focus of our investigation to Bluetooth Classic and take advantage of *Ubertooth One*[4] to contrast passive discovery with the traditional inquiry process. We conduct a comparison and quantify the advantage of using Ubertooth One over the traditional method, focussing on the following aspects of passive discovery:

- the ability to capture unique or potentially unique device identifiers for initial discovery and persistent tracking;
- the reliability of captured identifiers based on known error indicators; and
- the seriousness of the associated privacy leakage.

In this context, aspects such as low-level technical performance and energy consumption comparison are out of scope of our investigation. Subsequently, the contribution of our work is a practical assessment of the stated aspects of passive discovery with the aim to inform future forensic applications. We use a consistent data collection approach using parallel multi-method discovery to provide us with an extensive assessment basis.

The rest of the paper is organized as follows. Section II provides an overview of related work. Sections III and IV describe the data collection method and outline the specifics of the analyzed data sets. The analysis process and resulting findings are presented in Section V. Further interpretation of results and associated implications are discussed in Section VI. Section VII concludes the paper and outlines directions for future research.

II. Related Work

Analogous to a Wi-Fi 802.11 media access control (MAC) address, a Bluetooth interface is assigned a 48-bit Bluetooth Address (BD_ADDR) that is split into three distinct components [6]. The Non-significant Address Part (NAP) together with the Upper Address Part (UAP) carry the 24-bit portion that represents the organizationally unique identifier (OUI). In combination with the OUI, the Lower Address Part (LAP) uniquely identifies the device. For Bluetooth Classic, knowing the UAP and LAP values is sufficient to be able to initiate device connections as a prerequisite for a possible subsequent attack or directed device fingerprinting [8].

Previous work shows that consumer-grade interfaces can be used to enumerate BD_ADDR values for devices running in discoverable mode using the standard inquiry process [2], [3], [7], [9], [10]. Multi-sensor deployments can be used to observe device presence and the data from such deployments could be used to derive comprehensive models of human behavior [11]. As Bluetooth devices are usually mobile and timing can adversely affect the reliability of the discovery process, the device enumeration rate could potentially be improved by adopting a non-standard implementation or using multiple, uniquely configured interfaces [12]. However, only devices that respond to inquiry are revealed using this method possibly leaving a large portion of non-discoverable devices completely unseen.

Non-discoverable devices cannot be detected via Bluetooth inquiry and other methods need to be used. While device enumeration using BD_ADDR value brute-force search may be possible, its practical applicability is limited due to address space size and timing requirements [13]. These limitations can be mitigated through address space reduction to a set of known device vendors and corresponding OUI values. However, this reductive approach would exclude recently manufactured devices or previously unknown vendors. Other mitigation techniques include the reduction of scan timing delay and parallelization of the scanning process using multiple devices or interfaces [14].

Alternatively, passive eavesdropping can be used to discover devices communicating in a *piconet* - an ad-hoc network of two or more synchronized devices. This capability generally requires access to highly-specialized and costly commercial-grade hardware and software components[5,6] used as part of Bluetooth product development. Fortunately, practical Bluetooth security research of this nature has gradually become more affordable and accessible [15], [16]. Much cheaper components can be used for full-spectrum passive packet capture, albeit with potentially irreversible hardware modifications to donor equipment [17].

The introduction of Ubertooth One has further lowered the barrier to entry and enabled additional opportunities for research into passive enumeration of Bluetooth devices, including those that adopt non-discoverability as a potential security measure. Work involving Ubertooth One has mainly focused on targeted security assessments and usually examined the BLE stack implementation [18]–[20]. Other studies have shown that generic high-accuracy passive fingerprinting using clock skews and packet preambles is also possible [21].

In terms of passive discovery, Wi-Fi-based fingerprinting and sensitive information inference have been realized using data contained in management frame (probe request) attributes [22], [23], which can be collected easily using low-cost hardware and open-source software. However, this approach is protocol-specific and is not applicable to Bluetooth. Other protocol-agnostic methods such as physical layer-based discovery and fingerprinting may be possible for Bluetooth and other wireless protocols, but many challenges such as performance, accuracy and cost still remain unresolved [24]. To the best of our knowledge, there are no documented studies that examine practical passive Bluetooth Classic device discovery using Ubertooth One in a real-life application context.

III. Experimental Setup

Unlike previous studies that featured fixed sensors, we used vehicle-mounted sensors. This approach was selected

[4] http://ubertooth.sourceforge.net/

[5] http://www.fte.com/products/FTS4BT.aspx

[6] http://www.ellisys.com/products/bex400/

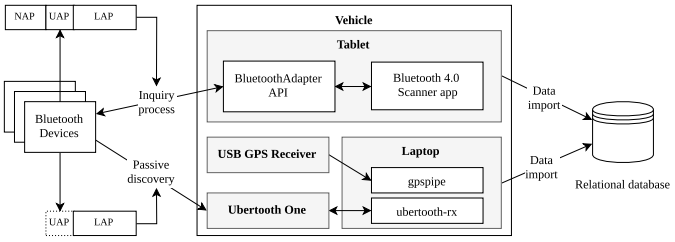


Fig. 1. Experimental setup depicting the flow of BD_ADDR components to each sensor as part of parallel multi-method device discovery using a vehicle as the sensor base. The dotted line around UAP in the context of passive discovery indicates that its capture is not guaranteed.

because it represents a realistic scenario and can inform future work around heterogenous sensor networks and wireless trace profiling in urban environments. As one possible practical forensic application, police patrol vehicles or aerial drones could be equipped with an array of multi-protocol wireless sensors including those that facilitate the discovery of Bluetooth devices. The sensing could be used to support various missions, such as:

- **Reconnaissance** - longitudinal observation of a designated urban area to capture its baseline wireless profile.
- **Targeted Search** - focussed search for a specific device or a group of devices based on known unique or potentially unique identifier or its portion.

Our experimental setup is presented in Fig. 1. We used a separate sensor for each of the discovery methods examined. Specifically, we used a consumer-grade Android tablet running a specialized Bluetooth scanning app[7] for the standard inquiry process. The app leverages the *BluetoothAdapter*[8] application programming interface (API) to manage the discovery process by triggering consecutive inquiry scans each taking approximately 12 seconds to complete. Data was subsequently extracted from the internal app database using the Android Debug Bridge (ADB).

For passive discovery with Ubertooth One, we used a consumer-grade laptop equipped with a USB-based GPS receiver running data acquisition scripts based on *ubertooth-rx*[9] and *gpspipe*,[10] with unmodified output stored in text files. This basic approach was selected because it involved no customizations to existing tools, thus, maintaining the forensic soundness of the technological artifacts. Both sensors also had locational capabilities that were leveraged further in the analysis. The collected data were imported into a relational database for subsequent pre-processing and analysis at the completion of each data collection run.

IV. Data Sets

The data collection runs included both mobile and stationary scenarios with six predetermined routes covering freeways, major and minor arterial roads and residential streets primarily during morning and afternoon peak hour traffic periods.

[7]https://play.google.com/store/apps/details?id=com.bluemotionlabs.bluescan&hl=en

[8]https://developer.android.com/reference/android/bluetooth/BluetoothAdapter.html#startDiscovery()

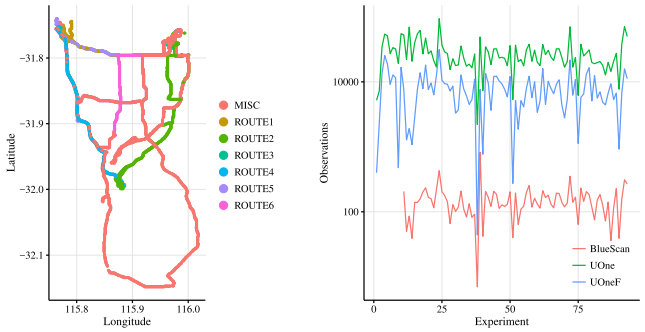
[9]https://git.io/vXzrD

[10]http://catb.org/gpsd/gpspipe.html

TABLE I
Data Set Overview

Name	BlueScanZ	BlueScan	UOne	UOneF
Days	60	45	50	50
Locations	6454	5638	34555	26621
Observations	16656	12827	2670125	769506
Devices	2681	1993	33524	31376
Local names	259	253	-	-
LAPs	-	-	31126	29178
UAPs	-	-	2398	2198
Vendors	122	124	-	-
OUIs	1302	1019	-	-

Note: Days are not always consecutive.



(a) (b)

Fig. 2. Cumulative route trips overview (2a) and observation count comparison for inquiry scan (BlueScan) and passive LAP decoding based on processed (UOneF) and unprocessed (UOne) data (2b).

In addition to the set routes, we conducted experiments alongside twelve ad-hoc routes at various times of day.

The vehicle that was driven alongside these routes covering parts of the Perth metropolitan area in Western Australia between February and April 2016. The cumulative route trips overview is presented in Fig. 2a. The routes contain common arrival or departure locations as well as overlapping sections and, thus, should not be considered completely distinct. We conducted 93 data collection rounds in total with 79 (85%) items representing set routes and 14 (15%) items representing ad-hoc routes respectively. We refer to the resulting datasets as *BlueScan* and *UOne* as presented in Table I.

A. BlueScanZ

The BlueScanZ data set represents initial exploratory activity that predates all experiments involving Ubertooth One. This data were used to assess the feasibility of the study prior to conducting the comparison. We include these data in the cumulative analysis of Bluetooth vendor and OUI distributions.

B. BlueScan

The BlueScan data set represents data collected during parallel multi-method experiments. As shown in Table I, it features less observation days than UOne because the corresponding sensor was not always active during the initial set of experiments involving Ubertooth One.

C. UOne

The UOne data set consists of LAP and UAP observations collected using the *ubertooth-rx*[11] command-line utility. Unlike the standard inquiry process, LAP and UAP discovery using this method is completely passive. As passive discovery does not facilitate NAP acquisition, the data set does not contain any vendor or OUI mappings.

D. UOneF

The observation count comparison across two sensors is presented in Fig. 2b. In addition to unique passive observations, Ubertooth One also collects inquiry process-based traffic and the volume of this traffic appears sufficient to reflect the BlueScan observations. To mitigate the impacts of this effect, we complete pre-processing steps to filter out any non-unique observations as follows.

Packets generated by the sensor that initiates inquiry scans are eliminated using the reserved LAP value of 0x9E8B33. Packets generated by discoverable devices responding to these scans are removed on the basis of BlueScan BD_ADDR to UOne LAP value matching. Any additional packets collected by Ubertooth One outside of the BlueScan sensor activity period are also removed. Subsequently, we reduce the size of the UOne data set by more than 70% (with respect to observation counts) and refer to it as *UOneF*. Despite the significant reduction in the volume of data, visual similarity between respective lines still remains. We discuss our hypotheses behind this occurrence later in Section VI.

We also need to clarify certain aspects of Ubertooth One operation to introduce the *UOneF* data set. The Ubertooth interface presents a number of attributes alongside each demodulated packet including the computed error value. This value is reflective of packet bit error rate and is used to determine whether the decoded LAP value is genuine [25]. The associated *max_ac_errors* threshold (0-4) can be used to instruct the interface to discard any packets with error value above that threshold. By default, the threshold is set to 2 and values above the default are not recommended due to significant risk of false positives. To perform additional comparison based on packets with the lowest bit error rate, we can remove all packets where error value is greater than 0. We examine the potential impact of error value filtering at the end of the next section.

V. Analysis

In this section, we present the comparative analysis of observations collected by each sensor.

A. Device Vendor Context

We combine the BlueScanZ and BlueScan datasets to obtain an indicative view of device vendor distribution, as shown in Fig. 3. We observe 150 distinct vendors with 2343 associated OUI values, as identified during the inquiry process. More than 35% of devices could not be matched to a vendor using existing industry-standard OUI databases[12,13] and we discuss possible reasons behind this finding in Section VI. Significant presence of major consumer electronics vendors such as Apple and Samsung is not surprising. These two vendors alongside Nokia account for 11% of all observed OUIs.

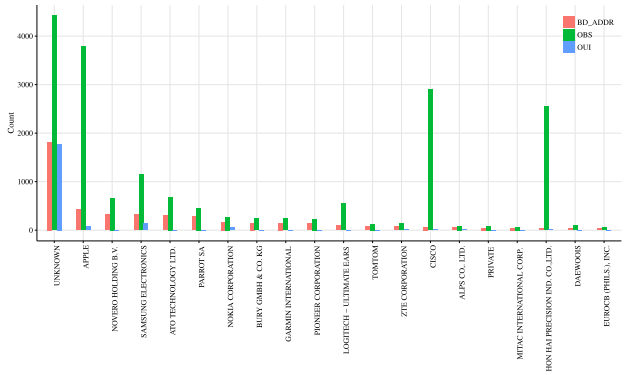


Fig. 3. Top 20 Bluetooth vendors and associated observation and OUI counts based on the number of unique BD_ADDR values in BlueScanZero and BlueScan data sets.

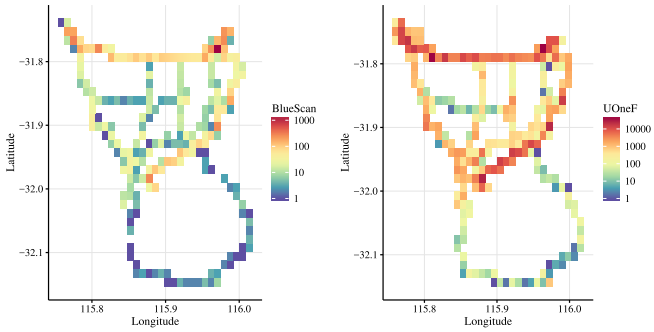


Fig. 4. 2D heat maps showing observation density based on BlueScan (4a) and UOneF (4b) data.

However, the list also includes a number of automotive wireless communication technology vendors such as Novero and Bury, as well as common makers of car audio and Global Navigation Satellite System devices. This mix is reflective of the data collection context and points towards the wide variety of device types that can be discovered using the inquiry process in a typical Australian urban environment.

The associated observation counts are not always representative of device counts. For instance, the observation count for Cisco is comparable to that of Apple, despite including a significantly smaller number of unique devices. We also observe 47 devices from privately registered vendors that could be using this registration method as an additional protection mechanism.

B. Geographic Overview

To examine the potential advantages of Ubertooth One for passive discovery, we begin by comparing the associated observation densities in a geographical context. We present the respective heat maps in Fig. 4. Whilst the heat map

[11] https://git.io/vw9k8

[12] http://standards-oui.ieee.org/oui.txt

[13] https://code.wireshark.org/review/gitweb?p=wireshark.git;a=blob_plain;f=manuf

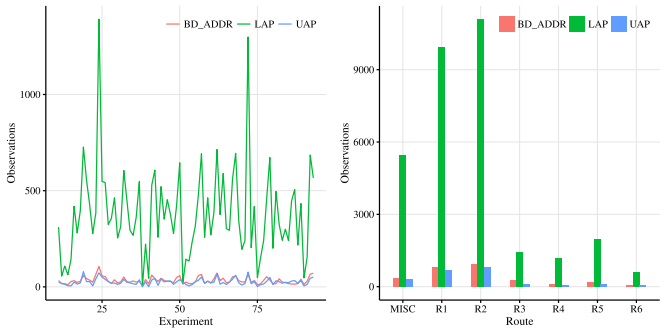


Fig. 5. Unique value count comparison between BlueScan BD_ADDR and UOneF LAP and UAP values by experiment (5a) and associated route (5b).

scales are of different magnitudes, the outlines are isomorphic. We therefore conclude that there are no significant differences between the geographic coverage of each method. The southern segment that makes part of a section of the *MISC* route group has a significantly lower volume of observations due to being an ad-hoc route. Conversely, The central and northern segments are associated with pre-determined routes and contain areas with the largest volume of observations.

C. Unique Device Discovery

Despite similar geographic coverage, Ubertooth One is able to capture significantly more unique BD_ADDR components (LAP and UAP values). The comparison between the filtered results for 83 trips that include parallel data collection are presented in Fig. 5. There are relatively strong indications of a positive linear relationship between the number of inquiry responses INQ and the number of unique LAP ($r = 0.83$) and UAP ($r = 0.80$) values. This finding is consistent with our previous observation based on Fig. 2b. The relationship may suggest that the quantity of discoverable devices in any given urban area could potentially be representative of the volume of non-discoverable devices that can be identified with passive discovery.

As shown in Fig. 5a, in the context of individual experiments, the number of UAP observations in some cases matches or even exceeds the corresponding number of devices identified using the inquiry process. Since knowing the correct NAP value is not required for obtaining device names, identified UAP and LAP combinations can be used as a basis for extracting this information from devices that are not running in discoverable mode. In our context, this approach would have the potential to identify more devices than the standard method applied in some cases.

We also aggregate comparison results by the route taken and present our findings in Fig. 5b and Table II. We observe that Ubertooth One is able to discover between five and fifteen times more unique LAP values. While LAP values alone are not sufficient for further device interaction, they may still present a potential tracking identifier, as discussed later in this section. Ubertooth One is also able to discover a comparable volume of additional LAP and UAP combinations than the standard method (up to 87% in the case of R2) entirely using passive monitoring.

TABLE II
INQUIRY PROCESS AND PASSIVE DISCOVERY RESULTS COMPARISON

Route	Trips	$\bar{t}$	INQ	LAP (INQ%)	UAP (INQ%)
MISC	10	60	358	5466 (1527%)	308 (86%)
R1	32	35	820	9916 (1209%)	676 (82%)
R2	20	50	935	11092 (1186%)	812 (87%)
R3	14	35	263	1433 (545%)	110 (42%)
R4	3	28	116	1173 (1011%)	63 (54%)
R5	2	137	177	1957 (1106%)	124 (70%)
R6	2	41	58	594 (1024%)	45 (78%)

Note: $\bar{t}$ represents mean route traversal time in minutes.

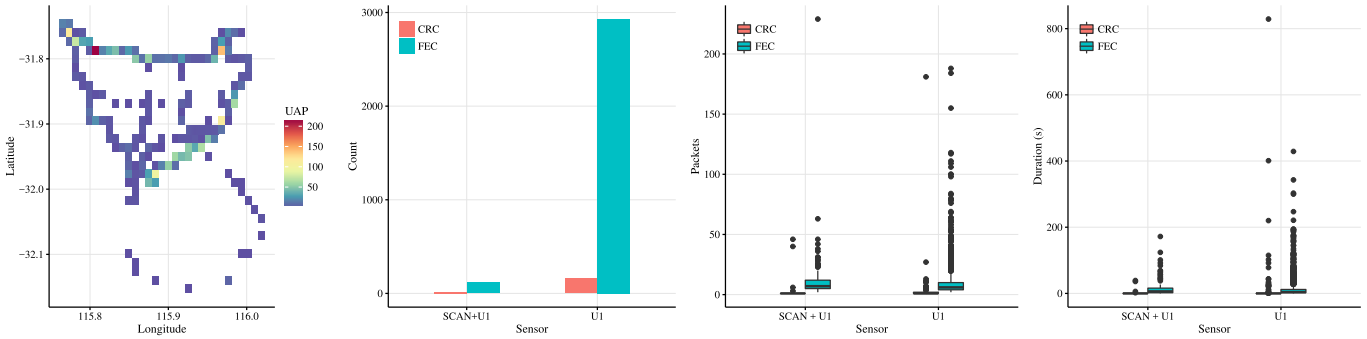
D. UAP Discovery

UAP value identification deserves specific attention, because it provides the missing address component required for initiating connections to devices running in the connectable state. These values are not transmitted in the clear and UAP discovery requires specialized techniques. When a packet contains a payload with a Cyclic Redundancy Check (CRC), CRC matching can be used to compute the value [15]. In practice, most observed packets do not include the CRC and UAP computation can be facilitated using the Forward Error Correction (FEC) method that relies on packet type knowledge [16]. Due to difficulties around determining the packet type correctly as well as the inherent clock drift, UAP discovery is prone to "inconclusive" results that have the potential to return false positives and negatives, resulting in limited forensic value [26].

Despite the high volume of successfully decoded LAP packets, we observe that the associated UAP values are only obtained for approximately 7% of all identified LAP values (see Table I). We present the UAP component discovery analysis in Fig. 6. As shown in 6a, UAP discovery is not contextualized to a particular geographical area. Observation density is non-uniform mostly represented by segments containing 50 or less events. In 6b we confirm that packets that carry a CRC are quite rare. In the case of Ubertooth-specific observations, these packets account for only 5% of all UAP discovery events. Packets elicited using the inquiry process and also captured by Ubertooth One (based on partial LAP and UAP match in the BlueScan data set) make up 4% of UAP observations, which means that most of the UAP values are computed on the basis of piconet traffic not related to our parallel inquiry scans.

In 6c, 6d and Table III we show that CRC-based UAP discovery is usually swifter and requires a smaller number of packets. This is not surprising as the CRC-based method is straightforward and highly reliable. However, in one case, 181 packets were observed before a packet with a CRC was encountered. UAP discovery can take more than 13 minutes when packets are not being transmitted frequently.

Although FEC-based discovery usually takes longer and requires a greater number of packets to be successful, we can



(a) (b) (c) (d)

Fig. 6. 2D heat map showing observation density for inquiry process-assisted (SCAN+U1) and Ubertooth One-specific (U1) UAP values (6a). Bar chart showing UAP observation count for each discovery technique - Forward Error Correction (FEC) and Cyclic Redundancy Check (CRC) (6b). Box plots showing of the number of packets (6c) and UAP discovery duration (6d) for each observation category and discovery technique.

TABLE III
UAP DISCOVERY STATISTICS

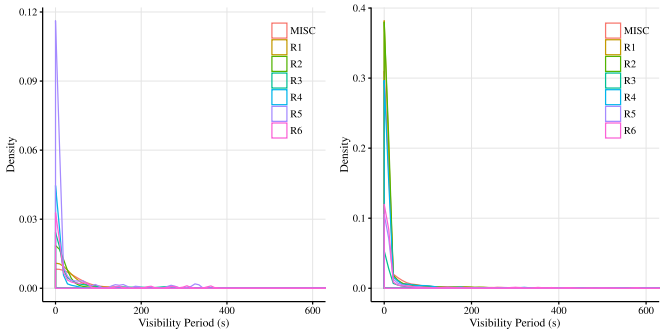
Sensor	Type	Min	Median	Max	Mean
Prior Packet Count					
SCAN + U1	CRC	1	1	46	-
SCAN + U1	FEC	2	7	229	-
U1	CRC	1	1	181	-
U1	FEC	2	6	188	-
UAP Discovery Duration (s)					
SCAN + U1	CRC	0	0	39	4.638
SCAN + U1	FEC	0	6	172	15.63
U1	CRC	0	0	829	13.71
U1	FEC	0	4	429	12.27

see cases where the process completes in less than one second using a relatively small number of packets (between two to six). Thus, the success of existing passive UAP discovery techniques is highly context-dependent. In [26], Ossmann suggested that further improvements could be made to this process and it would be beneficial to evaluate the success rate of the FEC-based method after implementing these changes.

E. Device Visibility

We also analyze the device visibility period to examine how long devices stay within sensor range. We use a moving visibility period window of 20 minutes to determine whether the device is still in range. Rolling visibility period values are added to the results set when device is not observed for more than 20 minutes - an approach similar to that in use be a commercial visitor analytics platform. We present our findings in Fig. 7 and Table IV. We observe that a significant portion of devices seen by both sensors only appear momentarily. The median values suggest that Ubertooth One has the capability to detect short-session device transmissions for up to two seconds. This interval is likely to be too short to carry out any targeted attacks or directed fingerprinting but is still sufficient to capture possibly unique device identifiers that can be used for tracking.

Looking at the maximum period lengths, our original analysis included traces from our reference device - an in-dash multimedia unit in our vehicle. The associated observations



(a) (b)

Fig. 7. Device visibility period comparison between BlueScan BD_ADDR (INQ) (7a) and UOneF LAP observations based on travsersed route (7b).

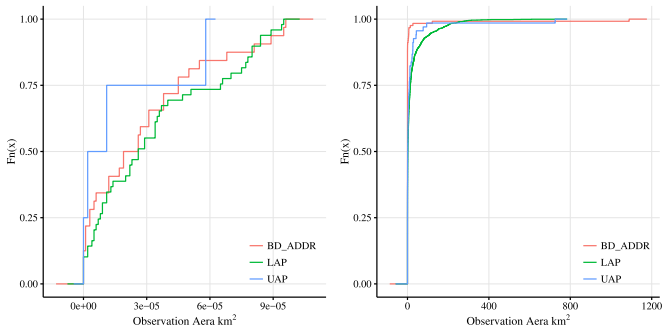
TABLE IV
DEVICE VISIBILITY STATISTICS

Route	Min		Median		Mean		Max	
	INQ	LAP	INQ	LAP	INQ	LAP	INQ	LAP
MISC	0	0	0	1	42.23	32.55	1118	5293
R1	0	0	0	1	37.34	29.11	1111	2943
R2	0	0	0	2	17.07	27.69	976	3156
R3	0	0	0	2	170.49	63.19	4751	3005
R4	0	0	0	1	3.681	15.97	142	1095
R5	0	0	0	2	37.46	43.02	1008	2029
R6	0	0	0	2	31.18	59.76	360	1904

Note: Zero (0) represents a visibility period shorter than 1 second.

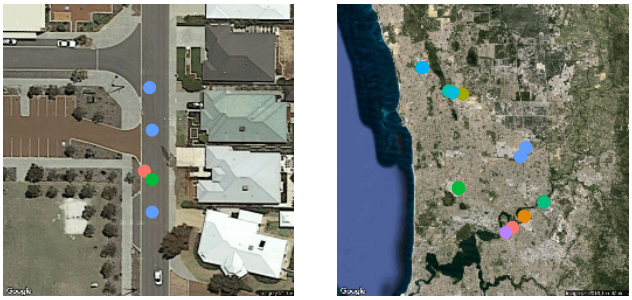
were skewing the maximum values because that device was being observed by both sensors at the same time for more than 2 hours consecutively in some cases. Surprisingly, we also saw that maximum values were significantly different across sensors with UOneF data set containing shorter periods for our reference device. This could mean that due to channel hopping Ubertooth One does not always happen to observe inquiry responses but that would also be unlikely that it happens consistently over a 20 minute period.

Subsequently, we removed period values associated with the reference device based on BD_ADDR and LAP filtering. Further analysis shows that some of the maximum period



(a) (b)

Fig. 8. Recurring observation bounding box area comparison between based on discovered address component for areas up to 100m^2 (8a) and over (8b). Only devices that were observed across multiple trips are included.



(a) (b)

Fig. 9. Example satellite views of the area containing device repeat observations for a potentially stationary device (9a) and a device observed across multiple city locations (9b).

values are associated with devices carried by the researcher at the time of data collection (a laptop and a smartphone). However, none of these devices were running in discoverable mode, and yet could still be observed passively in a consistent manner.

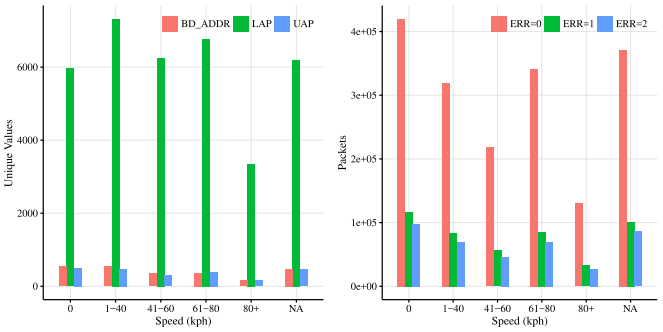
F. Device Tracking

To examine the potential of passive LAP and UAP discovery for persistent tracking, we determine the area of the bounding box that contains all collected locations for each distinct BD_ADDR value or LAP and UAP component for devices that have been observed during more than one trip. We identify a total of 2526 devices that were observed across multiple trips of which 156 (6%) and 2370 (94%) were identified by using the standard inquiry process and Ubertooth One respectively. In the latter case, 72 devices also had their UAP values discovered and repeat observations were noted on the basis of these values. By comparison, device tracking using the standard method in the context of our experiment involving non-stationary sensor and moving targets represents only a small fraction of recurring devices.

We also present the device observation area analysis in Fig. 8. We see that 85 (3%) devices were repeatedly observed within an area of up to 100m^2 meaning that these devices are most likely stationary. We use this area threshold as a representative value of the potential theoretical discovery range. Furthermore, 1392 (55%) of devices were observed across an area of more than 1km^2 and we believe that these observations represent devices that were seen at various places during different trips. Fig. 9 presents two examples corresponding to each of the described scenarios. In the example case shown in 9b, the device was observed during 12 trips and across 6 routes, with the associated bounding box area of approximately 409km^2. Assuming persistence of LAP identifiers throughout device lifespan, our observations show that persistent tracking on the basis of partial address components is possible, albeit in a limited number of cases.

G. Movement Impact

Given that data collection took place on the road, we were interested to examine the potential effect of vehicle movement by looking at how vehicle speed reported by the GPS receiver



(a) (b)

Fig. 10. Bar charts showing speed value categories and the associated unique address component value count (10a) and number of packets collected by Ubertooth One based on reported error value (10b).

could impact the discovery process. We introduce six distinct categories four of which denote typical speed restriction zones in Western Australia, with the remaining two representing cases where our vehicle is not moving (speed is zero) or when no speed information is available. We expected to see differences indicating signal loss at higher speeds, for example, in situations where a packet is sent by a device but the sensor has already left the range. This scenario could also realistically occur when our vehicle and a device contained in another vehicle move in opposite directions.

While the number of devices observed when travelling at speeds over 80km/h is considerably lower, it is also representative of the fact that most travel occurred below this speed. There is no significant difference between the speeds of 1-40km/h (acceleration and braking) and 41-60km/h (cruise speed in populated areas). However, an accurate comparison is not feasible because the data collection did not run for equal amounts of time for each category. As a general observation, it is evident that Ubertooth One can be used for passive discovery in both stationary and moving sensor scenarios including minor, major roads and freeways.

We also examine the impact of speed upon the reported Ubertooth One error value, as previously discussed in Section IV-D. Noting that Fig. (10b) presents packet count (as opposed to unique devices) it is also evident that stationary data collection produces more packets, but that does not

TABLE V
ERROR VALUE PORTION BY SPEED CATEGORY

Speed (kph)	0	1-40	41-60	61-80	80+	NA
ERR=0	66%	68%	68%	69%	69%	66%
ERR=1	18%	18%	18%	17%	17%	19%
ERR=2	16%	14%	14%	14%	14%	15%

necessarily translate into a higher number of unique devices. The error value portions for each category are shown in Table V. The distribution is consistent and we observe that packets that carry the lowest bit error rate are expected to account for approximately 68% of all observations.

H. Ubertooth One Error Value

Following on from our analysis of speed impact, we examine the error value as a possible measure of perceived data reliability. While 68% of decoded packets are associated with the lowest bit error, observations with higher error rates may also carry some overlap with this portion. In the UOne data set, only 6193 (0.002%) LAP observations with error value of 1 do not have a matching LAP observation with the error value of 0. The number of likely unreliable LAP observations with the error value of 2 accounts for 9765 (0.004%) of all LAP observations. Therefore, applying strict filtering on the basis of error value of 0 would not result in a significant error reduction but could lower unique device counts. Notwithstanding the fact that error value determination results can be inconclusive, such low overall percentage of likely unreliable data can be attributed to the fact that piconets generate packets frequently and encountering a packet with a low bit error rate is common even under noisy conditions. This high reliability of captured LAP values can be used to strengthen the argument around their validity for forensic purposes.

I. LAP Values of Interest

We observed two LAP values that merit further attention. First, we collected 373 observations of LAP 0x000000, which could be assigned to an initial prototyping or test device. Second, we observed a LAP value of 0x9E8B3F on 4 occasions. This value is part of the assigned value range for Baseband[14] and at the time of writing appears as reserved for future use. Our finding could indicate that this reservation status is outdated, or that we came across a possibly non-standard implementation. However, as Bluetooth addresses can easily be spoofed,[15] these values could also be injected artificially.

J. Privacy Leakage Estimation

Given that passive discovery and inquiry process reveal different attributes, we conduct a comparison of perceived privacy leakage between two methods. To quantify this leakage, we adopt the concept of a *privacy unit* [27], which refers to portions of data that include private or sensitive information.

[14]https://www.bluetooth.com/specifications/assigned-numbers/baseband
[15]http://tools.kali.org/wireless-attacks/spooftooph

TABLE VI
PRIVACY UNITS BY TYPE AND DISCOVERY METHOD

Category	Unit	Inquiry process	Passive discovery
Identity	Name	Yes	No
Location	Sensor coordinates	Yes	Yes
Device	Unique identifier	Yes	Yes
Device	Type	Yes	No
Device	Vendor	Yes	No

TABLE VII
$S(P, L)$ STATISTICS

Unit	Min		Median		Mean		Max	
	INQ	LAP	INQ	LAP	INQ	LAP	INQ	LAP
Name	0	-	0	-	0.01	-	1	-
Location	0.05	0.05	0.1	0.1	0.1	0.1	0.45	2.85
Device	1.05	0.8	1.05	0.8	1.06	0.8	1.3	1
Combined	1.1	0.85	1.15	0.9	1.18	0.91	2.35	3.85

The original classification categorizes these units into user privacy (such as name and location) and infrastructure privacy (such as device identifier and type). Both are applicable in the context of our analysis. The breakdown of privacy units potentially available via each method is presented in Table VI.

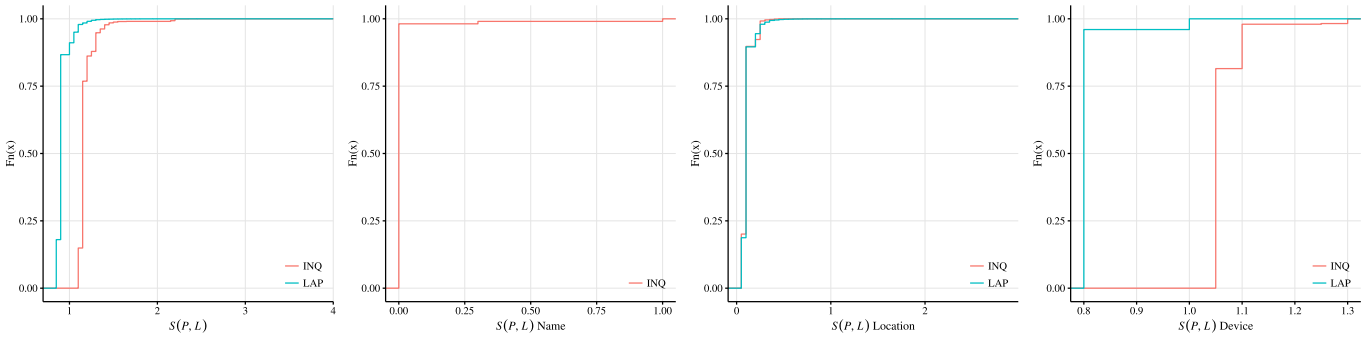
In the case of inquiry process, user and infrastructure units can be extracted from obtained device friendly names, such as *John Doe's Galaxy S5*. However, certain names may only carry a partial unit or a unit subset, such as *John's Phone* or simply *Galaxy S5*. In some cases, these names also contain what appears to be a serial number of the device, for example *nuvi #1234567890*, which can be used to infer both device type (navigation device) and identifier (likely, unique serial number). Additional attributes such as device major and minor class can be extracted using the extended inquiry process, but are not incorporated into our analysis, as this data were not collected during our experiments.

To quantify the privacy leakage for Bluetooth Classic, we adopt the seriousness of privacy leakage concept, as described in [27]. The calculation is performed using the following formula:

$$S(P, L) = \sum w_i \cdot p_i \cdot l_i \tag{1}$$

The seriousness of privacy leakage score $S(P, L)$ is the sum of leaked units multiplied by their corresponding percentage weights that are assigned by the user. In this model, $l_i = 1$ when privacy unit p_i is leaked, and $l_i = 0$ otherwise. For a detailed description of the model please refer to the original work. We also recognize that privacy units can be made up of multiple components (for example, identity comprising first name and last name) and, thus, assign weights to individual components as shown in Table VIII. The weight assignment is performed by the authors taking into account the potential forensic value of each privacy unit and the following assumptions:

1) **Name:** last name leakage is assumed to be more serious than first name due to greater perceived uniqueness. The



(a) (b) (c) (d)

Fig. 11. Cumulative distribution function of privacy leakage estimation score comparing inquiry process and passive discovery in terms of the combined (11a), name (11b), location (11c), and device (11d) privacy leakage.

TABLE VIII
PRIVACY UNIT COMPONENT WEIGHTS

Unit	Component	Weight
Name	First name	30%
Name	Last name	70%
Location	Single trace	5%
Location	Trace cluster	10%
Location	Trace cluster (multiple days)	25%
Device	LAP	80%
Device	UAP (or complete BD_ADDR)	20%
Device	Type	5%
Device	Vendor	5%
Device	Serial number	20%

full name leakage produces the score of 1 for the name unit.

2) **Location:** location leakage is assumed to be more serious when associated device traces contain multiple location clusters that contain observations captured on more than one day. As [28] suggest that as little as four unique spatiotemporal points are required to reveal unique individual movement patterns, four clusters with observations over multiple days will produce the score of 1 for the location unit. More clusters can yield a higher score.
3) **Device:** the seriousness of leakage is based on the portion of leaked BD_ADDR value and its suitability for subsequent tracking. Based on the combined set of collected unique BD_ADDR values we determine that only a single device with LAP 0x00000 was associated with multiple distinct BD_ADDR values. This finding means that over 99% of LAPs could potentially be unique and suitable for persistent passive device tracking. Having access to the UAP component increases the seriousness as it allows a potential attacker to initiate device connections. Therefore, the combined LAP and UAP leakage produces the score of 1 for the device unit. Other components such as device type and vendor are considered auxiliary and carry minor additional weight, with the exception of the serial number.

The analysis of friendly device names to detect leakage of identity and applicable device units was performed by the authors manually. Location clustering was performed using the DBSCAN algorithm [29] with the value of ϵ set to 1.5 km.

The comparisons of the combined $S(P, L)$ score as well as the corresponding scores for each privacy unit are presented in Fig. 11 and Table VII. It is not surprising to see that information disclosure via device friendly names results in more serious privacy leakage for the inquiry process. However, location privacy leakage that enables device tracking for both methods is comparable, with passive discovery showing a significantly greater maximum score. Location traces could be used to mount additional inference attacks, such identifying possible device to business or place of residency associations using the approach similar to that described in [30], which is based on reverse goecoding and proximity-based searching of publicly available directory services. While our findings once again highlight the previously raised privacy concerns associated with the original Bluetooth stack, they also confirm the potential of passive discovery for tracking device that adopt non-discoverability as the possible mitigating measure.

K. Sensor Range Difference Implications

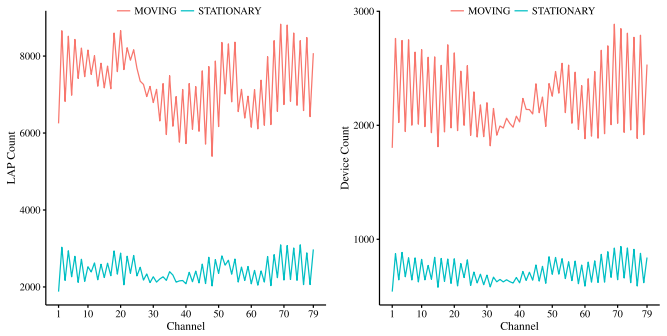
One possible reason for a tenfold increase in the number of devices discovered by Ubertooth One could be attributed to its likely higher sensitivity and, subsequently, longer range. Ubertooth One is positioned as a Bluetooth power Class 1 *comparable* device,[16] that may have an intended range of up to 100 meters under ideal conditions. Consumer-grade smartphones and tablets are generally considered to include Bluetooth power Class 2 interfaces (intended range of up to 10 meters), primarily due to potential battery drain considerations. However, it is impractical to assume the declared class ranges to be representative of real-life performance, especially in a densely populated complex urban environment with many stationary and moving objects and obstacles. Our preliminary field experiments showed that the tablet devices used as the inquiry process sensor was capable of detecting devices just over 50 meters away, which is significantly higher than the stated range of a class 2 device. We apply a set filtering threshold to quantify the impact of filtering LAP and UAP observations based on a set RSSI value. Assuming 50 meters

[16] http://ubertooth.sourceforge.net/hardware/one/

TABLE IX

FILTERED INQUIRY PROCESS AND PASSIVE DISCOVERY RESULTS COMPARISON (UBERTOOTH ONE RSSI $\geq -78dBm$)

Route	Trips	$\bar{t}$	INQ	LAP (INQ%)	UAP (INQ%)
MISC	10	60	358	4046 (1130%)	251 (70%)
R1	32	35	820	7785 (949%)	545 (66%)
R2	20	50	935	9152 (979%)	686 (73%)
R3	14	35	263	997 (379%)	93 (35%)
R4	3	28	116	867 (747%)	52 (45%)
R5	2	137	177	1366 (772%)	92 (52%)
R6	2	41	58	471 (812%)	42 (72%)



(a) (b)

Fig. 12. Line charts showing the number of LAPs (12a) and the number of unique devices (12b) collected on each channel when moving or stationary (speed = 0).

to be realistic range for both sensors in the context of our investigation, we conducted field experiments and determined that Ubertooth One reported the maximum RSSI value of -78dBm for packets captured at this distance. Subsequently, we filter out all Ubertooth One observations with RSSI below this value, as presented in Table IX. Filtering reduces the number of discovered unique address components, on average, by 25%, with the overall number of additional devices discovered by Ubertooth One being eight times greater.

L. Ubertooth One Channel Hopping

We examined the impact of the channel hopping scheme implemented in the standard Ubertooth firmware upon its ability to capture packets across all 79 channels. As shown in Fig. 12, we inspected both moving and stationary scenarios. There is no direct relationships between the number of LAPs collected on a particular channel and the number of unique devices discovered (for instance, refer to channels 30 to 40). However, our findings indicate that the standard implementation is likely suboptimal, because even channels generally yield a higher number of observations. This outcome is consistent with the issues presented in [31]. Poor firmware optimization was discovered to cause resource contention between channel hopping and host data streaming, which could cause hopping delays and result in non-uniform channel coverage.

VI. DISCUSSION

While it was reasonable to assume that a specialized Bluetooth development platform would yield a higher discovery rate due to its advanced capabilities, we did not expect a tenfold increase in the number of unique address component observations. Our findings suggest that previous work relying only on discovery using the inquiry process may have developed conclusions based on a limited view of surrounding devices. In the context of traffic analysis, this limitation can be overcome by using additional data sources such as traffic count tables [32]. Passive discovery can, therefore, also be used to complement the original method as it provides much greater device visibility as an additional data source. A passive technique is also attractive from a digital forensic standpoint as it does not require device interaction. Finally, such passive technique is associated with less serious privacy leakage, and could be considered as a better alternative for retail analytics solutions.

One limitation of our work lies in the assumption that Classic Bluetooth BD_ADDR values are unique. The so-called Bluetooth *anonymity mode*, as expected as part of the specification version 1.2 was meant to mitigate device tracking using persistent identifiers. However, we were not able to locate evidence of this mode being implemented as part of the Classic Bluetooth stack. Specific provisions were described in [33]. Following on, [34] suggest that address randomization for Classic Bluetooth would not be implemented as it would adversely affect existing implementations. It has been suggested that identical BD_ADDR values can be shared by different devices [35], such as fleet vehicles. At the same time, the authors describe the volume of duplicate addresses as "negligible" and warn of the potential increase in the uptake of address duplication practices.

While we can assume address uniqueness with a high degree of confidence, assuming that identical LAP values observed on different trips belong to the same device can be dangerous. There could be cases where the same LAP is used by two or more devices from different vendors. But our inspection of the BlueScan data set did not reveal any such cases. At this stage, we do not offer a mechanism for quantifying LAP value reliability for tracking purposes but we theorise that previously described duplicate address elimination and trip detection approaches could be examined for this purpose. In practice, any such mechanism would likely require a widespread sensor network.

Another limitation lies in the sensor configuration used. First, discovery using an Android app with a single interface is not optimal in contrast to a non-standard custom implementation with multiple interfaces. As the result, the number of reported INQ observations could be significantly lower than the actual number of discoverable devices present, possibly overstating the benefit of passive discovery. Second, sensor range differences could be attributes to such a significant difference in the number of unique devices observations, however filtering based on experimentally determined RSSI values does not result in drastic reduction. Third, the firmware that is shipped with Ubertooth One has been described as “poorly optimized for real-time frequency hopping” and two interfaces with custom firmware and a controller node would be required to address these issues [31]. Even in light of these limitations, we believe that a single-interface sensor can

greatly complement the standard inquiry process or replace it in situations where revealing sensor presence is not desirable.

VII. Conclusion

We examined passive device discovery using Ubertooth One and contrasted the outcomes with the standard inquiry process for Classic Bluetooth. We confirmed that Ubertooth One yields a much higher device discovery rate and produces partial identifiers that are 1) forensically sound, 2) most likely unique, and 3) repeatedly trackable, despite short device visibility periods. We would like to highlight the following additional observations:

- Device population comes from a rich variety of vendors, but vendor identification based on OUI values is not generally successful
- UAP discovery is opportunistic and context-dependent
- Standard Ubertooth One firmware has limitations that can result in non-uniform spectrum channel coverage
- Despite not having access to additional device attributes (such as device friendly name or full BD_ADRR value), passive discovery can lead to location privacy leakage

Mitigating the risk of privacy leakage would require modifications to the Classic Bluetooth stack. However, we view such modifications as unlikely, given the number of devices in existence, as well as other implementation constraints. Therefore, passive discovery is expected to have practical applicability until BLE completely supersedes the original stack. In future, we plan to examine possible approaches to assist with quantifying LAP value uniqueness for tracking purposes. We also wish to examine the potential for passive discovery in the context of BLE and repeat our experiments in other geographic contexts.

Acknowledgment

The authors would like to thank the creators of Ubertooth One for enabling low-cost passive Bluetooth device discovery. Special thanks also to Dominic Spill for answering our questions and valuable suggestions.

References

[1] "Market data for Bluetooth," ABI Research, Oyster Bay, NY, USA, Tech. Rep. MD-BLTH-169, 2016.

[2] M. Haase and M. Handy, "BlueTrack—Imperceptible tracking of Bluetooth devices," in *Proc. 6th Int. Conf. Ubiquitous Comput. (UbiComp)*. 2004, pp. 1–2.

[3] P. Jappinen, I. Laakkonen, V. Latva, and A. Hamalainen, "Bluetooth device surveillance and its implications," *WSEAS Trans. Inf. Sci. Appl.*, vol. 1, no. 4, pp. 1056–1060, 2004.

[4] J. P. Dunning, " Taming the blue beast: A survey of Bluetooth based threats," *IEEE Security Privacy*, vol. 8, no. 2, pp. 20–27, Mar./Apr. 2010.

[5] L. Carettoni, C. Merloni, and S. Zanero, "Studying Bluetooth malware propagation: The BlueBag project," *IEEE Security Privacy*, vol. 5, no. 2, pp. 17–25, Feb. 2007.

[6] "Bluetooth core specification version 4.0," Bluetooth Special Interest Group, Kirkland, DC, USA, Tech. Rep., 2010.

[7] "War nibbling: Bluetooth insecurity," O. Whitehouse, Cambridge, MA, USA, Tech. Rep., 2003.

[8] M. Herfurt and C. Mulliner, "Remote device identification based on Bluetooth fingerprinting techniques," Trifinite Group, Tech. Rep. Version 0.3, 2004.

[9] M. Pels, J. Barhorst, M. Michels, R. Hobo, and J. Barendse, "Tracking people using Bluetooth: Implications of enabling Bluetooth discoverable mode," Ph.D. dissertation, Univ. Amsterdam, Amsterdam, The Netherland, 2005.

[10] A. Solon, M. Callaghan, J. Harkin, and T. McGinnity, "Case study on the Bluetooth vulnerabilities in mobile devices," *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 6, no. 4, pp. 125–129, 2006.

[11] L. Caldwell, S. Ekerfelt, A. Hornung, and J. Y. Wu, "The art of bluedentistry: Current security and privacy issues with Bluetooth devices," 2006.

[12] A. Franssens, "Impact of multiple inquires on the Bluetooth discovery process and its application to localization," M.S. thesis, Univ. Twente, Enschede, The Netherlands, 2010.

[13] K. Haataja, "Two practical attacks against Bluetooth security using new enhanced implementations of security analysis tools," in *Proc. IASTED Int. Conf. Commun. Netw. Inf. Secur. (CNIS)*, Phoenix, AZ, USA, 2005, pp. 13–18.

[14] D. Cross, J. Hoeckle, M. Lavine, J. Rubin, and K. Snow, *Detecting Non-Discoverable Bluetooth Devices*. Boston, MA, USA: Springer, 2008, pp. 281–293.

[15] D. Spill and A. Bittau, "BlueSniff: Eve meets Alice and Bluetooth," in *Proc. USENIX Workshop Offensive Technol. (WOOT)*, 2007, pp. 1–10.

[16] M. Ossmann and D. Spill, "Building an all-channel Bluetooth monitor," in *Proc. ShmooCon*, 2009.

[17] J. Cache, J. Wright, V. Liu, E. Scott, B. Antoniewiecz, and C. Wang, *Hacking Exposed Wireless*. New York, NY, USA: McGraw-Hill, 2010.

[18] M. Ryan, "Bluetooth: With low energy comes low security," in *Proc. USENIX Workshop Offensive Technol. (WOOT)*, 2013, p. 4.

[19] M. L. Hale, D. Ellis, R. Gamble, C. Waler, and J. Lin, "Secu Wear: An open source, multi-component hardware/software platform for exploring wearable security," in *Proc. IEEE Int. Conf. Mobile Services (MS)*, Jun. 2015, pp. 97–104.

[20] W. K. Zegeye, "Exploiting Bluetooth low energy pairing vulnerability in telemedicine," in *Proc. Int. Telemeter. Conf.*, 2015, pp. 1–10.

[21] J. Huang, W. Albazrqaoe, and G. Xing, "BlueID: A practical system for Bluetooth device identification," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Aug. 2014, pp. 2849–2857.

[22] M. Cunche, M. A. Kaafar, and R. Boreli, "I know who you will meet this evening! Linking wireless devices using Wi-Fi probe requests," in *Proc. IEEE Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2012, pp. 1–9.

[23] A. D. Luzio, A. Mei, and J. Stefa, "Mind your probes: De-anonymization of large crowds through smartphone WiFi probe requests," in *Proc. IEEE INFOCOM 35th Annu. IEEE Int. Conf. Comput. Commun.*, Apr. 2016, pp. 1–9.

[24] B. Danev, D. Zanetti, and S. Capkun, "On physical-layer identification of wireless devices," *ACM Comput. Surv. (CSUR)*, vol. 45, no. 1, p. 6, 2012.

[25] D. Spill, "Bluetooth packet sniffing using project Ubertooth," in *Proc. Ruxcon*, 2012.

[26] M. Ossmann, *Discovering the Bluetooth UAP*, accessed on Nov. 19, 2015. [Online]. Available: http://ubertooth.blogspot.com.au/2014/06/discovering-bluetooth-uap.html

[27] N. Cheng, X. O. Wang, W. Cheng, P. Mohapatra, and A. Seneviratne, "Characterizing privacy leakage of public WiFi networks for users on travel," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2769–2777.

[28] Y.-A. De Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel, "Unique in the crowd: The privacy bounds of human mobility," *Sci. Rep.*, vol. 3, Mar. 2013, Art. no. 1376.

[29] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. KDD*, vol. 96. 1996, pp. 226–231.

[30] S. Seneviratne, F. Jiang, M. Cunche, and A. Seneviratne, "SSIDs in the wild: Extracting semantic information from WiFi SSIDs," in *Proc. IEEE 40th Conf. Local Comput. Netw. (LCN)*, Oct. 2015, pp. 494–497.

[31] W. Albazrqaoe, J. Huang, and G. Xing, "Practical bluetooth traffic sniffing: Systems and privacy implications," in *Proc. 14th Annu. Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, New York, NY, USA, 2016, pp. 333–345. [Online]. Available: http://doi.acm.org/10.1145/2906388.2906403

[32] G. Michau, A. Nantes, E. Chung, P. Abry, and P. Borgnat, "Retrieving trip information from a discrete detectors network : The case of Brisbane Bluetooth detectors," in *Proc. 32nd Conf. Austral. Inst. Transp. Res. (CAITR)*, Univ. New South Wales, Sydney, Australia, Feb. 2014, pp. 1–8.

[33] C. Gehrmann and K. Nyberg, "Enhancements to Bluetooth baseband security," in *Proc. Nordsec*, 2001. pp. 191–230.

[34] C. Douligeris and D. N. Serpanos, *Network Security: Current Status and Future Directions*. Hobken, NJ, USA: Wiley, 2007.

[35] A. Bhaskar *et al.*, "Is bus overrepresented in Bluetooth MAC scanner data is MAC-ID really unique?" *Int. J. Intell. Transp. Syst. Res.*, vol. 13, no. 2, pp. 119–130, 2015.

Maxim Chernyshev is currently a Researcher with the Security Research Institute, Edith Cowan University, Perth, WA, Australia. His current work focuses on digital surveillance and multiprotocol wireless device tracking methods, fingerprinting techniques and their application for digital forensics and intelligence purposes, web browsers and web applications, IoT security, security data visualization, and geoforensics.

Craig Valli is currently a Professor and the Director of the Security Research Institute, Edith Cowan University (ECU-SRI), Perth, WA, Australia. He has over 25 years of experience with the IT industry, conducts research on network security and digital forensics issues, and consults to industry. His main research focus is on securing networks and critical infrastructures, detection of network borne threats, and forensic analysis of cyber security incidents. He is the Congress Chair for the annual ECU-SRI Security Congress.

Michael Johnstone received the M.Sc. and Ph.D. degrees from Curtin University in 1997 and 2008, respectively. He has been a Contractor for private industry, and government and research organizations, and has held various roles, including programmer, systems analyst, project manager, and network manager before moving to academia. He is currently an Associate Professor with Edith Cowan University (ECU), WA, Australia, where he teaches secure programming and software engineering. He is a member of the Security Research Institute at ECU. His research interests focus on resilient systems and include secure development methodologies, wireless sensor networks, and the security of IoT devices.