Traffic Signature-based Mobile Device Location Authentication

Jack Brassil, Pratyusa K. Manadhata, Ravi Netravali†
HP Laboratories, Columbia University†
[jack.brassil, pratyusa.k.manadhata]@hp.com, ran2290@gmail.com

***Abstract*—Spontaneous and robust mobile device location authentication can be realized by supplementing existing 802.11x access points (AP) with small cells. We show that by transferring network traffic to a mobile computing device associated with a femtocell while remotely monitoring its ingress traffic activity, any internet-connected sender can verify the cooperating receiver's location. We describe a prototype non-cryptographic location authentication system we constructed, and explain how to design both voice and data transmissions with distinct, discernible traffic signatures. Using both analytical modeling and empirical results from our implementation, we demonstrate that these signatures can be reliably detected even in the presence of heavy cross-traffic introduced by other femtocell users.**

***Index Terms* – Distance bounding, GPS, location privacy, small cells, proximity testing, side channels.**

I. INTRODUCTION

Internet based Location Application Providers (LAPs) ranging from discount distributors such as *GroupOn* to geo-social services including *Foursquare* stand to benefit from authenticating the physical locations of their users. Yet few mechanisms are available to LAPs to spontaneously authenticate a client's location, particularly a new client with whom they have no pre-existing relationship, such as a consumer entering a retail shopping mall.

Mobile operators provide widely available network-based location services as well as location applications (e.g., AT&T's FamilyMap), though these services are limited to their subscribers. Authorized access to operator location services would benefit LAPs who might wish to partner with operators; however, operators currently have no straightforward means of authorizing and sharing subscriber location information with third parties while ensuring subscriber privacy. As a result inexpensive and widely deployed GPS receivers have made handset-based location service the preferred choice of LAPs. Inexpensive solutions such as QR Codes may also be used in certain scenarios, e.g., targeted retailing. But as users recognize the benefit of location authentication, the economic incentives to provide false location information are growing. Hence we anticipate that authenticating client location will become increasingly important as emerging location-driven ecosystems evolve, and that some LAPs – and their partners such as advertisers – will demand to authenticate clients to both enhance and measure service delivery quality.

Authentication is also a fundamental building block of Location-Based Access Control systems. Mobile user authentication can be used to grant limited access permissions to off-site workers and customers. Location authentication applications also arise in military settings; prior to transmission it is desirable to verify the destination of location-specific content such as maps of areas for future reconnaissance.

To address these challenges we have proposed to authenticate a mobile device's location by placing small cells (e.g., femtocells and picocells) at existing public WiFi sites [1]–[3]. The short wireless range of these basestations permits us to locate associated User Equipment (UE) to within tens of meters, and indoor operation is supported. Different applications require different authentication granularity, e.g., locating a consumer in a shopping mall vs. identifying proximity to a retailer in the mall. Hence we have focused on a relatively fine-grained yet inexpensive solution to support a wide range of applications. In this paper we show how by impressing a signature in network traffic while remotely monitoring femtocell ingress link activity, any internet-connected user can remotely verify *any* cooperating mobile party's location. Our key contributions include

1) a lightweight, non-cryptographic method of verifying a cooperating but untrusted party's location;
2) an authentication architecture requiring *no* hardware or software modifications to existing mobile handsets, operator infrastructure, or public WiFi APs;
3) a reliable means of authenticating either a voice-only phone or smartphone user's location;
4) the ability to authenticate location while keeping the located party's and the verifier's locations unknown to the location service provider; and
5) an evaluation of our approach through both analysis and empirical study of a prototype system.

The remainder of the paper is organized as follows. Section II describes our design goals, outlines our proposed authentication system architecture and operation, and describes a prototype we implemented to empirically evaluate our proposal. Section III examines the problem of designing and detecting voice-based traffic signatures in the presence of interfering cross-traffic including voice calls, text messages, and data transfers introduced by other parties sharing the femtocell. The next section presents an analytical model to evaluate the detection performance of voice signatures. Section V examines the problem of designing and detecting data-based traffic signatures, which we show to be detected quickly and easily relative to their voice counterpart. The possibility of using short messaging signals for authentication is explored in

Digital Object Indentifier 10.1109/TMC.2013.91

Section VI, and the following section discusses the advantages and disadvantages of using each traffic signature type. Security and privacy properties of the authentication scheme are studied in Section VIII, and we review related research in Section IX. The final section summarizes our contributions, and identifies several envisioned enhancements of our approach.

II. SYSTEM DESIGN, ARCHITECTURE AND OPERATION

Consider a LAP seeking to authenticate a previously unknown client's current location. Suppose that the client carries a mobile device, but the LAP has no knowledge of the device capabilities, nor relationship with the client's mobile operator. The LAP requires a *spontaneous, one-time* authentication which is ideally 1) *device-independent* – including basic phones, smartphones, and tablets, and supports 2) *multiple carrier operation* – including multiple wireless operators and devices spanning different data transmission technologies such as 3G and LTE.

The service must be reasonably trusted by the LAP, but we do not seek to create a cryptographically-strong authentication. The system need not be unbreakable by a determined adversary, but should be sufficiently *hard for the client to defeat*, as determined by a LAP's *investment*s in the transaction, e.g., a discount retail coupon's value, or an unauthorized system access's cost. Such a design consciously sacrifices authentication strength for low-cost and scalability, and is well suited for relatively low value internet transactions. As with all location-based services, security and privacy requirements are paramount. Clients should *opt-in* to each location verification, and the transaction itself should take place with a high-degree of client location *privacy*.

A. System Architecture

To realize these operational objectives we supplement existing public Wifi hotspots with off-the-shelf femtocells [4] . We rely on various femtocell properties (e.g., limited transmission range, exposed uplink, private ownership, and integrated GPS) to authenticate the location of a femtocell-associated mobile device, without requiring mobile operator involvement or any modifications to operator infrastructure or services.

Femtocells are low-power, limited range (e.g., tens of meters) wireless access points that operate in licensed spectrum to connect subscribers' mobile devices to their mobile operator's network, typically using wired public internet access as backhaul. The devices satisfy the various regulatory, compliance, and spectrum use requirements of macrocells, including supporting location service.

Residential femtocells typically support only 2-8 active mobile device associations (i.e., users), though such limits can be dictated by an assumption about the necessary available uplink bandwidth to ensure adequate quality-of-service for multiple active voice calls. Each call consumes roughly a continuous 50 kbs duplex rate, depending on the coding mechanism employed. Voice calls can originate on residential femtocells, and subsequently be handed over to cell towers as callers leave the coverage area; however, active calls originating elsewhere may not be handed to a femtocell. GPS signal availability is typically required, and can be achieved in indoor devices through cabled remote antennas. A diverse collection of larger capacity small cells are appearing in the commercial marketplace. Though we focus here on inexpensive femtocells, our results can be extended to the larger universe of small cells.

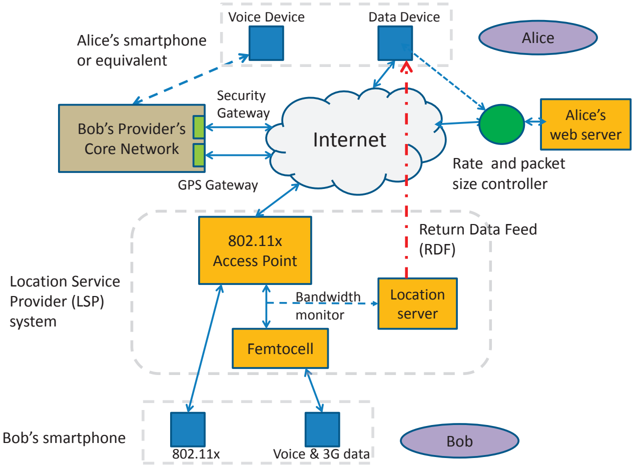


Fig. 1: Architecture of a single-carrier location authentication system using network traffic to authenticate a mobile device. A multi-carrier system would employ one femtocell for each mobile operator.

Voice and data traffic to and from the femtocell are directed to a Security Gateway (SG) at the edge of the operator's core network. Some control traffic may also be directed to other service points, such as a GPS Gateway. Voice, data, and control traffic between the mobile operator's core network and femtocell is tunneled and encrypted with protocols such as the Encapsulated Security Payload protocol [5], and transported over UDP. Hence, confidentiality is assured against exactly the passive monitoring that we will describe in the next section.

B. Participants in Location Authentication

Participating in a location authentication are:

1) *Bob* is a mobile device user whose location is to be authenticated. He is willing to cooperate with the authentication to realize some benefit but we can not trust his assertion of his location. To be located Bob requires a mobile device (e.g., voice-only or smartphone) capable of associating with a femtocell at his current location.
2) *Alice* seeks to verify Bob's present location (with his explicit approval). Alice and Bob do not need to have any pre-existing relationship; Alice could be a LAP unknown to Bob. In some applications, however, Alice and Bob may have a relationship, e.g., family member or employer, that compels his cooperation. In general, Alice will extend some benefit to Bob only after verifying his location. Alice must have the equivalent capability of a smart phone, or more precisely a (mobile or landline) voice-only phone plus minimal compute and display capability; a web browser suffices.
3) The *Location Service Provider (LSP)* seeks to provide a public-access location authentication service. The lo-

cation itself – say a coffee shop – might already offer a public WiFi service. The LSP is incented to provide location service to realize either a direct benefit (e.g., a payment from Alice for participating in a verification), or an indirect benefit (e.g., to be known as a discount coupon distributor). The site location is assumed to be fixed over time. The LSP – the coffee shop owner – need have no prior relationship with either Alice or Bob, each of who can remain permanently anonymous to the LSP.

C. System Operation

Figure 1 depicts our authentication system architecture. To an existing 802.11x access point with an internet connection, an LSP minimally adds 1) a femtocell (for each supported carrier), and 2) a computer operating as a *location server*. The location server hosts a web server, and offers a public page with detailed site location information (e.g., GPS, postal address, and contact information). The location server also continuously monitors the average bandwidth on the (encrypted) downlink between the AP and femtocell; an average bandwidth for each 1 second interval is measured, and these values form a *Return Data Feed* (RDF) that is publicly exported. Note that the computational burden of the location server is sufficiently small that in practice it can be run directly on either the AP or the femtocell. Internet middleboxes might exist between Alice and Bob, limiting her ability to use network geo-location techniques to locate him.

The figure also depicts Bob's mobile Service Provider's core network. Alice need not share a common operator network with Bob, nor even know Bob's operator. Regardless of source, any voice or data communication from Alice to Bob will ultimately traverse Bob's operator's network on route to Bob.

Alice must communicate with Bob during the authentication. We assume that Bob carries a mobile device, is in range of the LSP's femtocell and has associated with it. If Bob has a voice-only phone, Alice will establish a voice call to Bob; otherwise Alice will perform a data transfer. Alice controls a data source (e.g., a web server) that can be used to exchange data with Bob's smartphone.

Consider the following basic authentication process:

1) Bob successfully binds to the femtocell.
2) Bob communicates with Alice, and provides her with the LSP's location URL and his phone capabilities (e.g., basic phone).
3) The location server continuously monitors the (encrypted) AP-femto downstream link and exports an RDF stream reporting 1) the average bandwidth over each one second interval, and 2) the number of packets received in the previous second of each observed packet length.
4) Alice communicates with Bob. If Bob has a voice-only phone, Alice initiates a voice call and sometime later terminates the call. If Bob has a smartphone, Alice transfers data to Bob and controls her transfer's rate and packet sizes to impress a data traffic signature on the AP-femto link.
5) Alice monitors the exported RDF for characteristics of her voice call or data transfer.

Of course, these operations can be automated and need not be performed manually. Alternately Alice can assign a third party to perform the transaction. When Alice communicates with Bob, she expects the bandwidth measured on the femtocell ingress to increase and expects the bandwidth to fall when she terminates communication.

- If the behavior of the RDF convinces Alice that she is observing her own voice or data traffic traverse the AP-femtocell link, Alice confirms Bob's phone's association with the femtocell, and concludes that Bob is present at the specified location.
- If the observed RDF does not reflect Alice's transmissions, she can not conclude that Bob is on-site. Alice can elect to retry her transmission at a later time to confirm Bob's presence.

Alice's transmission to Bob impresses a distinct traffic envelope on the AP-femtocell downlink. Within a few seconds of initiating a voice or data transfer, Alice expects to observe the measured average bandwidth values increase by her transfer rate. She expects a similar decrease within a few seconds of terminating her call or transfer. Note, of course, that other subscribers of Bob's mobile operator might be present at the location, be associated with the femtocell, and also might be receiving voice and data traffic through the femtocell. But many of those present will likely select the available higher-bandwidth and less costly Wifi data service, and opt less for data service through the femtocell channel.

D. Prototype System

To explore the practicality of our proposed location authentication system we constructed a complete single-carrier system prototype. Our prototype uses the Verizon 3G Network Extender (Samsung 2CS-2U01) femtocell; the bandwidth measurements we report here are representative of voice codecs and transport protocols deployed by Verizon Wireless. An x86-based commodity PC with multiple ethernet NICs running a standard Linux 2.6.34 kernel serves as the location server. In contrast to Fig. 1, the server is located inline between the AP and femtocell, and traffic is forwarded between NICs via a standard network bridge. Bandwidth measurements are taken by reading a bridged interface directly with one of various, widely available tools such as *bwm-ng v.0.6* and *ifstat v.1.1*. The upstream link from the wireless AP is a shared DSL connection with rates of 3 Mbs downstream and 768 kbs upstream, which we would expect to be representative of the modest bandwidth available for many broadband public internet access channels.

An Apache web server offers users a static page with detailed site location information, including GPS coordinates, and a URL to access online bandwidth measurements. Real-time measurements are initiated on-demand, and exported via *netcat* on a separate interface to not impact bandwidth measurements. Verifiers are also able to request graphical views of bandwidth measurements for an epoch to permit a visual indication of ingress link traffic characteristics; compact *sparklines* are generated with Javascript for remote parties who are display-limited (e.g., smartphones). Our detection

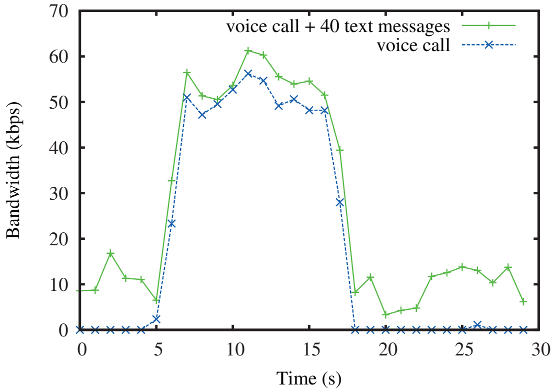


Fig. 2: Average bandwidth measurements captured from our prototype for a 30 second epoch containing an 11 second voice call (dashed), and the same call with 40 randomly time-offset text messages (solid).

algorithms – to be introduced in the next section – are compactly implemented in Python. We next describe how these algorithms detect the presence or absence of Alice's communication, even when competing with significant cross-traffic from other users of the AP-femto link. We study Alice's use of voice and data transmissions separately in the next two sections.

III. VOICE AUTHENTICATION SIGNAL DESIGN AND DETECTION

Consider the problems of 1) the design of the voice traffic signal Alice chooses to use to serve as her *fingerprint* that she is indeed using the link, and 2) extracting that signal from other traffic generated by femtocell users on site (e.g., voice calls, text messages, and web accesses), and 3) evaluating the probability that Alice herself is using the link, and consequently authenticating Bob's location.

Recall that we limit our attention to traffic signals Alice can send with no change to existing mobile handsets or infrastructure; using voice signals permits authentication of the location of voice-only UE, which continue to represent more than 50% of mobile phone users. In Section V we consider preferred approaches for data-capable mobile devices. Fig. 2 illustrates the captured bandwidth samples of a typical inbound 11 second voice call (dashed). To represent a heavily used link, the figure also shows the aggregate bandwidth of Alice's call occurring concurrently with cross-traffic we constructed by adding 40 randomly time-offset copies of a captured text message in a 30 second interval (solid). This 'noisy' signal is returned to Alice typically after a few seconds delay for her to evaluate the presence or absence of her call.

Suppose Alice uses a *single* voice call to authenticate Bob's location. Though she initiates the call, Alice has imprecise control over both call establishment timing and the shape of the bandwidth envelope associated with the call's packets arriving to the monitored link. Prior to initiating a call test, Alice defines an observation window (or epoch) of duration T, taken to be sufficiently long to complete her call test and observe its effect on the return channel (e.g., $T = 30$ sec.). Alice records an estimate of call start time $\hat{t}_{start}$ and stop time $\hat{t}_{stop}$, and calculates an estimated call duration $\hat{D} = \hat{t}_{stop} - \hat{t}_{start}$.

The signal observed by Alice on the return channel in each epoch is $r[i], \quad i = 0, 1, \dots, T-1$. Alice executes a detection algorithm on the received stream to choose between the hypotheses

$$r[i] = \begin{cases} s[i] + n[i] & H_1 : Alice's\ call\ present \\ n[i] & H_0 : Alice's\ call\ not\ present. \end{cases} \quad (1)$$

The received signal is modeled as the sum of two components: a signal $s[i]$ of duration D corresponding to the transmitted call, and a noise signal $n[i]$ which captures the bandwidth contribution of any cross-traffic on the link.

Our detection algorithm comprises 3 heuristics informed in part by the maximum-likelihood detection of signals in classical digital communication systems such as pulse-width and pulse-position demodulation. However, unlike a conventional communication system, the transmitted signal is not completely known to the sender, but can be constructed approximately; Alice does not directly control the voice signal's encoding or packetization, rather Bob's operator's network does. Our detection algorithms combine amplitude detection, edge detection, and the structure of the convolution of the received signal with our estimate of the transmitted signal; additional details can be found in [3].

The interfering cross traffic types we face are text messages, data transfers (primarily web downloads), voice calls, and control traffic. We will not consider control traffic here since 1) it consumes negligible bandwidth in the femtocell's 'operational' state, and 2) we have no control over its transmission. Text messages are typically low bandwidth (e.g., 1 or 2 kbs) transfers of only a few seconds duration. To study a large number of text messages arriving independently of each other in an epoch, we sampled the bandwidth of an actual arriving text message, then summed copies of the bandwidth samples with random time offsets across an epoch. Figure 2 shows that the aggregated messages form smooth, time-homogeneous traffic; these message have virtually no impact on our ability to detect the presence of a voice call. However, while never observed experimentally it is possible that several text messages could be queued in the mobile operator's network due to network congestion or temporary transmission failure, and suddenly be released in a burst. In such a case, even a modest number of arriving text messages (e.g., 10) could interfere with voice call detection.

It is the timing rather than the magnitude or number of interfering voice calls that cause a voice call detection failure. For example, one or more existing calls that outlast a test call look like time-homogeneous background traffic that do not inhibit call detection. But interfering calls that either start of stop near in time to the test call can disrupt its detection. Figure 3 shows the received signal when the voice call of Figure 2 'collides' with a second voice call that begins at time $t = 15$ secs. Though this interference appears disruptive, note that the trailing edge of Alice's call is intact, and our algorithm successfully detects the presence of the call.

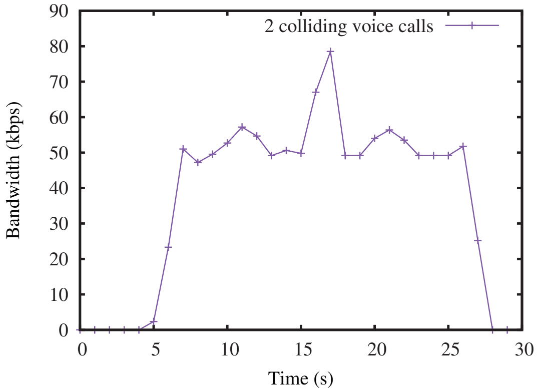


Fig. 3: Average bandwidth measurements when a second voice call slightly overlaps in time with the call we seek to detect.

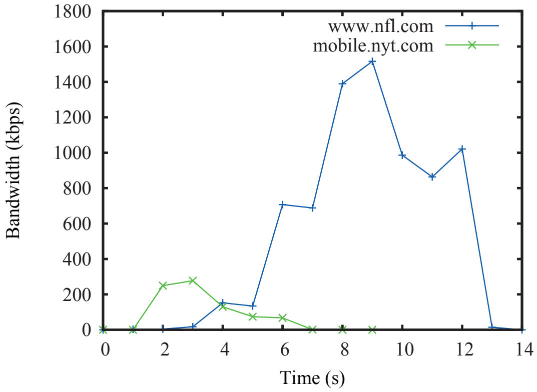


Fig. 4: Average bandwidth measurements for two web page downloads. Some downloads both consume high bandwidth and span a large fraction of the observation interval (nfl.com).

For the 3G femtocell we tested, data traffic (e.g., file transfers, web pages, and streaming media) was typically high bandwidth (e.g., $100 - 2000$ kbs) bursts of several to tens of seconds duration; Figure 4 shows typical examples. Our tests show that the transmission of even a single data transfer near in time to the start or stop of a test call is nearly certain to disrupt detection. In general, a voice call whose duration exceeds that of interfering data traffic promises to be most easily identified. Of course, Alice is always at liberty to issue a sequence of multiple test calls of varying duration if she is uncertain that she is observing her own traffic.

IV. An Analytical Model of Detection

We next develop an analytical model to determine the detection probability for a voice signal in the presence of interfering cross-traffic. As long as the channel bandwidth is not fully utilized (i.e., saturated), a mix of interfering voice, text, and data traffic bandwidth is additive. In general this noise forms a non-stationary process, though to begin we consider an idealized, stationary noise model.

Suppose that in each epoch we take the arrival times of individual interfering traffic bursts to be a Poisson point process with rate λ. The expected number of arrivals in an interval of duration t is then λt. Each interfering traffic burst has a variable bandwidth, and has a duration lasting several seconds, and hence interfering traffic bursts can overlap in time. These overlaps can cause the instantaneous bandwidth consumed by interfering traffic to vary greatly, in some cases far exceeding the level of the voice signal we seek to detect.

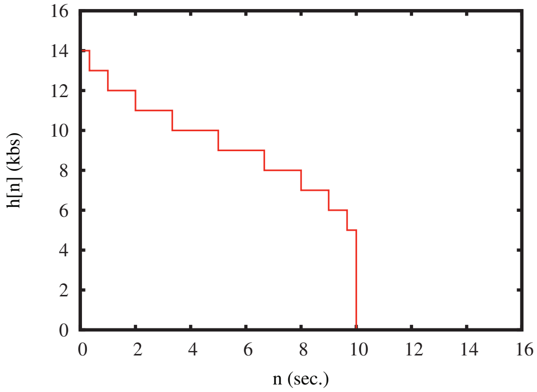


Fig. 5: A synthetic model of the bandwidth consumed by a single instance of aggregated, interfering text message cross traffic. The duration of the noise burst is 10 seconds.

Suppose we let $h[n], n = 0, 1, ..., T - 1$, be a sequence corresponding to the bandwidth consumed each second (kbs) by a noise burst of maximum duration T, with $h[n] = 0$ for $n < 0$ and $n \geq T$. Each burst might represent a single interfering message, such as a text message or data transfer, or an aggregate of several such messages arriving in an interval of duration T. The resulting noise model is similar to the well-studied, continuous-time ‘shot noise’ in electronic circuits, where poisson impulses arrive at random time instances t_i to a circuit with impulse response $h(t)$ to produce a noise process

$$\hat{s}(t) = \sum_i h(t - \hat{t}_i). \tag{2}$$

The probability density function of the shot noise $\hat{s}(t)$ ([6], p. 565) is

$$f(s) = e^{-\lambda T} \sum_{k=0} g_k(s)(\lambda T)^k / k!, \tag{3}$$

where $g_0(s) = \delta(s)$, $g_1(s) = g(s) * \delta(s) = g(s)$, ... , $g_k(s) = g_{k-1}(s) * g(s)$, ... , and the density function $g(s)$ satisfies

$$\int_0^s g(x)dx = Pr[h(t) \leq s]. \tag{4}$$

Informally, in our setting we seek the probability that sum of a sufficient number of interfering noise instances – each defined by a continuous-valued, discrete-time function $h[n]$ – arriving at an average rate λ will exceed some threshold value s (i.e., $\Pr[\hat{s}(t) > s]$) and hence interfere with our detection. Suppose we model the bandwidth of each instance of noise by the sequence of bandwidths $h[n]$ depicted in Figure 5. In this example, the average duration of the noise instance is 10 seconds, and the magnitude is initially 14 kbs trailing off to 5 kbs. The corresponding probability density function $g(s)$ is shown in Figure 6.

Figure 7 depicts the cumulative distribution function of the aggregated noise as we vary the arrival rate of this noise. Recall that the voice signal we seek to detect has bandwidth of roughly 50 kbs. Hence, an aggregate noise bandwidth nearing

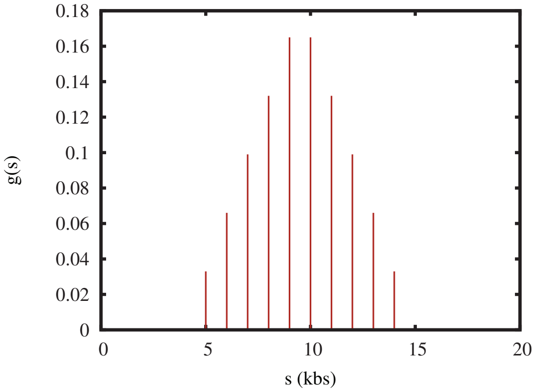


Fig. 6: The probability density function of a noise instance.

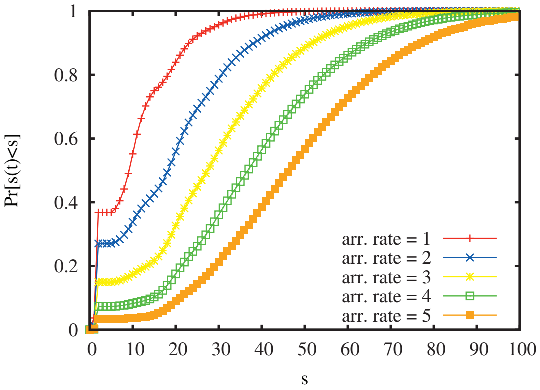


Fig. 7: Cumulative Distribution Function of aggregated noise as arrival rate λ varies.

that value near the start or end of a voice call will likely disrupt signal detection; in our implementation noise exceeding 40 kbs near the signal edge would be disruptive. The figure shows that the probability that our aggregate noise exceeds 40 kbs is $1 - .991 = .009$ when the noise arrives at rate $\lambda = 1$, but increases to $1.0 - 0.385 = 0.615$ when the arrival rate increases to 5. This latter result is consistent with our intuition; each arriving interfering signal has bandwidth of roughly 10 kbs, and the expected number of (overlapping) noise signals is 5, then we would expect the aggregated noise to exceed 40 kbs with probability of more than half.

V. DATA AUTHENTICATION SIGNAL DESIGN AND DETECTION

The high bandwidths achievable by non-messaging based data transfers suggest that they may be ideally suited for use as an easily identifiable authentication signal. Indeed, the efficacy of using a fixed-rate voice-based authentication signal will diminish as interfering data transmission rates increase (e.g., LTE systems operating at 15 Mbs and higher). In this environment Bob must have a data-capable device such as a smartphone or mobile computer, and Alice must be capable of controlling a data transfer. The data can be pushed or pulled, and the underlying transfer protocol is unrestricted. One simple approach that is consistent with our design objectives – namely mobile device independence and mobile user opt-in – is for Alice to provide Bob the URL of a data file on a web server she controls, and allow Bob to initiate the data transfer. Note that *http* transfers potentially avoid the need for Bob to have a special-purpose application to receive the transfer. In the next sections we discuss several approaches that might be used for data signaling. In the first scheme, Alice controls the rate at which data is transmitted.

A. *Rate Encoder Implementation*

To begin we explored the feasibility of Alice transferring rate-controlled data to authenticate Bob's location. Figure 1 shows how in our prototype system we introduced an *httpd* server on a second x86-based commodity PC running Linux to serve as Alice's controlled data source. Immediately prior to a location authentication transfer we create a randomly named file with dummy data using the *dd* utility; the file must be sufficiently large in size to continuously transmit for the duration of the epoch at a rate specified by Alice; we typically used $200 - 800$ KB file sizes.

Rate control was implemented using native Linux traffic control on the egress interface; we chose to use a *Hierarchical Token Bucket* (HTB). Note that the rate determined by Alice should be lower than the available bandwidth on the end-to-end transmission path between the server and Bob, otherwise the transmitted packets would be delayed in the network and the average bandwidth rates observed at the femtocell ingress would be less than the rates transmitted at the source. We typically operated conservatively by using a maximum transmission rate in the range of 50-300 kbs for authentication.

Figure 8 reveals that our prototype can perform rate control accurately. A file transfer from Alice's server to Bob is rate-limited at all times, ensuring the rate envelope is achieved due to a continuous backlog of data to transfer. Rather than send a fixed-rate transmission, the rate of our HTB was modified each second by a sinusoid with amplitude 100 kbs and period $T = 10$ seconds (i.e., fundamental frequency $f_0 = 0.1$ Hz), i.e.,

$$s(n) = 200 + 100\,cos(0.2\pi n), \quad n = 0, 1, 2, \ldots. \tag{5}$$

The figure shows the target rate imposed by our limiter, and the actual transfer rate at the egress of Alice's server. Both rate and timing are controlled sufficiently accurately to impress a discernible signature on a traffic envelope.

B. *Using Rate Controlled Transfers*

Network queuing and congestion will modify the envelope of a transmitted flow before its arrival to the femtocell, even if that envelope is slowly changing. The flow's path through the network is long; from web server through the public internet through Bob's operator's network and back out across the internet to the femtocell.

Given this imprecise control of the arrival stream, how should Alice rate-control an authentication data transfer to ease her detection of her signal's presence in the exported RDF? One simple approach is to alternate the transmission rate between two fixed values (e.g., 150 kbs and 250 kbs) chosen randomly by Alice on a per-transaction basis. Alice

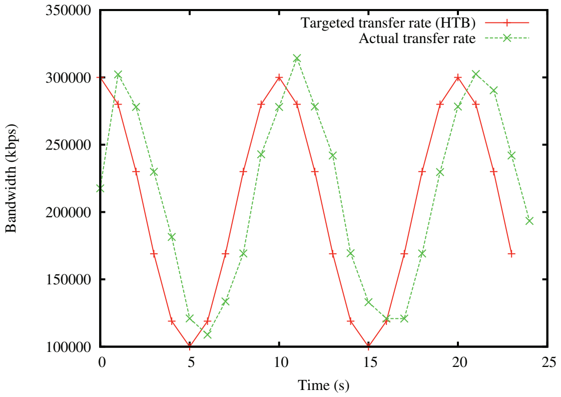


Fig. 8: The targeted transfer rate of backlogged data leaving Alice's rate-controlled server (solid) is a raised sinusoid with 200 Kbs average rate and a 10 second period. The actual measured transfer rate (dashed) closely follows the rate target.

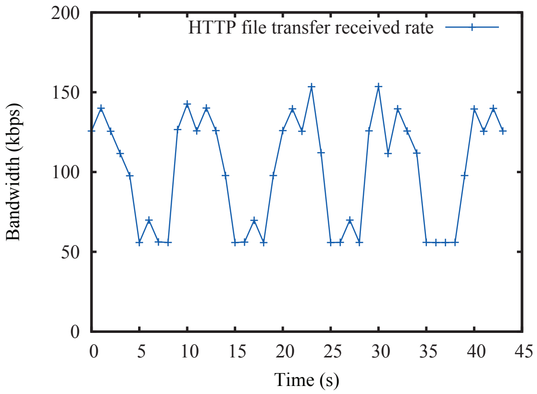


Fig. 9: The arrival rate to the femtocell for a source alternating between 100 kbs and 150 kbs with a 10 second period.

can then observe the channel for rate changes of approximately ±100 kbs occurring at the times she adjusts rates. Of course, cross-traffic sharing the femtocell downlink can interfere with detecting this signal.

Figure 9 shows the average arrival rate at the femtocell for a transmission oscillating between target rates of 50 kbs and 150 kbs every 5 seconds. Such a slowly time-varying envelope can be readily detected by Alice, though it requires an observation period of 10 seconds or longer. Other types of envelope shapes can shorten the necessary observation period. Modulating Alice's transmission envelope with a raised sinusoid of fixed but randomly-chosen amplitude and frequency promises several compelling advantages. First, rate-limiting at the sender is no more difficult than for a simpler signal. More important is that signal detection is simpler. Detecting such a signal should be robust; we intuitively expect relatively little energy observed at the sinusoid's fundamental frequency due to interfering cross traffic. Finally, the presence of this signal is less easily perceived by any observers of the channel. Figure 10 shows the average arrival rate at the femtocell for an authentication signal modulated with rate given by $s(n) = 100 + 50\,cos(0.2\pi n),\ n = 0, 1, 2, \ldots$.

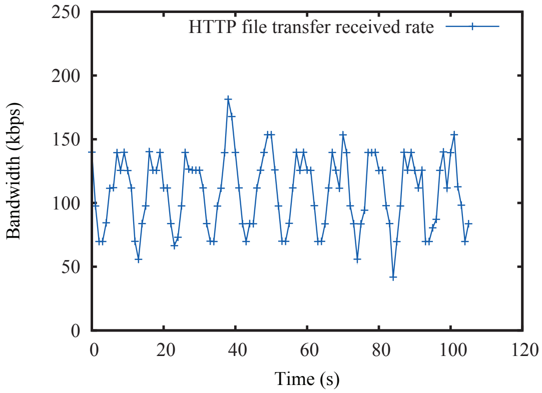


Fig. 10: The arrival rates for a rate-controlled source modulated by a raised sinusoid.

By modulating rates with a raised sinusoid, Alice is effectively sending a *hidden tone* as her authentication signal. Hence, our detector should resemble a frequency-selective bandpass filter tuned to the the selected tone. The implementation is simple; Alice receives the set of returned bandwidth samples for each epoch, i.e., $\{r[i],\ i = 0, 1, \ldots, T-1\}$, and calculates its Discrete Fourier Transform (DFT). Alice evaluates the amplitude of the DFT coefficient corresponding to the frequency of the hidden tone, and determines if that value is larger than the coefficient evaluated in other epochs when she is not transmitting.

The effectiveness of this detector is demonstrated in the following synthetic example. For 30 seconds we embedded the captured sinusoidal authentication signal in Figure 8 in 300 time-offset instances of a captured text message, with offsets either randomly chosen or correlated (in separate experiments). In each case the nominal bandwidth of the aggregated interfering messages is roughly 120 kbs, while that of the signal is only 100 kbs. As expected, the random interfering traffic has little energy at the signal's fundamental frequency. The magnitude of the amplitude of the corresponding DFT coefficient always exceeded that of the noise alone by a factor of 5 to 10, permitting easy detection of the presence of the authentication signal.

Nonetheless, additional experiments show that as the rates of interfering data transfers increase (e.g., LTE systems), energy across a wide range of frequencies can increase rapidly and unpredictably, making robust hidden tone detection more difficult.

C. Using Packet Sizing in Data Transfers

Now suppose Alice seeks to create an easily discernible traffic signature by modifying packet lengths associated with her data transmission. Her objective is to set each packet size to a randomly-selected, infrequently observed value; this size could be fixed, or could vary over the transfer lifetime. To determine such a value(s), we observe that the typical length distribution for packets arriving to femtocell ingress is bimodal. Voice traffic comprises almost entirely of small packets (e.g., 40-200 bytes), and data transfers are a mix of small (e.g., TCP acknowledgments for outbound data) and large (e.g., 1300 bytes) packets transporting data. Hence Alice chooses a value (or values) in the range of 400-1000 bytes, avoiding a few commonly occurring sizes (e.g., 512 bytes).

Suppose a file transfer normally includes N packets of size greater than 1200 bytes with a path MTU of 1500 bytes. If instead Alice chooses to reduce her packet sizes to a maximum of 550 bytes (e.g., by temporarily setting her server's NIC's MTU to 550), we expect the data transfer to contain approximately $2N$ packets of length approximately 550 bytes. Recall that transfers to the femtocell are encapsulated by the mobile operator, representing a packet length increase of roughly 10% at the ingress link.

Consider the following example. Figure 11a depicts the measured average bandwidths of a high rate web transfer that might represent cross-traffic while Alice is transmitting her authentication signal. Figure 12a shows the numbers of packets at each length for that interfering transfer. As expected, we see a bi-modal distribution of entirely either small or large packets. Figure 11b depicts a rate-controlled transfer from Alice where she does not control packet length. Packets lengths associated with this transfer appear in Figure 12b; here we see approximately equal numbers of packets of two, tightly clustered lengths: large (i.e., 1390 B) and medium-sized (i.e., 390 B). Figure 11c illustrates the same data transmission with Alice electing to both rate-control and set packet size to 512 B for the duration of the transfer. As expected, Figure 12c shows (solid) that we no longer see large packets during the transfer, but instead see more than double their number arriving with length of 590 B, the size of the largest possible transmitted packet with encapsulation overhead. The packet counts shown as dashed correspond to what Alice would also observe if the web transfer of Figure 11a occurred in the same interval as her authentication transfer. Clearly, a detector looking for the expected largest packet size of Alice's transmission – in this case the unusual size 590 B suddenly arriving at a rate of 20 packets/sec – would rapidly determine that Alice is using the channel, and confirm Bob's location.

Our observations of femtocell ingress voice and data traffic indicate that each packet length on 16 B boundaries in the range of 600-1300 B occurs for less than 0.1% of arriving packets; most lengths are not observed at all. If desirable, of course, Alice could further improve the reliability of detection by sending a sequence of very short transfers each of which has a distinct, unusual packet length from that range. Such an approach would also strengthen the system from attacks, a topic we discuss in Section VIII.

VI. AUTHENTICATION WITH MESSAGING

The possibility that a voice call will be undetectable when swamped by high-rate interfering data traffic suggests that data traffic – rather than voice – should preferentially be used as the authentication signal. To continue to permit authenticating the location of basic phone users, we next turn to signals based on the Short Messaging Service/Multimedia Messaging Service (SMS/MMS).

Simple text messages represent too little data to be readily detected in the presence of either interfering cross-traffic or control traffic to and from the femtocell itself. But MMS transmissions – messages with large media object attachments – can be used for data transmissions of up to roughly 1 MB before encountering timeouts resulting in uploading failure.

Unfortunately, the implementation of message delivery makes these signals unsuitable as authentication signals. Messaging operates in a store-and-forward mode, with a message upload and download separated in time by an unpredictable, and often long (e.g., 8-10 second) delay. Even more difficult, the message upload and download transmissions proceed at rates determined by those channels; the upload typically advances more slowly. Hence, even if Alice is equipped to observe the bandwidths of both the upload and download of her transmission to Bob, her ability to associate them is limited if cross-traffic is present on the femtocell ingress link. Figure 13 shows how the average bandwidths of a 440.3 KB video message appears when uploaded by Alice, and subsequently downloaded to Bob. In general, Alice is unable to discern reliably that traffic arriving to the femtocell is hers rather than messaging or data transfer destined to other users of the femtocell.

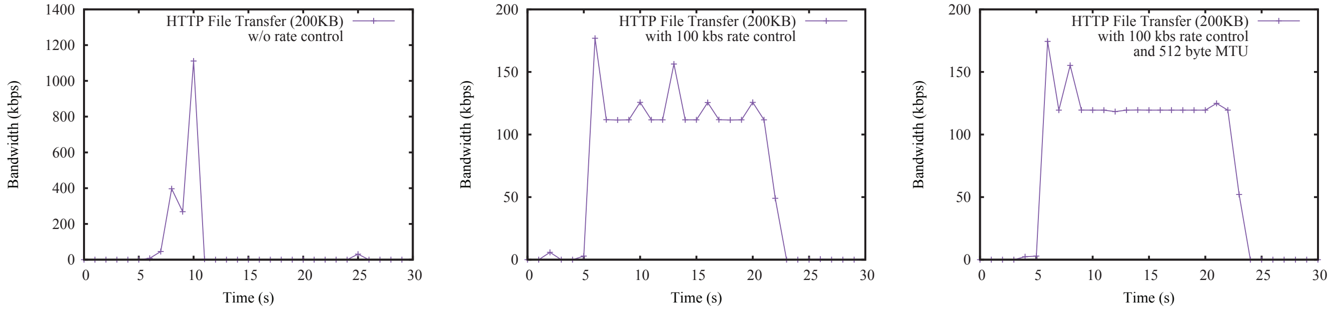
VII. DISCUSSION OF AUTHENTICATION SIGNAL PROPERTIES

We have identified several types of authentication signals, each of which has distinct properties. If Bob has a voice-only phone, a voice call is the only authentication signal available to Alice. The principle advantage of using voice call based authentication is that it can operate from any voice source – even a landline – and can authenticate any voice-capable mobile device. But there are multiple disadvantages of this technique. Voice call initiation requires Bob to go off-hook, which typically occurs several seconds after Alice initiates dialing, making even a quick authentication call relatively time-consuming (e.g., 5-10 seconds).

Further, since the envelope of every voice call of similar duration is roughly identical, any roughly contemporaneous voice call potentially interferes with Alice's ability to detect her authentication call. Additionally, as network transmission speeds increase, the bandwidth of data cross-traffic overwhelms that of an authentication call, making its detection less reliable.

To authenticate data-capable mobile devices, Alice can either manipulate the rate or the size of her transmitted packets. Rate-controlled data signals can use rates up to the channel bandwidth, so an authentication signal can be made sufficiently high bandwidth that most cross-traffic (e.g., text messages) will likely not interfere with signal detection. Further, Alice can change her authentication signal with each use, and the unpredictable nature of such a transmission can make detection easier, and any adversary's job more difficult. A challenge with this technique, however, is that detecting the presence of an authentication signal can be relatively slow. Given that Alice sends a relatively few number of packets each second, an envelope waveform such as a raised sinusoid requires several seconds of transmission to detect reliably.

Authentication signals based on packet length manipulation are easy to generate, and easy to reliably detect in the presence

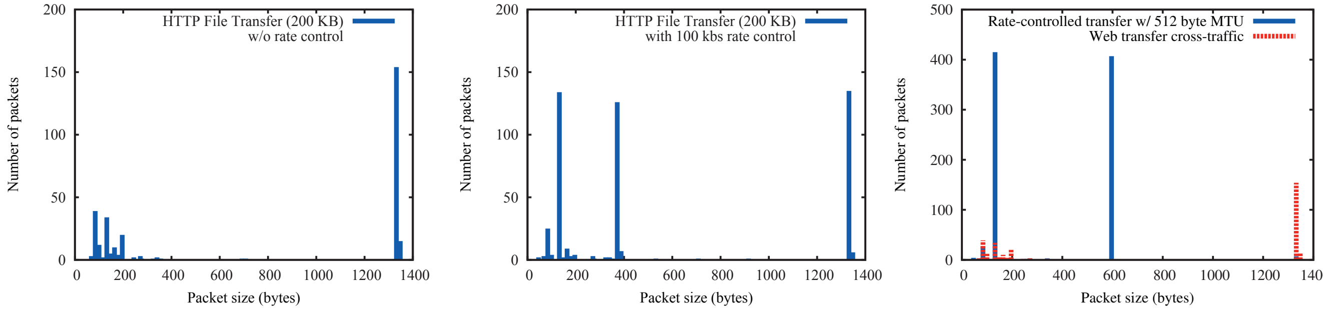


(a) Measured bandwidth with no server rate control.

(b) Measured bandwidth with 100 kbs server rate control and 1500 byte MTU.

(c) Measured bandwidth with 100 kbs server rate control and 512 byte MTU.

Fig. 11: Received bandwidth with/without server rate control.



(a) Received packet lengths with no server rate control.

(b) Received packet lengths with 100 kbs server rate control and 1500 byte MTU.

(c) Received packet lengths with 100 kbs server rate control and 512 byte MTU.

Fig. 12: Received packet lengths with/without packet length control.

of background traffic. As with rate-controlled signals, the encoding of an authentication signal into packet lengths is performed at transmission time, and the signal can be unique to each authentication transmission. Alice's packet length detector compares the lengths of packets she transmitted as her authentication signal versus the packet lengths reported on the RDF, taking the round-trip delay into account.

Suppose that to perform an authentication Alice transmits exactly k_i packets of an infrequently observed length i. Suppose also that N cross-traffic packets arrive in her observation period, and packets of length i occur independently of other lengths with probability p_i. Alice would incorrectly conclude that her packets arrived to Bob's purported femtocell if she observed an RDF with exactly the same number of packets arriving with the length she selected. The probability that Alice observes exactly that number of packets – and consequently makes a false confirmation of Bob's presence – is

$$\binom{N}{k_i} p_i^{k_i} (1 - p_i)^{N-k_i}. \tag{6}$$

Our observation of current femtocell ingress traffic suggests that such an error probability would ordinarily be exceedingly low (e.g., less than 10^{-3}). Note, of course, that if packet length based authentication was a standard technique, Alice would be competing with other authenticators transmitting on a limited number of rarely used packet lengths (i.e., roughly 90 available lengths), increasing her authentication error rate.

Fortunately, many system enhancements are available to increase the sophistication of the detection system and further lower the error probability, if desired, particularly in an environment with adversaries. Consider the following example – the RDF could return the actual *sequence* of observed packet lengths in each epoch, and Alice could look for a particular length sequence corresponding to her transmission. Of course, such a system would require the RDF bandwidth to increase slightly, and slightly increase the complexity of Alice's detector.

VIII. Security Analysis

We next examine the security and privacy properties of the proposed Location Authentication (LocAuth) system, and discuss its resistance to some frequently suggested attacks. While many attacks are easily conceived, they can be deceptively complicated or costly to successfully implement. We do not strive to exhaustively consider all possible variants, rather we highlight those that we consider likely to be most effective or difficult to prevent. Finally, we describe several simple enhancements to improve overall system robustness. We also introduce countermeasures for specific attacks, some of which are simple to implement yet can render attacks ineffective, more easily detected, or more expensive to mount.

The location system's attack surface is defined by the three principals (or actors) – Bob, Alice, and the LSP – and five critical system components, namely the AP, femtocell, location server, web server, and UE (i.e., primarily smartphones). Of the principals, either Bob or the LSP might interfere with a verification. Recall that we consider Bob to be cooperative but untrustworthy. More precisely, Bob may simply seek to appear to be cooperative, even if misrepresenting his location and working to interfere with a verification. In general, the LSP stands to gain by hosting a verification venue (e.g., perhaps by receiving direct or indirect compensation from Alice), and any of his current and future gain is put at risk if he is caught interfering with authentications. External parties might also attack the system; we refer to these as malicious non-principals (MNPs).

A. *Disrupting Service*

An attacker can prevent a LocAuth from succeeding either by compromising one or more system components, or by disrupting operation of the network(s) interconnecting those components. Like any internet-attached service, a LocAuth system is subject to network-based Denial-of-Service (DoS) attacks. Under an attack, Alice might be unable to verify Bob's location, or Bob might be unable to establish his location. DoS can be executed by MNPs or the system principals themselves. As an example, an LSP can simply interrupt its service at any time (e.g., denying any parties within range from being authenticated by remote parties).

A DoS attack on a LocAuth site can focus on either the femtocell or the location server. A location server under attack might be unable to respond to a web request for the location URL, or continuously transmit the exported RDF. An attack on the femtocell or AP can saturate the downstream bandwidth to the device such that either incoming or outgoing calls can not be forwarded by the femtocell. Interestingly, attacks on the femtocell or AP can be initiated locally by an onsite MNP, either by consuming all available bandwidth (preventing Bob from communicating) or by broadcasting a radio jamming signal. A network DoS attack launched against Alice's network can also impede her ability to authenticate any party. Of course, Alice can mitigate such attacks by initiating authentications from multiple locations, or having proxies perform authentications on her behalf.

Observe that the authentication system comprises a network of decentralized, independent, geographically distributed verification sites with no centralized component. This offers considerable protection against DoS attacks; while individual LocAuth venues can be attacked, the effort (e.g., bandwidth) required in a multi-location attack grows with the number of attacked authentication locations.

An attacker can alternately subvert a verification by *compromising* any system component (e.g., establishing supervisory control of a component). AP and web server compromises are both achievable and well-studied, but are outside the scope of this paper. Lack of physical security has also been raised as a vulnerability potentially facilitating femtocell compromises [7]. Finally, smartphone users' willingness to download non-certified applications with little reservation remains a compromise threat whose extent has yet to be fully understood [8].

B. *Deceiving the Verifier*

We next consider how Bob can act to deceive Alice by attempting to convince her that he is at the claimed authentication site when he is elsewhere. Deception attacks invariably take one (or both) of the following forms; either Bob attempts to deceive Alice that she is communicating with the claimed location rather than his actual present location, or Bob deceives Alice by manipulating the RDF exported from the claimed location to indicate his presence.

Deception attacks can be mounted individually by Bob, or with the assistance of a colluder. Let's first consider the former. Suppose Alice transfers network traffic (either voice or data) through the femtocell by initiating a communication to Bob. For Bob to deceive Alice, she must observe behavior on the RDF that closely resembles what she expects – an increase in traffic of approximately 50 kbs shortly after a voice call initiates, and a similar decrease when she terminates the call, or a sequence of packet sizes consistent with her data transmission. To accomplish this, Bob must either 1) ensure that a call (or data transfer) with timing, bandwidth usage, and packet sizes consistent with Alice's expectation arrives to the femtocell, or 2) modify or substitute the RDF with a counterfeit feed consistent with her expectations.

To achieve the former, Bob can remotely 'forward' a (logical) copy of Alice's transmitted packet stream to the femtocell ingress at the claimed location. Note that any traffic forwarded to the femtocell does not necessarily need a recipient (or receiving application); even if dropped by the femtocell the bandwidth appearing on the femtocell ingress is sufficient for deception. Such a forwarding action must be performed quickly or Alice might detect the delay in the appearance of her traffic to the femtocell. Forwarding traffic to the femtocell through the femtocell's associated mobile operator would take several seconds, and likely be detected. Hence, Bob's preferred approach would be to direct a data stream mimicking Alice's transfer directly to the femtocell's IP address.

To achieve the latter, Bob can alternately send a *modified* RDF to Alice. For example, he can insert himself 'in-the-middle' between the (claimed) location server and Alice, and forward a modified version of the location server's RDF, enhanced to falsely include the channel characteristics associated with Alice's transfer.

As an alternative, Bob can send a substitute stream to Alice by providing her a false location URL pointing to a web site he operates from which he can also export an RDF he controls. A particularly elaborate version of this attack is as follows. Suppose Bob operates a Private Location Authentication system (PLA) at his current location, effectively impersonating the claimed location's LocAuth system. Bob provides Alice with a location URL that mimics the claimed location's, with his own RDF. In the absence of a central database of valid authentication sites, Alice places her trust in a network of unverifiable LocAuth system operators. Without taking additional steps to verify the legitimacy of the PLA site, it is possible for Alice to be deceived.

But note how difficult it would be for Bob to sustain this deception over time if he attempts to use the same PLA system

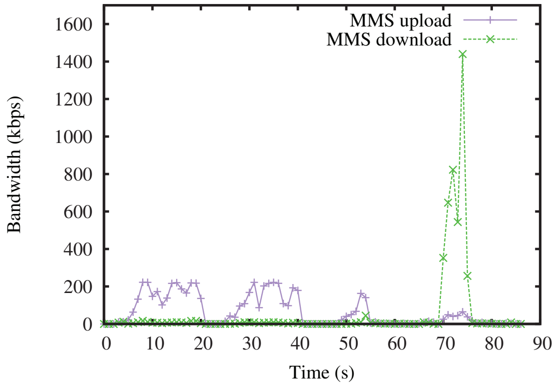


Fig. 13: Transmitted MMS message being slowly uploaded (solid) in the time interval from 4-55 seconds, and later downloaded rapidly (dashed) in the interval from 70-77 seconds.

to support multiple deceptions. LocAuth sites are more-or-less permanent and fixed; that is, a location page and an RDF are expected to be unchanged over long time periods. Hence, Alice expects to be able to reach these resources at *any* time in the future. As a result, Bob is obligated to keep these services running indefinitely. If Alice revisits the location URL, and finds it unavailable or changed, she can invalidate any previous confirmation of Bob's location.

Further, Alice can maintain a list of all URLs and RDFs previously provided by Bob. Suppose Bob attempts to use a single PLA to deceive Alice about his location in a sequence of deceits over time. When performing an authentication, Alice expects that the only RDF indicating her call is that of Bob's present location; Alice can monitor all past feeds to check for any activity her current call generates at Bob's purported previous (other) locations.

An example demonstrates the effort required by Bob to operate a single PLA. In deceit A, he claims to be at Location A, and creates a location web page that mimics that of the claimed location. For a later deceit B, he creates a second location web page mimicking location B. For Alice not to detect the deception, Bob must ensure that the IP addresses of the (supposedly different) web servers differs. More significantly, both location URLs would export the same RDF from the single femtocell in his PLA. If Bob claims to be a location B, Alice could monitor the previously provided feed for location A and detect Bob's deception. Hence, Bob must not only execute his current deception, but ensure that all previous deceptions remain active and do not raise suspicion.

C. Collusive Deceptions

Let's next turn to collusive attacks where Bob has assistance from a confederate (i.e., the colluder). Such an assistant is usually equipped with Bob's phone and positioned at Bob's claimed location. Of course, if Bob is unknown to Alice and passes his smartphone to an on-site colluder, we will be unable to distinguish him from another; Bob's private key on his smartphone is his identity.

Recall that if Bob possesses a voice-only phone, the best Alice can do is locate Bob's phone, and establish that Bob is in possession of his phone by speaking with him (if he is known to her). Hence a commonly proposed attack is for an on-site colluder to 'forward' Alice's voice conversation to Bob.

A common misconception is that this attack can be executed with mobile operator-based 'call forwarding' service. In many cases, however, this service is network-based redirection, and the incoming call would not reach the femtocell targeted by Alice, and she would not observe expected activity on the femtocell ingress. It is possible, however, for the colluder to implement UE-based forwarding. Forwarding via a mobile network is likely to result in a call-initiation delay detectable by Alice. Yet a call-setup delay can be eliminated if the colluder keeps a pre-established connection to Bob in anticipation of Alice's call. A preferred attack is for the colluder to convert the received voice signal to VoIP, and send to Bob over an internet connection. Such an attack would require modest technical sophistication to prevent Alice from hearing audible indications of forwarding (e.g., echoes and dial tone).

Collusive attacks require the existence of a relatively low-latency communication channel between Bob and the colluder to support a coordinated, timely deception. The low delay requirement generally rules out 3G/4G communication channels, where end-to-end delays can be significant. Bob can communicate with the colluder using IP over the claimed location's 802.11x AP, or a separate IP communication channel 'carried in' by the colluder that does not rely on LSP infrastructure.

A challenging collusive attack to detect has the LSP operating as Bob's colluder. In this attack, Bob signals the LSP to modify the bandwidth of the exported RDF stream by simply indicating the call initiation and termination times. The incentive for an LSP to collude with Bob would necessarily have to outweigh the risk that the deceit is detected, and the LSP's service is flagged as untrustworthy; a loss of all future revenue for the LSP could be the result.

Next we consider a collection of minor system modifications and *countermeasures* that make deceiving the verifier more difficult. A first tool to detect deception lies in the amount of information that is returned to Alice. In general, more information can assist Alice in both verifying location and identifying suspicious behavior, at the expense of consuming additional traffic on the RDF. In some cases, even a small amount of additional information can be valuable. For example, if the femtocell is receiving and dropping an excessive amount of incoming traffic, the femtocell could indicate a 'health' status indicating that the system might be under attack.

As a second example, the system can supplement the RDF with measurements of the femtocell *egress* link characteristics (e.g., average bandwidth measurements). Alice can observe the egress data to detect attempts to manipulate traffic on the femtocell ingress. As an example, if Alice is speaking with Bob she would expect the femtocell egress to behave in a fashion consistent with the ingress, with respect to her communication. If Bob is simply redirecting a data stream to the femtocell from a remote location, the RDF would not exhibit the expected egress link behavior. Rather, depending on system implementation Alice might see a small increase in traffic associated with ICMP redirects in response to data sent to an inactive TCP or UDP port.

Another means of protecting the system from external manipulation is to have the bandwidth monitor on the femtocell ingress report only the amount of bandwidth traffic whose source IP address corresponds to the femtocell's mobile operator security gateway (or domain). Any attack traffic originating from other IP sources would not be observed by Alice, forcing Bob to spoof IP source addresses.

Where possible, Alice should control the timing of her communication with Bob; she should 'push' data transfers to Bob, rather than let Bob 'pull' data from her web server. Otherwise, Bob can quickly send a URL provided by Alice to a colluder at the claimed location, who can then pull the data himself. This attack is particularly threatening since with no additional challenges a colluder would not need to carry Bob's phone to mount the attack.

Of course Alice can also perform a variety of actions to confirm a verification if she suspects deceit. Alice can execute multiple authentication transfers to Bob. Another approach is for Alice to add a challenge such as a request that Bob call her. If an RDF also reports bandwidth on the femtocell egress, than Bob would need to have a colluder support a forwarding of Bob's call to Alice (opposite in direction from the earlier attack where Alice calls Bob and the colluder forwards).

D. Privacy

We next examine the information exchange – and hence the potential information loss – between the system participants and/or external parties. To begin note that Bob realizes his principal privacy requirement, the ability to *opt-in* to an authentication on a per-transaction basis. His opt-in action takes the form of informing Alice of his location and the site URL.

In most applications Bob reveals his identity to Alice. If Bob has a voice-only phone, his speaking voice can serve as a personal identifier if he is known to Alice. In the case where Bob has a smartphone and the authentication is entirely automated, Alice can confirm Bob's identity by asking for part of his exchange to include a digital signature. Alice need not be known to Bob, though she too can sign an exchange to reassure Bob that he is revealing his location information to the intended party. This action can help Bob detect a malicious party seeking to track his location.

The LocAuth system has the unusual property that the location service provider – namely the site operator – need not have any knowledge of Alice nor Bob, nor the fact that they engage in a transaction. Nor does Bob's mobile operator, despite use of its infrastructure. Both Alice and Bob are protected from revealing their identity (or relationship) to the LSP. Further, since transactions are encrypted and usually not known to the LSP, no records are maintained that might later be revealed in a compromise of the system. In particular, an LSP eavesdropping on the femtocell ingress does not see Alice's IP address; all traffic appears to be to/from the operator's security gateway. Note, however, that Alice should anonymize her network address to minimize her risk that her identity can be determined by the LSP when she accesses the site URL. The LSP is also in a position to monitor unencrypted traffic through the public AP (i.e., that traffic not associated with the femtocell), making femtocell traffic analysis a relatively unattractive target to an eavesdropping LSP.

Consider the amount of information revealed to an external party eavesdropping on an RDF. Though the data stream appears to contain little valuable information, it forms a *covert* channel that can provide a remote party with an indication of the site occupancy. Such information is of potential value to a burglar waiting for an empty store to rob. In another example, a business analyst could examine the overall network utilization of all RDFs of every location of a certain business (e.g., a coffee shop chain) as an indication of store visit trends, and perhaps infer business activity. Of course a location can be densely occupied, but if none of the occupants are using the femtocell than this information is not revealed to an observer of the feed. Similarly, even a single occupant using the femtocell can download enough data to nearly fully utilize the femtocell downlink. Note, however, that traffic analysis by the eavesdropper might be able to distinguish between a single user, and multiple users, of a femtocell.

Finally, we note that Alice's data transfers to Bob exiting her web server may not be encrypted until reaching the SG and are subject to eavesdropping. But any traffic sent from Alice to Bob appears to be destined to a web proxy in Bob's operator's network domain. Hence an observer eavesdropping on Alice's server will not be aware when and if she is communicating with Bob.

IX. Related Work

Despite nearly 2 decades of research [9]–[16], authenticating mobile client location remains difficult. Classical authentication system proposals often relied on distance bounding [17], [18]. Location proof architectures almost invariably rely on deploying trusted infrastructure, often distributing trust across multiple system elements in a complex authentication overlay. Such systems typically strive to achieve a high degree of confidence in verification, frequently using cryptographic protocols to bind devices and identities. In contrast, our system places no trust in infrastructure beyond their normal operation, and aims for a simple architecture that avoids the complexities of trusted infrastructure management, but provides authentication strength consistent with the commercial needs of existing LAPs .

Our approach is similar to related work in two aspects. First, in principle, we assume that we trust an entity's location and then prove that a mobile device is near the entity; the entity could be a femtocell or an 802.11x AP [9], [19], [20]. Second, in implementation, we extend existing infrastructure by adding femtocells and location servers. In comparison, prior work requires certification authorities [10], APs capable of issuing cryptographic location proofs [9], [19], and trusted platform modules (TPM) [21]–[24]. Hence, the proposed approaches' success depends on the widespread deployment of either femtocells, cryptographically enhanced APs, and/or TPMs in smartphones. Our approach, however, differs in one key aspect: we don't use any cryptographic primitives and rely on lightweight traffic signals for authentication; hence we avoid managing complex infrastructure such as public key infrastructure and TPMs. Zeng et al. also use non-cryptographic techniques for authentication in a different context; they use physical layer characteristics for user authentication and device identification in wireless networks [25].

Lenders et al. use localization/certification authorities to securely tag location information to content generated on mobile devices [10]. Their approach, however, depends on an external mechanism to identify device location. Authentication systems that assume trusted user devices have also been proposed. Dua et al. [21] and Saroiu & Wolman [22] use TPMs to protect the integrity of raw sensor data. Similarly, Gilbert et al. use TPMs to guarantee the integrity of data derived from raw sensor data [23], [24]. TPMs, however, are not universally found in mobile devices, e.g., to the best of our knowledge, no commodity smartphone has a TPM chip. Moreover, even if devices had TPMs, the location sensing device inputs remain vulnerable to manipulation, e.g., using GPS signal simulators [26].

Due to the vast deployment of 802.11x wireless APs, the research community has focused almost entirely on location proof systems based on APs. Several proposals extend an AP's basic functionality to support location authentication; Luo & Hengartner [19] and Saroiu & Wolman [9] propose solutions that involve APs capable of issuing location proofs. Faria and Cheriton [27] introduce an authentication architecture where a centralized wireless appliance controls a group of APs, and broadcasts a set of random nonces through its controlled APs.

Some research on location authentication cleverly exploits channel observations in broadcast wireless networks (e.g., broadcast packets [11], [28] and modulated power [29]) to form shared secrets to establish user proximity to an AP. An alternate approach to reduce trusted infrastructure and resist collusion relies on the presence of on-site corroborators to verify user presence; some systems strengthen trust in unknown third parties by turning to reputation systems [12]. In contrast, our approach doesn't rely on any other system user's presence or actions.

Community interest has recently shifted to authentication systems using other communications technologies. Bertino and Kirkpatrick explore Near-Field Communications and dedicated location devices to create an access control scheme [30]. Relatively little research has focused on the role femtocell technology can play in providing location services. Borgaonkar et al. describe how the lack of physical security makes femtocell location reporting an appealing target for hackers [7]. Indeed, it is precisely this lack of physical security – femtocells are located on customer premises – that permits us to construct an authentication service.

Despite the proposed location proof systems' broad diversity, most systems – including ours – remain vulnerable to certain attacks. Collusive 'wormhole' attacks – where a remote party colludes with an on-site associate to fake one's presence – are the most challenging shared threats. Though distance bounding techniques may be a practical solution to these threats [31], it too suffers from weaknesses [32].

Despite these vulnerabilities, location based systems have enjoyed tremendous success in practice. WiFi Positioning Systems (WPS) – such as offered by Skyhook Wireless [33] – and hybrid WPS/GPS systems are the most popular location determination systems in use today for indoor/outdoor applications. More recently, *location-as-a-service* or *Where 2.0* companies (e.g., LOC-AID [34], Veriplace [35]) have begun to serve as intermediaries between mobile operators and third parties seeking client location. These aggregators not only locate clients with any mobile phone device, but serve the crucial role of locating clients served by different operators. While promising, bootstrapping these services is challenging; each client and third party must proactively establish a relationship with each aggregator.

X. CONCLUSION

We have proposed and demonstrated a novel approach to infrastructure-based location authentication that operates in a spontaneous, transaction-oriented fashion. Our approach strives to be well aligned with the evolving needs of internet location-based application providers, and particularly their desire to authenticate new users on-the-spot. We introduced techniques to use voice calls to authenticate voice-only phone users, and data transfers to authenticate smartphone users, and explored a diverse set of traffic signals that can authenticate users rapidly and reliably. Yet no single query can authenticate a mobile device user's location with certainty, particularly in the presence of adversaries. While we have studied the performance of each of the proposed traffic signatures in isolation, we anticipate that multiple techniques will be combined – and repeated over the duration of a call – to permit the authenticator to achieve her desired confidence in the authentication at a cost of additional time, bandwidth and complexity.

Many possible embellishments of our basic system proposal are fairly straightforward, e.g., a multi-femtocell configuration to support more users in a small physical space. Multi-carrier operation can be achieved by simply arraying femtocells from each service provider, and monitoring each downlink separately. Femtocells are, of course, not widely deployed today, as would be required to scale our system. But, apart from enabling new services, the basic advantages of wider deployment of small cell technology – both to operators and consumers – remain plentiful. Our system requires no changes to operator infrastructure or mobile user equipment. Hence, the technology required to deploy a large-scale location authentication system exists, is inexpensive, operates off-the-shelf, and can be deployed incrementally. While future large-scale deployment of femtocells is uncertain, we do envision the integration of femtocell and 802.11x radios in a single multi-access unit as being a potential catalyst for wider-scale deployment.

That said, many practical limitations must be addressed for the system to scale. The number of simultaneous voice authentications per femtocell is limited by the number of simultaneous voice users, which is typically 4-32 today. While multiple femtocells help overcome this constraint, RF spectrum limitations and interference concerns limit the number of femtocells in close proximity. While the number of simultaneous data authentications is far higher, that too is limited by factors including the backhaul bandwidth capacity per site, and the number of UEs that can camp on a single femtocell. The anticipated evolution to a wider range of 'small cells' with greater capacity will permit increased scale.

Our system exploits mobile-operator technology without actually involving the operator directly in a transaction. Yet we believe that more robust authentications can be achieved with the mobile operator's active involvement. In particular, operators control the infrastructure, have preferential network vantage points, and can create easily discernible authentication fingerprints.

References

[1] R. Netravali, J. Brassil, "Femtocell-assisted Location Authentication (poster/extended abstract)," *IEEE LANMAN 2011*, Oct. 2011.
[2] J. Brassil, P.K. Manadhata, "Verifying the Location of a Mobile Device User," *Proc. of MobiSec 2012*, June 2012.
[3] J. Brassil, R. Netravali, S. Haber, P.K. Manadhata, P. Rao, "Authenticating A Mobile Device's Location Using Voice Signatures," *Proc. of IEEE WiMob 2012*, Oct. 2012.
[4] V. Chandrasekhar, J. Andrews, A. Gatherer, "Femtocell Networks: A Survey", *IEEE Communications Magazine*, Vol. 46, No. 9, September 2008, pps. 59-67.
[5] S. Kent, "IP Encapsulating Security Payload (ESP)," *IETF RFC 4303*, December 2005.
[6] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, McGraw-Hill, 1965.
[7] R. Borgaonkar, K. Redon, J.-P. Seifert, "Experimental Analysis of the Femtocell Location Verification Techniques," *Proc. of 15th NordSec*, 2010.
[8] W. Enck, M. Ongtang, P. McDaniel, "On Lightweight Mobile Phone Application Certification," *Proc. of 16th ACM CCS*, 2009.
[9] S. Saroiu, A. Wolman, "Enabling New Mobile Applications with Location Proofs," *Proc. of HotMobile 2009*, pp. 1-6.
[10] V. Lenders, E. Koukoumidis, P. Zhang, M. Martonosi, "Location-based Trust for Mobile User-Generated Contents: Applications, Challenges and Implementations," *Proc. of Hotmobile 2008*, 2008.
[11] A. Narayanan, N. Thiagarajan, M. Lakhani, M. Hamburg, D. Boneh, "Location Privacy via Private Proximity Testing," *Proc. of NDSS 2011*, 2011.
[12] M. Talasila, R. Curtmola, C. Borcea, "Location Verification through Immediate Neighbors Knowledge," *Proc. of Mobiquitous'10*, 2010.
[13] R. Want, A. Hopper, V. Falco, J. Gibbons, "The Active Badge Location System," *ACM Trans. on Information Systems*, Vol. 10, No. 1, 1992, pp. 91-102.
[14] N. Priyanatha, A. Chakraborty, H. Balakrishnan, "The Cricket Location Support System," *Proc. of MobiCom'00*, Aug. 2000, pp. 32-43.
[15] D. E. Denning, P. F. MacDoran, "Location-Based Authentication: Grounding Cyberspace for Better Security," *Computer Fraud & Security*, Feb. 1996.
[16] T. Kindberg, K. Zhang, N. Shankar, "Context Authentication Using Constrained Channels," *Proc. of Fourth IEEE WMCSA*, 2002, pp. 14-21.
[17] S. Brands, D. Chaum, "Distance-Bounding Protocols," *Advances in Cryptology - EuroCrypt*, Lecture Notes in Computer Science, 1994, vol. 765/1994, pp. 344-359.
[18] N. Sastry, U. Shankar, D. Wagner, "Secure Verification of Location Claims," *Proc. of WiSe '03*, 2003.
[19] W. Luo, U. Hengartner, "VeriPlace: A Privacy-Aware Location Proof Architecture," *Proc. of 18th ACM SIGSPATIAL GIS 2010*, 2010, pp. 23-32.
[20] W. Luo, U. Hengartner, "Proving your Location without giving up your Privacy," *Proc. of HotMobile 2010*, Annapolis, MD, 2010.
[21] A. Dua, N. Bulusu, W. Hu, W. Feng, "Towards Trustworthy Participatory Sensing," *Proc. of USENIX HotSec*, August 2009.
[22] S. Saroiu, A. Wolman, "I Am a Sensor, and I Approve This Message," *Proc. of HotMobile 2010*, pages 37-42.
[23] P. Gilbert, L. Cox, J. Jung, D. Wetherall, "Toward Trustworthy Mobile Sensing," *Proc. of HotMobile 2010*, pps. 31-36, 2010.
[24] P. Gilbert, J. Jung, K. Lee, H. Qin, D. Sharkey, A. Sheth, L. Cox, "YouProve: Authenticity and Fidelity in Mobile Sensing," *ACM SenSys*, 2011.
[25] K. Zeng, K. Govindan, P. Mohapatra, "Non-cryptographic Authentication and Identification in Wireless Networks," *Wireless Communications 2010*, vol. 17, no. 5, pp. 56-62, Oct. 2010.
[26] N. Tippenhauer, C. Ppper, K. Rasmussen, S. Capkun, "On the Requirements for Successful GPS Spoofing Attacks," *Proc. of ACM CCS*, 2011.
[27] D. Faria, D. Cheriton, "No Long-term Secrets: Location Based Security in Overprovisioned Wireless LANs," *Proc. of HotNets-III*, 2004.
[28] Y. Wei, K. Zeng, P. Mohapatra, "Adaptive Wireless Channel Probing for Shared Key Generation," *Proc. of IEEE Infocom 2011*, 2011.
[29] Y. Zhang, Z. Li, W. Trappe, "Power-Modulated Challenge-Response Schemes for Verifying Location Claims," *IEEE Globecom 2007*, 2007.
[30] M. Kirkpatrick, E. Bertino, "Enforcing Spatial Constraints for Mobile RBAC Systems," *Proc. SACMAT'10*, 2010, pp. 99-108.
[31] K.B. Rasmussen, S. Capkun "Realization of RF distance bounding," *Proc. of 19th USENIX Security Symposium*, 2010.
[32] C. Cremers, K. B. Rasmussen, S. Capkun. "Distance Hijacking Attacks on Distance Bounding Protocols," *Cryptology ePrint Archive: Report 2011/129*, 2011.
[33] Skyhook Wireless, http://www.skyhookwireless.com/
[34] LOC-AID, Inc., http://www.loc-aid.com.
[35] Veriplace, Inc., http://veriplace.com.

Jack Brassil received the B.S. degree from the Polytechnic Institute of New York in 1981, the M.Eng. degree from Cornell University in 1982, and the Ph.D. degree from the University of California, San Diego, in 1991, all in electrical engineering.

Dr. Brassil has been with Hewlett-Packard Laboratories since 1999. He currently is a Research Scientist and Program Manager in Princeton, NJ. Before joining HP he held multiple research positions at Bell Laboratories in Murray Hill and Holmdel, NJ.

Pratyusa K. Manadhata received his B.Tech. and Ph.D. degrees in computer science from IIT Kanpur and Carnegie Mellon University, respectively. He is a researcher in HP's Cloud and Security Lab with a current focus on big data analytics for security. Before HP he spent 2 years at Symantec Research Labs building big data analytics systems for malware detection.

Ravi A. Netravali is currently a Ph.D. Student at MIT in the Computer Science and Artificial Intelligence Lab (CSAIL). Prior to joining MIT, Ravi received his BS degree in Electrical Engineering from Columbia University. He completed two summer research internships at HP labs. Ravi is interested in Mobile Networks and Systems.