Identification of Fake Channel Characteristics using AUXILLARY Receiver in Wireless Transmission

S Pooja[1] Dr.R.Jegadeesan[2] K S Pavithra[3] AMounikasri[4]
[1,3,4]Student [2]Assistant Professor
[1,2,3,4]Department of Computer Science Engineering
[1,2,3,4]RMK Engineering College, Chennai, India

Abstract— In remote systems, area refinement intends to identify area changes or encourage confirmation of remote clients. To accomplish area qualification, late research has concentrated on exploring the spatial uncorrelation property of remote channels. In particular, contrasts in remote channel qualities are utilized to recognize areas or distinguish area changes. Nonetheless, we find another assault against all current area qualification approaches that are based on the spatial uncorrelation property of remote channels. In such an assault, the foe can without much of a stretch conceal her area changes or imitate developments by infusing counterfeit remote channel qualities into an objective recipient. To protect against this assault, we propose an identification procedure that uses an assistant recipient or reception apparatus to distinguish these phony channel qualities. We likewise talk about such assaults and comparing barriers in OFDM frameworks. Test comes about on our USRP-based model demonstrate that the found assault can create any coveted channel trademark with a fruitful likelihood of 95.0% to vanquish spatial uncorrelation based area refinement plans and our novel location strategy accomplishes a discovery rate higher than 91.2% while keeping up a low false alert rate.

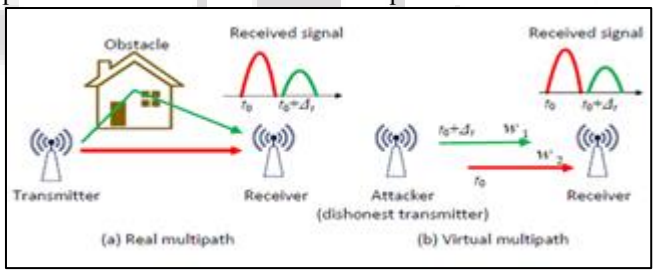
Key words: Channel Impulse Response, Multipath, Security, MIMO, OFDM

I. INTRODUCTION

Location distinction in wireless networks aims to detect a wireless client's area change, development or encourage location based confirmation. Upholding area refinement is imperative for some remote applications [1], [2]. For instance,

- Wireless sensor systems are generally used to screen an objective territory by detecting the physical or natural conditions (e.g., temperature, sound, and weight). Overseers of the sensor systems might want to uphold area refinement to keep an unapproved individual from moving the sensors from the region of intrigue.
- Wireless systems are defenseless against Sybil assaults because of the communicate idea of the remote medium [3]. Here, an enemy fashions a significant measure of phony client identities to fool a networked system. Location distinction can tell regardless of whether all characters are begun from a similar area, and therefore identify such assaults

In our study, however, we discover a new attack against all existing location distinction approaches built on the spatial uncorrelation property of wireless channels .By launching such an attack, the adversary can generate any chosen wireless channel characteristics at a target receiver to deteriorate the location distinction capability of the receiver. The key idea of the discovered attack is to create a virtual multipath channel as undetectable camouflage to make the receiver believe a specified channel characteristic chosen by the attacker. To demonstrate the virtual multipath channel, we first explain the multipath effect, which is the fundamental reason for the spatial uncorrelation property. Wireless signals normally propagate in the air through multiple paths due to obstacle reflection, diffraction, and scattering [1]. Therefore, for wireless signals sent from different locations, the receiver can observe different channel characteristics from these signals, because they experience different multipaths and accordingly undergo different channel effects (e.g, power attenuation, phase shifting, and delay). To fool a receiver, the attacker needs to create an "artificial channel" that can exhibit a multipath propagation feature similar to the real-world multipath. We give an example to illustrate how the attacker can create such a channel. Figure 1(a) shows a simple real multipath scenario, where a signal sent by the transmitter travels on two paths, i.e., the reflection path and the direct path. At time t0, the receiver starts to receive the signal copy that travels on the direct path. The reflection path is longer than the direct path, and thus at a later time t0 + Δt, the receivers receives the aggregation of the signal copy from the direct path and the one from the reflection path.



Presently think about the situation in Figure 1(b): there is just a single direct way between the assailant (i.e., an untrustworthy transmitter) and the recipient, however the aggressor needs to influence the collector to trust that two ways exist like the genuine multipath proliferation appeared in Figure 1(a). To this end, the assailant sends the flag alone first. After length Δt, she superimposes a new flag duplicate onto the one as of now in transmission. The assailant scales both the first flag and the time-postponed duplicate by lessening factors w1 and w2 to impersonate the flag adequacy weakening caused by genuine ways. Subsequently, the recipient watches a conglomeration of one flag in addition to a period deferred duplicate, with each experiencing a specific sufficiency weakening, and along these lines feels that they are caused by the multipath impact. The case in Figure 1(b) accept that there exists just a single direct way between the aggressor and the beneficiary (i.e., no multipath impact is considered). By and by, the aggressor's made multipath flag is influenced by the genuine multipath impact too, and she ought to have an approach to

manage the effect of this genuine multipath. Our exploration uncovers that the aggressor can without much of a stretch accomplish this objective by figuring out existing remote channel estimation calculations and performing straight changes on the first flag. To safeguard against this assault, we propose an identification method using an assistant recipient (or receiving wire) at an alternate area to distinguish the virtual multipath channels and the phony channel attributes. Specifically, the assailant must specialty its transmitting sign to influence the objective recipient to trust a specific channel trademark. In any case, we demonstrate this made flag shows conflicting channel qualities to the assistant recipient. In view of this outcome, we make a guard conspire that does not require the recipients to have any earlier information about the genuine channel attributes amongst themselves and the transmitter.

II. PRELIMINARIES

In this section, we show how location distinction is usually enforced and introduce the prevalent algorithms that are used to estimate wireless channel characteristics.

A. Channel Impulse Response

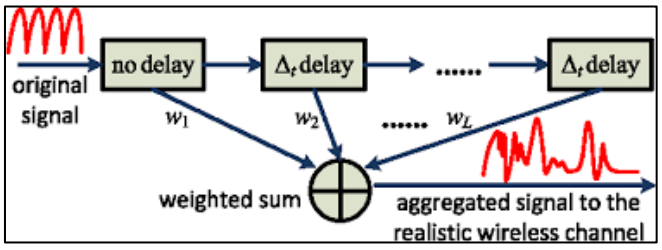
As talked about, a remote flag as a rule engenders noticeable all around along numerous ways because of reflection, diffraction, and scrambling. A recipient at that point gets various duplicates of the flag from various ways, every one of which has an alternate postponement because of the way it crosses. The got flag is the aggregate of these time deferred duplicates. Every way forces a reaction (e.g., deferral and lessening) on the flag going along it [1], and the superposition of all reactions between two hubs is alluded to as a channel motivation reaction [8]. Remote channels can be described by channel drive reactions. The multipath impacts of various remote connections are unique, as are the channel motivation reactions [1]. Because of this reason, a channel impulse response has been utilized to provide area refinement [1], [2]. Specifically, to decide whether the transmitter has changed its area, the beneficiary gauges the channel motivation reaction of a recently got flag and contrasts it and the past estimation result. The area change is distinguished if the distinction between the recently assessed channel motivation reaction and the past one surpasses a specific edge.

B. Estimating Channel Impulse Responses

Estimating channel impulse responses is a must-have function for most modern wireless systems [8], [9]. Note that the signal propagation paths are unresolvable (i.e., each multipath component signal cannot be extracted from the composite signal) if the differences between the arrival times of the signals traveling on these paths are much smaller than the symbol duration, which is the transmission time of a wireless physical-layer unit [8]. Hence, existing channel estimation algorithms assume a resolvable multipath, i.e., the arrival times of signal copies traveling on different paths are larger than the symbol duration. Channel impulse responses are usually estimated using training sequences [10]. Specifically, the transmitter sends a training sequence (i.e., a sequence of bits) over the wireless channel, while the receiver uses the same training sequence and the corresponding received signal samples to estimate the channel impulse response. The training sequence can be pre shared [10] or reconstructed from the received signal [1]. The physical layer channel estimation can be processed in either frequency (e.g. [1], [2]) or time domain (e.g., [10]), which are inter-convertible due to the linear relation between the two domains. In the following, we describe the channel estimation method in the time domain.

III. ASSUMPTIONS & ATTACK MODEL

The location distinction system consists of a transmitter and a receiver. Both are equipped with radio interfaces that can transmit and receive wireless signals. The receiver aims to verify whether or not the transmitter has changed location. Towards this goal, the receiver estimates the channel impulse response from a wireless signal received from the transmitter, and then compares it with the previous estimation results to generate a decision. To constantly enforce the location distinction, the receiver periodically sends an inquiry to the transmitter, and the transmitter responds to the inquiry by sending wireless signals back to the receiver. We assume that the transmitter is malicious and aims to hide her location change or impersonate movements while she is actually static. To achieve this objective, the transmitter attempts to mislead the receiver through creating a virtual multipath channel, which can fool the receiver to estimate a fake wireless channel impulse response chosen by the transmitter. We assume that the malicious transmitter knows the training sequence used for the channel estimation. We assume that the channel impulse response is stable in a short period of time (e.g., a packet duration), which is a common assumption for designing wireless communications. We further assume that the malicious transmitter knows the actual channel impulse response between herself and the receiver. This can be achieved by estimating the channel impulse response from the wireless signals (e.g., location distinction inquiries) emitted by the receiver.



IV. VIRTUAL MULTIPATH ATTACK

In this section, we describe how to create a virtual multipath channel to defeat location distinction algorithms. The attacker can launch two types of attacks. In a basic attack, the attacker can use any weights to craft a virtual multipath signal. This will fool the receiver to obtain random, incorrect estimates of the channel impulse response. In an advanced attack, with the knowledge of the real channel impulse response between herself and the receiver, the attacker is able to compute exact weights that make the receiver estimate the chosen channel impulse responses specified by the attacker. In the following

discussion, we focus on the advanced attack due to the more misleading nature of such attacks.

V. DEFENDING AGAINST THE VIRTUAL MULTIPATH ATTACK

Virtual multipath assailants can influence the collector to trust any channel trademark the aggressor picks. At the beneficiary, it appears that there is no real way to tell whether the flag experiences genuine or virtual multipath situation. Thus, existing area refinement techniques based after recognizing areas from channel qualities (e.g., [1]– [3], [6]) will be effectively crushed by virtual multipath assaults. The instinct behind our safeguard methodology is that no one can make one key to open two unique entryways. At the end of the day, if a recipient can't tell whether there is an assault or not, perhaps a moment beneficiary can. Subsequently, the proposed approach makes use of an

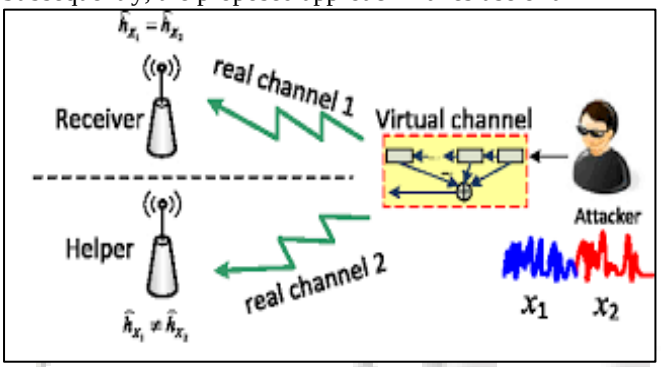


Fig. 5: Defense against virtual multipath attacks

auxiliary receiver or antenna, which we refer to as a helper. The helper is placed more than half a wavelength away from the receiver to ensure a distinct channel characteristic. We let the receiver use two different training sequences x1 and x2 to estimate the channel impulse response alternatively. Without loss of generality, we assume that the receiver uses x1 to estimate the channel from the first transmission, and uses x2 to estimate the channel from the second transmission. We discover that for both transmissions, at the receiver, the virtual channel created by a malicious transmitter (i.e., the attacker) can result in the same estimated channel impulse responses (equal to the one chosen by the attacker). However, at the helper, the virtual channel leads to different estimated channel impulse responses. We summarize the defense approach in Figure 5. The reason that the attacker cannot fool both the receiver and the helper is detailed next.

A. *Defense analysis*

Let h denote the real channel impulse response between the attacker and the receiver. For the first transmission, the attacker must solve the weights, so that the equation h∗xa1 = ha∗x1 hold and the receiver will obtain ha as the channel impulse response, where xa1 is the aggregated signal with weighted time-delayed copies of the training sequence x1. Let h help denote the real channel impulse response between the attacker and the helper. The corresponding signal received by the helper can be represented as h help ∗xa1. Thus, the channel impulse response ˆ hhelp1 estimated by the helper can be solved from the equation that ˆ hhelp1 ∗x1 = h help ∗xa1, and we have ˆ hhelp1 = (X1HX1)−1X1H(h help ∗xa1), (4) where X1 is a To eplitz matrix of x1. For the second transmission, both the receiver and the helper use the training sequence x2 to estimate the channel. Similarly, to fool the receiver, the attacker must generate another weights w2, so that the corresponding aggregated signal xa2 makes the equation h∗xa2 = ha ∗x2 hold. The corresponding channel impulse response ˆ hhelp2 estimated by the helper is ˆ hhelp2 = (X2HX2) −1X2H(h help ∗xa2), (5) where X2 is a to eplitz matrix of x2. Note that for both transmissions, the channel impulse response estimated by the receiver are always the same, because the weights are "customized" so that the receiver will obtain the attacker's chosen channel impulse response after the channel estimation. However, from Equations 4 and 5, we can see that the first estimated channel impulse response ˆ hhelp1 is not necessarily equal to the second estimated channel impulse response ˆ hhelp2, because X1 6= X2. This means the attacker cannot fool the receiver and the helper at the same time. Thus, if the successive estimated channel impulse responses show dramatic changes in a short time at the helper, the helper then triggers an alert at the receiver regarding the existence of potential virtual multipath attacks. In practice, the helper may use a threshold to enforce the detection. If ||ˆ hhelp1 − ˆ hhelp2|| is larger than the threshold, then the attack is assumed. The threshold can be selected based on the empirical studies to achieve optimized detection accuracy. In Section 7.4, we show an example of the threshold selection. Note that in the defense system, the helper and the receiver can switch their roles, i.e., if the attacker attempts to fool the helper instead of the receiver, the receiver will estimate two different channel impulse responses and therefore detect such an attack.

1) *Attackers with Helper*

The attacker may also bring a second transmitter to confuse the receiver. Figure 6 shows such a scenario. We refer to the attacker's second transmitter as the attacker's helper. Let h11, h12, h21, h22 denote the channel impulse responses between the attacker and the receiver, the attacker and the receiver's helper, the attacker's helper and the receiver, and the attacker's helper and the receiver's helper, respectively. To successfully launch the virtual channel attacks without being detected, the attacker must generate the same channel impulse response at the receiver's helper for both transmissions. Let h help denote such a channel impulse response. Further let ha denote the one that the attacker expects to generate at the receiver for both transmissions. The attacker needs to make the following equation hold:

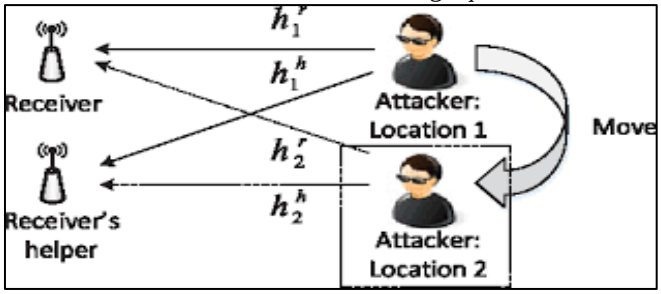


Fig. 6: The attacker also brings a second transmitter to confuse the receiver

$$\begin{aligned} h11 \ast xa1 + h21 \ast xh1 &= ha \ast x1 \\ h12 \ast xa1 + h22 \ast xh1 &= h\ help \ast x1 \\ h11 \ast xa2 + h21 \ast xh2 &= ha \ast x2 \end{aligned} \quad (6)$$

$$h12 * xa2 + h22 * xh2 = h\,help * x2$$

where xa1, xh1, xa2, and xh2 are the actual signals to be transmitted by the attacker and her helper for the first and second transmissions. To break the proposed defense, the attacker must solve them from Equation 6. This implies that h11, h12, h21, h22 should be all available to the attacker. Otherwise, the linear system lacks necessary coefficients to generate solutions. However, the acquisition of h12 and h22 will impose difficulty for the attacker, because the receiver's helper can be designed passive, i.e., it receives wireless signals but doesn't actively send out wireless signals to the channel. Due to the close proximity, the receiver can communicate with its helper through the cable connection or internal circuit. A passive helper of the receiver eliminates the chance for the attacker to extract the channel impulse responses based on heard wireless signals.

2) Defense Discussion

The receiver can normally use one passive helper, i.e., a secret wireless tap ,to detect the attacks. The exception happens when the attacker knows all channel information from her and her helpers to the receiver's passive helper (by placing a spy node co-located with or extremely close to the receiver's helper), which is in fact a very harsh requirement for the attacker. We point out that under this circumstance it is still feasible to detect virtual multipath attacks as long as the receiver has more helpers than the attacker. A significant advantage of the receiver over the attacker is that the receiver just needs to find contradiction to detect the attack; while the attacker has to know all channel information for signal manipulation to make sure no contradiction is found. In particular, when the receiver adds one more passive helper, it actually reduces the attack situation to the normal case. In order to beat the defense, the attacker must meet all the following requirements at the same time to beat the receiver: (1) add one helper, (2) add one spy node at the exact location of the receiver's new helper to know the channel information, (3) synchronize herself and all her helpers to transmit the manipulated signal at the physical layer symbol level. Hence, the attacker has much more costs to beat the receiver with more passive helpers.

VI. VIRTUAL MULTIPATH ATTACKS AND DEFENSES IN OFDM SYSTEMS

Orthogonal frequency-division multiplexing (OFDM) is a popular wireless communication scheme that encodes the digital signal using multiple sub-carrier frequencies. These subcarriers are normally narrow-band (e.g., 802.11 a/g physical layer advocates an OFDM sub-carrier bandwidth less than 0.5 MHz). Thus, OFDM systems are robust against channel fading caused by the multipath effect. For an OFDM system, the channel estimation is done by estimating the channel impulse response of each sub-carrier. Due to the lack of the multipath fading, the channel estimation result of each sub-carrier is a complex number rather than a vector, and the final channel estimation output of an OFDM system is formed by these complex numbers. In this section, we explore virtual multipath attacks and corresponding defenses in OFDM systems.

A. Attacks against OFDM Systems

The virtual multipath attacks can be easily extended to OFDM systems, because the mapping from the time-domain to frequency-domain is linear. The delay-and-sum process can be replaced by a much simpler procedure, in which the attacker multiplies chosen weights to sub-carriers. Specifically, let [h1,h2,...,hn] denote the actual channel characteristic between the attacker and the receiver, where hi is the channel characteristic of the i-th sub-carrier and n is the number of sub-carriers. Further let [x1,x2,...,xn] denote the training sequence encoded by the OFDM modulator, where xi is the i-th element of the encoded training sequence. The symbol received at the i-th carrier can be represented by yi = hixi. To fool the receiver to obtain a fake channel estimation result of [ha1,ha2,...,han], the attacker needs to make the equation hixai = haixi hold, where xai is the symbol to be transmitted by the attacker at the i-th sub-carrier. Thus, xai = haixi hi , and the weights that the attacker needs to multiply to sub-carriers are ha1 h1 , ha2 h2 ,..., han hn .

B. Defenses in OFDM systems

Despite the ease for an attacker to extend virtual multipath attacks to OFDM systems, as described above, there are no straightforward ways to extend the previously discussed detection approach to these systems, because the channel estimation of an OFDM system is significantly different from that of a traditional communication system. Let hr i and hh i denote the actual channel characteristic between the attacker and the receiver and between the attacker and the helper, respectively. Let xi1 and xi2 denote the i-th element of the first and second training sequences. Let xai1 and xai2 denote the symbol to be transmitted by the attacker at the i-th sub-carrier in the first and second transmissions. Further let hr ai and hh ai denote the fake channel estimation results that the attacker would like to generate at the i-th sub-carrier of the receiver and the helper. The conditions for the attacker to launch the attack without being detected are summarized as

Thus, when the attacker causes the receiver to observe the same channel estimation results for the first and second transmissions, the two channel estimation results at the helper side are also the same. Therefore, the virtual multipath attack in OFDM systems cannot be detected by the previously proposed regular defense, which just observes the difference of two channel estimates at the helper side for two transmissions with different training sequences. However, we identify alternative ways to close the loophole of the regular defense and defend against virtual multipath attacks in OFDM systems. We first categorize two typical objectives of attackers to confuse the location distinction:

1) Motion camouflage: The attacker is moving but she aims to deceive the receiver about the moving activities. Towards this end, the attacker makes the receiver believe that she is stationary by causing the estimated channel at the receiver to appear unchanged.
2) Immobility camouflage: When the attacker is stationary, she wants to make the receiver believe that she moves to a new location by changing the estimated channel at the receiver. The typical example targeting this

objective is the Sybil attack, in which the attacker pretends to change her location and therefore identity while she indeed just changes the channel between herself and the receiver, as the receiver will observe differing channels between transmitters in different locations. In practice, the two objectives may happen alternatively. For attacks against OFDM systems, we propose a corresponding defense strategy for each attack goal.

VII. EXAMPLE ATTACKS

We examine three example attacks:

1) injecting a randomly chosen channel impulse response into the receiver,
2) reproducing a same channel impulse response in the CRAWDAD data set; and
3) mimicking another location while hiding the true location. For all three attacks, we place the transmitter at location 2 shown in Figure 9.

A. Generating a Random Channel Response

First we show an attack with intent to generate a random channel impulse response. Figure 13 plots the real channel impulse response between the transmitter and the receiver, the channel impulse response chosen by the attacker, and the estimated channel impulse response at the receiver. The y-axis and the x-axis indicate the power gain and the relevant path respectively. We can see that the chosen channel impulse response and the estimated one are very similar to each other, but both of them significantly deviate from the real channel. The Euclidean distance between the chosen channel and the real channel is 0.3025, whereas that between the chosen channel and the estimated channel is as small as 0.0686.

B. Replicating a Same Channel Response in a Different Building

In the second example, an attacker aims to generate a channel impulse response in our office building such that the generated channel impulse response is exactly the same as one in the CRAWDAD data set, which was collected in an office building in the University of Utah. We note our USRP system is different from the CRAWDAD measurement system, Sigtek model ST-515, which has a much higher bandwidth (40 MHz) than the USRP (10 MHz). Therefore, the CRAWDAD measurement system can observe richer multipaths. Nevertheless, even with a relatively low-end USRP, we can still duplicate the resolvable paths in a channel impulse response measured in the CRAWDAD data set.

Specifically, we select one channel impulse response (between nodes 14 and 43) from the CRAWDAD data set and we plot it as "CRAWDAD channel" in Figure 14. We can see that this channel impulse response carries three peaks and thus exhibits three resolvable multipath. We launch the virtual multipath attack to make a replica of the same three resolvable multipath observed at the receiver in our experiment, which is shown as "Crafted channel" in Figure 14. The attack's crafted channel impulse response of the resolvable multipaths closely matches the CRAWDAD channel response and their Euclidean distance is as small as 0.0036.

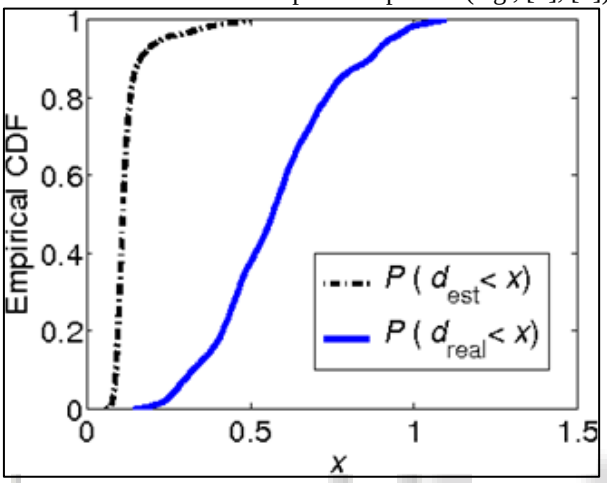
1) Actual Location Mimicking

In the third example, the attacker performs actual location mimicking, mimicking location 1 from location 2 shown in Figure 9. The attacker first records the real channel impulse response between herself and the receiver when she is at location 1, and then mimics this obtained channel impulse response when it moves to location 2. Figure 15 plots the real channel impulse responses between the transmitter and the receiver when the transmitter is at location 1 and 2 respectively, as well as the estimated channel impulse response at the receiver when the attacker performs the attack. We can see that in normal situation, the real channels between the attack and the receiver when the attacker is at location 1 and location 2 are quite different, and the Euclidean distance between them is 0.5290. However, when the attacker launches the virtual multipath attack at location 2, the estimated channel at the receiver is quite close to the real channel between the attacker and the receiver when the attacker is at location 1, and the Euclidean distance between the two channels turns to as small as 0.0964. Therefore, the attacker is able to effectively make the receiver believe that she is at location 1 while she is actually at location 2.

C. Overall Attack Impact

To examine the overall attack impact, we perform the following experiment. For each location in Figure 9, we estimate the channel impulse responses during a short time window (around 10 – 30 seconds). For each estimates, we perform 100 trials, and in each trial we randomly generate a length-5 vector whose elements range between 0 and 1. This vector is used as the attacker's chosen channel impulse response. We then launch the virtual multipath attack and record the Euclidean distance dreal between the chosen channel impulse response and the pervious channel impulse response estimated in the absence of the attacks (i.e., the real channel response), and also record the Euclidean distance dest between the chosen one and the channel impulse response estimated under the attacks. We repeat the same experiment for the other 9 locations. Ideally, a successful attacker should have a large value of dreal (indicating that the attacker's chosen channel significantly differs from the real channel) and a small value of dest (indicating that the attacker's chosen channel is close to the receiver's estimated channel). Denoted by P(dreal < x) and P(dest < x) the empirical CDFs of dreal and dest, respectively. Figure 16 shows P(dreal < x) and P(dest < x) for $0 \leq x \leq 1.5$. We can see that dest is less than 0.25 with probability 95.0%, dreal is larger than 0.9 with probability 95.0%. This means that dreal is much larger than dest with high probability, therefore the attacker can drag the estimated value of channel impulse response far away from its true value, and make it very close to her specified one. Existing schemes in general compare the difference between the receiver's current estimated channel and previous reference channel with a threshold to check a location change [1], [2]. Since our attacker can inject any random channel impulse response into the receiver with a very high accuracy, the performance of existing location distinction schemes can be significantly degraded by the virtual multipath attack. For

example, given a threshold set less than 0.5 for location change detection in our system, when the attack is launched, the receiver will think that the transmitter moves because all the differences between the estimated channel in the presence of the attack and the reference channel (attack-free channel) exceed the threshold of 0.5. However, the estimated channel and the real channel are actually measured at the same location, and thus the location distinction false alarm rate is raised to 100% under the virtual multipath attack. Similarly, the virtual multipath attack can also easily defeat any method verifying that nodes are from different locations based on examining the difference of their channel impulse responses (e.g., [3], [6]).



VIII. Conclusion

We identified a new attack against existing location distinction approaches built on the spatial uncorrelation property of wireless channels. By launching such attacks, the attacker can create virtual multipath channels to deteriorate the location distinction capability of a target receiver. To defend against this attack, we proposed a detection technique that utilizes a helper receiver to identify the existence of virtual channels. We also explored virtual multipath attacks and corresponding defenses in OFDM systems. We performed real-world evaluation on the USRP platform running GNU Radio. The experimental results demonstrated both the feasibility of the virtual multipath attack and the effectiveness of the defense approach.

References

[1] A. Alves et al. Web services business process execution language, version 2.0. OASIS Public Draft. Availiable at http://docs. oasis-open.org/wsbpel/2.0/, Nov. 2006.
[2] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing, Mar. 2004.
[3] U. Bellur and S. Bondre. xSpace: a tuple space for XML and its application in orchestration of web services. In Proceedings of the 21st ACM symposium on Applied computing – SAC'06, pages 766–772, 2006.
[4] A. N. Bessani, E. P. Alchieri, M. Correia, and J. S. Fraga. DepSpace: A Byzantine fault-tolerant coordination service. In Proceedings of the 3rd ACM SIGOPS/EuroSys European Systems Conference - EuroSys' 08, pages 163–176, Apr. 2008.
[5] A. N. Bessani, M. Correia, J. S. Fraga, and L. C. Lung. Sharing memory between Byzantine processes using policy-enforced tuple spaces. In IEEE Transactions on Parallel and Distributed Systems, 2008. to appear.
[6] M. Bichier and K.-J. Lin. Service-oriented computing. IEEE Computer, 39(3):99–101, Mar. 2006.
[7] D. Bright and G. Quirchmayr. Supporting web-based collaboration between virtual enterprise partners. In Proc of the 15th International Workshop on Database and Expert Systems Applications, 2004.
[8] D. Burdett and N. Kavantzas. The WS-Choreography model overview. W3C Draft. Available at http://www.w3.org/TR/ ws-chor-model/, Mar. 2004.
[9] L. F. Cabrera et al. Web Services Coordination Specification - version 1.0. Available at http://www-128.ibm.com/developerworks/ library/specification/ws-tx/, 2005.
[10] G. Cabri, L. Leonardi, and F. Zambonelli. Mobile agents coordination models for Internet applications. IEEE Computer, 33(2):82–89, Feb. 2000.
[11] N. Carriero and D. Gelernter. How to write parallel programs: a guide to the perplexed. ACM Computing Surveys, 21(3):323–357, Sept. 1989.
[12] M. Castro and B. Liskov. Practical Byzantine fault-tolerance and proactive recovery. ACM Transactions Computer Systems, 20(4):398–461, Nov. 2002.
[13] T. Dierks and C. Allen. The TLS Protocol Version 1.0 (RFC 2246). IETF Request For Comments, Jan. 1999.
[14] C. Dwork, N. A. Lynch, and L. Stockmeyer. Consensus in the presence of partial synchrony. Journal of the ACM, 35(2):288–322, Apr. 1988.
[15] F. Favarim, J. S. Fraga, L. C. Lung, and M. Correia. GridTS: A new approach for fault tolerant scheduling in grid computing. In Proceedings of the 6th IEEE International Symposium on Network Computing and Applications - NCA'07, pages 187–194, July 2007.
[16] D. Gelernter. Generative communication in Linda. ACM Transactions on Programing Languages and Systems, 7(1):80–112, 1985.
[17] M. Herlihy and J. M. Wing. Linearizability: A correctness conditionfor concurrent objects. ACM Transactions on Programing Languagesand Systems, 12(3):463–492, July 1990.
[18] L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem.ACM Transactions on Programing Languages and Systems,4(3):382–401, July 1982.
[19] R. Lucchi and G. Zavattaro. WSSecSpaces: a secure data-driven coordination service for web services applications. In Proceedings of the 19th ACM Symposium on Applied Computing – SAC'04, pages 487–491, Mar. 2004.
[20] Z. Maamar, D. Benslimane, C. Ghedira, Q. H. Mahmoud, and H. Yahyaoui. Tuple spaces for self-coordination of web services.In Proceedings of the 20th ACM Symposium on Applied computing –SAC'05, pages 1656–1660, 2005.
[21] N. H. Minsky and V. Ungureanu. Law-governed interaction: a coordination and control mechanism for heterogeneous distributed systems. ACM Transactions

on Software Engineering and Methodology, 9(3):273–305, July 2000.
[22] R. R. Obelheiro, A. N. Bessani, L. C. Lung, and M. Correia. How practical are intrusion-tolerant distributed systems? DI-FCUL TR 06–15, Dep. of Informatics, University of Lisbon, September 2006.
[23] Object Management Group. The common object request broker architecture: Core specification v3.0. OMG Standart formal/02-12-06, Dec. 2002.
[24] G. Papadopolous and F. Arbab. Coordination models and languages. In The Engineering of Large Systems, volume 46 of Advances in Computers. Academic Press, Aug. 1998.
[25] C. Peltz. Web services orchestration and choreography. IEEE Computer, 36(10):46–52, Oct. 2003.
[26] F. B. Schneider. Implementing fault-tolerant service using the state machine aproach: A tutorial. ACM Computing Surveys, 22(4):299– 319, Dec. 1990.
[27] E. J. Segall. Resilient distributed objects: Basic results and applications to shared spaces. In Proceedings of the 7th IEEE Symposium on Parallel and Distributed Processing – PDP'95, pages 320–327,Oct. 1995.
[28] P. Verissimo, N. F. Neves, and M. P. Correia. Intrusion-tolerant architectures: Concepts and design. In Architecting Dependable Systems, volume 2677 of LNCS. Springer-Verlag, 2003.
[29] V. Yegneswaran, P. Barford, and S. Jha. Global intrusion detection in the DOMINO overlay system. In Proceedings of the 11th Network and Distributed Security Symposium – NDSS 2004, Feb. 2004.

