Signalprint-Based Intrusion Detection in Wireless Networks

Rob Mitchell[1], Ing-Ray Chen[1], and Mohamed Eltoweissy[2]

[1] Dept. of Computer Science, Virginia Tech
{rrmitche,irchen}@vt.edu
[2] Dept. of Electrical and Computer Engineering, Virginia Tech
toweissy@vt.edu

Abstract. Wireless networks are a critical part of global communication for which intrusion detection techniques should be applied to secure network access, or the cost associated with successful attacks will overshadow the benefits that wireless networks offer. In this paper we investigate a new scheme called Nodeprints to extend the existing centralized Signalprints design for authentication to a distributed voting-based design for intrusion detection. We analyze the effect of voting-based intrusion detection designs, the probability of an individual node voting incorrectly, the ratio of mobile nodes to base stations, and the rate at which nodes are compromised, on the system performance measured by the probability that the intrusion detection system yields a false result. We develop a performance model for evaluating our Nodeprints design and identify conditions under which Nodeprints outperforms the existing Signalprints design.

Keywords: intrusion detection, wireless networks, Signalprints, identity-based attacks, performance analysis.

1 Introduction

Over the past few years various techniques have been proposed in the literature to authenticate mobile terminals in wireless networks. In particular fingerprinting techniques [1,3,4,8,9] based on side channel data [7] from the physical layer such as signal strength or signal phase and from the link layer such as timing or protocol have been developed to authenticate mobile terminals [10] Signalprints techniques [5] utilize signal metadata, specifically sequences of received signal strength indication (RSSI), collected to authenticate a reported identity and to deal with identity based attacks in wireless networks. Our work extends the existing centralized Signalprints design for authentication into a distributed voting-based design for detecting malicious nodes in wireless networks, recognizing that it is critical to apply intrusion detection techniques to secure network access, or the cost associated with successful attacks will overshadow the benefits that wireless networks offer.

Specifically, in this paper we propose, investigate and analyze a new intrusion detection design based on Signalprints, which we call Nodeprints, for securing

Q. Gu, W. Zang, and M. Yu (Eds.): SEWCN 2009, LNICST 42, pp. 77–88, 2010.

user access to wireless networks. Our Nodeprints design is fully distributed, thus eliminating the single point of attack or failure present in the Signalprints design. Moreover, we show that our Nodeprints design utilizing voting outperforms the existing Signalprints design in terms of the probability that the intrusion detection system yields a false result.

The rest of the paper is organized as follows. Section 2 surveys related work. Section 3 discusses system model, our Nodeprints intrusion detection design, performance model, as well as performance metrics used to evaluate the performance of the proposed Nodeprints intrusion detection design compared with the existing Signalprints design. Section 4 evaluates Nodeprints intrusion detection based on analysis. Finally, Section 5 summarizes the results and outlines some future research areas.

2 Related Work

Fingerprinting [1,3,4,8,9] utilizes side channel data to identify a terminal in a wireless network. The most readily-available side channel data is signal strength; this is typically measured in decibels referenced to one milliwatt (dBm). The most basic technique for fingerprinting involves a single node correlating historical signal strength data with a sample during authentication. This is a local, or host-based, signal strength approach. Patwari and Kasera enhance this basic technique by leveraging a time series of signal strength data to enhance authentication in [9]. They incorporate mobility into their design.

Desmond, et al. enhance authentication with another fingerprinting technique by analyzing 802.11 probe request frames [4]. This design associates nodes with a tuple of (Architecture, Wireless Network Interface Card Driver, Operating System). A major drawback of this approach is that it does not work with a homogeneous or near-homogeneous system. Nodes cannot be uniquely identified; they can only be categorized as having a certain configuration. A compromised node would defeat this countermeasure. This design is a passive countermeasure; it does not introduce any additional traffic into the network.

Crotti, et al. proposed a fingerprinting technique by using the premise of a traffic analysis attack as a countermeasure for authentication and availability attacks [3]. Specifically, they prosecute the size, interarrival time and sequence of datagrams to fingerprint/profile a flow of data. This is a passive technique.

Signalprints techniques deriving from fingerprinting have been proposed [5] to deal with identity based attacks specific to wireless networks. The Signalprints design uses signal metadata collected at multiple base stations. A centralized authentication server then takes those measurements to authenticate a reported identity.

Our work extends the Signalprints design to a distributed voting-based intrusion detection design for detecting malicious (or compromised) nodes to secure network access by exploring the tradeoffs between risks and rewards associated with distributed Intrusion Detection System (IDS) designs [6]. Our voting-based IDS design derives from a cooperative IDS design [2] which requires each node to preinstall a host-based IDS. Our design does not require a preinstalled host-based IDS to provide judgments if a neighbor node is compromised. Instead, as in the Signalprints design, a node utilizes physical-layer signal strength metadata as the means for detecting whether or not a neighbor node is compromised.

3 Nodeprints Intrusion Detection

3.1 System Model

While the Signalprints design [5] proposed for authentication assumes that base stations are trusted/known-good components, it treats terminals with unfettered skepticism. Like the Signalprints design, our Nodeprints intrusion detection design assumes that base stations are trusted/known-good components and treats terminals with skepticism. However, Nodeprints intrusion detection uses data from mobile nodes in the intrusion detection function. While these terminals are the target of the IDS, Nodeprints intrusion detection seeks to capitalize on the fact that not all terminals are compromised. Nodeprints is further distinguished by its voting based, distributed design. Instead of forwarding sample data, participants cast yes/no votes; this eliminates a centralized component as a single point of failure.

This threat model assumes that the adversary has some amount of physical access to the facility, but the access is not complete. In a low-tech attempt to spoof the RF fingerprint of a target, we can assume the adversary can access a hallway or office adjacent to the target. It is reasonable to assume the adversary cannot collocate precisely with the target.

This threat model also assumes that the adversary is using a discreet, unsophisticated antenna design, for example a single whip antenna. Contrast this with techniques that prosecute a much more focused attack on the physical layer by spoofing RF fingerprints with antenna arrays (collocated or geographically dispersed), parabolic designs and beamforming techniques.

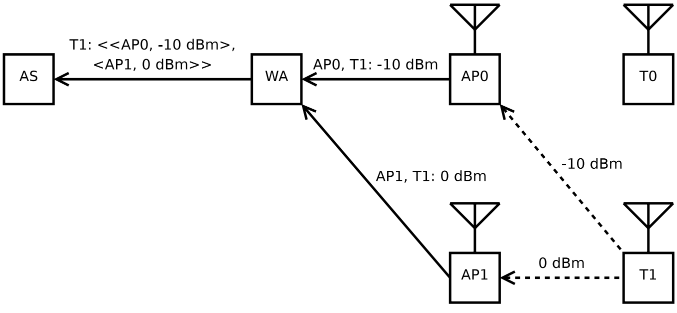

Fig. 1. Flow of information in the Signalprints design

3.2 Design

Our proposed IDS design is comprised of three functions: evaluation, detection and reaction. Figure 1 illustrates the flow of information in the Signalprints design. The "T" nodes are mobile terminals, who do not contribute data to the IDS. The "AP" nodes are access points or base stations, who are the only data contributors to the IDS in Signalprints. The "WA" node is a wireless appliance and the "AS" node is an authentication server which are discussed later.

In the Signalprints design in [5], base stations collect signal metadata and forward it to the wireless appliance (WA) which collates it (correlates metadata from independent base stations to the same authentication event). Specifically, base stations monitor the RSSI component of streams.

For detection, the WA applies "min-match" and "max-match" primitives to detect bad nodes prosecuting various authentication or availability attacks, e.g., masquerade attacks, in which a bad node impersonates a single node, or Sybil attacks, in which a bad node impersonates many nodes. The "min-match" and "max-match" primitives are implemented as $minMatches(S_1, S_2, \epsilon)$ and $maxMatches(S_1, S_2, \epsilon)$ where S_1 and S_2 are two Signalprints of interest. The output of these functions is an integer which conveys the number of positions which the Signalprints differ by at least ϵ and at most ϵ, respectively. Fewer "min-matches" and more "max-matches" correlate with greater confidence that the same terminal generated the two Signalprints which, in turn, strengthens authentication by fusing user credentials with user profile.

Figure 2 illustrates the flow of information in our Nodeprints intrusion detection design. The system selects m voters (base stations or mobile terminals) nearest to the mobile host (MH) and each will cast a yes/no vote; $m = 3$ and the MH is legitimate in this scenario. The network elects a coordinator to manage the voting process following an election protocol. The coordinator is selected randomly among the m vote-participants so that no particular node will always be the coordinator– this eliminates a single point of failure or attack. The coordinator must let all voters know each others' identities so that each voter will multicast its yes/no vote to other voters. At the end of the voting process, all voters will know the same result– either the MH is authenticated or evicted based on the majority vote. We contrast this distributed design with the single point of failure contained in the WA/AS in the Signalprints design. Note that T0 casts an erroneous vote, either accidentally or nefariously. Our analysis considers both of these situations.

While the Signalprints design uses "min-match" and "max-match" primitives to detect bad nodes in a design centralized at the wireless appliance or authentication server, our Nodeprints intrusion detection design uses a more generic voting-based algorithm which could be decentralized since the detection function of the IDS is distributed. Each vote-participant casts a yes or no vote based on the signal metadata collected with respect to the MH.

3.3 Performance Model

We develop a performance model to measure the performance of our Nodeprints intrusion detection design against the existing Signalprints design. Figure 3 lists

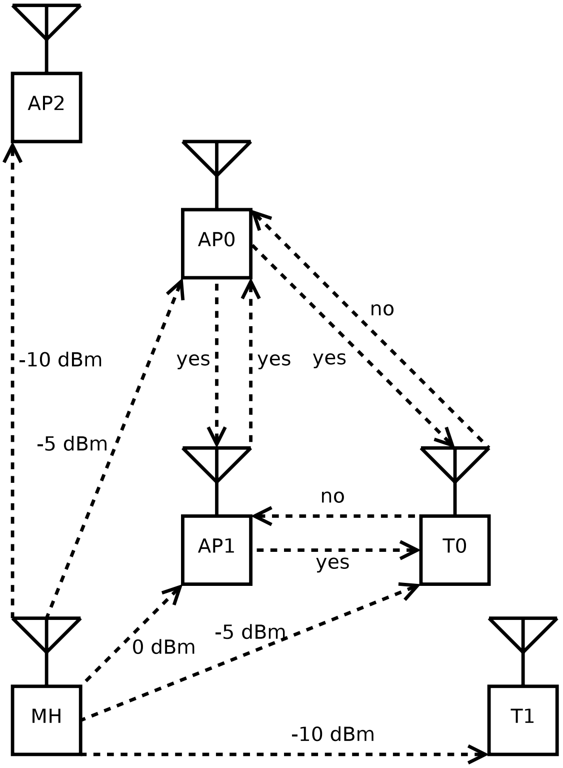

Fig. 2. Flow of information in the Nodeprints intrusion detection design

m	number of voters participating in an IDS detection event
N_{majority}	$\lceil \frac{m+1}{2} \rceil$
N_{bt}	number of bad terminals in the system
N_{gt}	number of good terminals in the system
N_{bs}	number of trusted base stations in the system
P_{fp}	probability of a false positive, a good node which the IDS detects as bad
P_{fn}	probability of a false negative, a bad node which the IDS detects as good
P_F	short for P_{fn} or P_{fp}
α	ratio of terminals to base stations
β	percentage of nodes that are captured
i	number of bad terminals voting, less N_{majority} in the special case of Figure 4
j	number of bad terminals voting in the general case of Figure 4
k	number of good terminals voting incorrectly in the general case of Figure 4
p	probability of a good node voting incorrectly

Fig. 3. Parameters

the model parameters used. For authenticating a terminal, our Nodeprints intrusion detection design involves m vote-participants selected out of those base stations and terminals reachable from a terminal node to be authenticated. If the majority out of these m vote-participants votes against the terminal node, then the terminal node is considered compromised. In our model, a *good node* is cooperative and protocol-compliant. *Bad nodes* are uncooperative (self-centered or greedy) or malicious; a bad node will always vote for a bad node and vote against a good node to facilitate attacks. When a good node casts a *correct vote*, it is the correct decision: it accurately reflects the reality of the situation. When good nodes cast *incorrect votes*, they don't do it with nefarious intentions. Rather, incorrect votes stem from error in the local IDS algorithm or the signal strength metadata that algorithm is using. A *false negative* result is one that comes from the distributed voting-based algorithm/cooperative IDS which incorrectly identifies a bad node as good. On the other hand, *false positives* mistakenly identify a good node as bad. A *false result* is a false negative or a false positive. In either case, the system fails to authenticate a reported identity. In the case of intrusion detection, the system fails to detect a compromised node.

In our model, as in the existing Signalprints design, base stations are trusted as a precondition. However, since we include terminals in intrusion detection, good nodes include both base stations and good terminals, while bad nodes include only bad terminals. The metrics used to evaluate the performance of our Nodeprints intrusion detection design against the baseline Signalprints design are P_{fn} and P_{fp}, which are the probabilities of false negative and false positive, respectively, in voting-based IDS.

The equation for P_{fn} or P_{fp} is given in Figure 4 which is the sum of two summations incorporating several system parameters, ambient conditions and indices/bounds of the model. The first parameter that system designers control is m; this is the number of voters participating in an IDS detection event. A higher value raises accuracy while a lower value economizes energy and channel overhead. N_{bs} is the other parameter that system designers control; it is the number of trusted base stations in the system. A higher value increases security

$$
\begin{aligned}
P_{\mathrm{fn}} &= P_{\mathrm{fp}} \\
&= \sum_{i=0}^{m-N_{\mathrm{majority}}} \left[\frac{\binom{N_{\mathrm{bt}}}{N_{\mathrm{majority}}+i} \cdot \binom{N_{\mathrm{gt}}+N_{\mathrm{bs}}}{m-(N_{\mathrm{majority}}+i)}}{\binom{N_{\mathrm{gt}}+N_{\mathrm{bt}}+N_{\mathrm{bs}}}{m}} \right] + \sum_{j=0}^{m-N_{\mathrm{majority}}} \\
&\left[\frac{\binom{N_{\mathrm{bt}}}{j} \cdot \sum_{k=N_{\mathrm{majority}}-j}^{m-j} \left[\binom{N_{\mathrm{gt}}+N_{\mathrm{bs}}}{k} \cdot p^{k} \cdot \binom{N_{\mathrm{gt}}+N_{\mathrm{bs}}-k}{m-j-k} \cdot (1-p)^{(m-j-k)} \right]}{\binom{N_{\mathrm{gt}}+N_{\mathrm{bt}}+N_{\mathrm{bs}}}{m}} \right]
\end{aligned}
$$

Fig. 4. Probability of a false result under Nodeprints intrusion detection

while a lower value economizes infrastructure cost. N_{bt}, N_{gt} and p are ambient conditions: the number of bad and good terminals in the system and the probability of a good node voting incorrectly, respectively. Systems designers may know what values are reasonable to expect for these items, but they can't choose them. i, j and k act as indices and lower and upper bounds of the summations in the model. i is the number of bad terminals voting, less N_{majority} in the special case of Figure 4. In the general case of Figure 4, j is the number of bad terminals voting and k is the number of good terminals voting incorrectly. The first summation is the special case; it aggregates the probability of a false result stemming from selecting a majority of bad nodes. That is, it is equal to the number of ways to choose a majority of bad nodes from the set of all bad nodes times the number of ways to choose a minority of good nodes from the set of all good nodes divided by the number of ways to choose m nodes from the set of all good and bad nodes. The second summation is the general case; it aggregates the probability of a false result stemming from selecting a majority of good nodes, some of which cast incorrect votes, coupled with selecting some number of bad nodes. That is, it is equal to the number of ways to choose a minority of bad nodes from the set of all bad nodes times the aggregate probability of a sufficient number of good nodes casting incorrect votes also divided by the number of ways to choose m nodes from the set of all good and bad nodes. The aggregate probability is a nested summation of the number of ways to choose a sufficient number of good nodes which cast incorrect votes and the remaining good nodes which cast correct votes.

4 Evaluation

In this section, we shall demonstrate improved performance of our Nodeprints intrusion detection design over the existing Signalprints design using mathematical analysis. With the baseline Signalprints design, there is only one source of error in the model: trusted base stations which vote incorrectly. With Nodeprints intrusion detection, there are three sources of error in the model: trusted base stations which vote incorrectly, good terminals which vote incorrectly and bad terminals. Our hypothesis is the increased number of terminals in the Nodeprints intrusion detection design will trade favorably with the increased error vectors in some scenarios.

4.1 Experimental Design

The effect of interest in this study is to identify design conditions under which Nodeprints intrusion detection outperforms the existing Signalprints design. There are three independent variables in this investigation: the treatment group, the capture ratio (β) and the probability of an incorrect vote (p). The first treatment group is the control group, which uses the Signalprints design; the second treatment group is the experimental group, which uses the Nodeprints intrusion detection design. The level of measurement is nominal in this case. The next independent variable, β, models the fraction of terminals that are bad. β's level of measurement is ratio; the true zero conveys no bad terminals. The final independent variable, p, models the probability that a good node votes incorrectly. p's level of measurement is ratio; no good nodes cast incorrect votes at the true zero. The dependent variable in this study, P_F is a ratio measurement. A true zero reading indicates the IDS performed correctly for the trial. N_{bs}, N_{bt}, N_{gt} and the ratio of terminals to base stations (α), which are highly covariant, are extraneous variables in this investigation. The level of measurement for the first three confounding variables, N_{bs}, N_{bt} and N_{gt} is interval. The final extraneous variable, α, models the ratio of terminals to base stations. α's level of measurement is ratio; the true zero conveys no terminals are present, only base stations.

4.2 Treatment Groups

Control Treatment. The control treatment consists of applying the existing Signalprints design to the model specified in Figure 4. Specifically, this means a configuration where $N_{\text{gt}} = N_{\text{bt}} = 0$. We exercise the independent variable p and the extraneous variable N_{bs}.

Experimental Treatment. The experimental treatment consists of applying the Nodeprints intrusion detection design to the model specified in Figure 4. We exercise the independent variables p and β and the extraneous variable α. We control N_{bs}, N_{bt} and N_{gt} by setting $N_{\text{bs}} = 3$.

4.3 Measurements

Evaluation of the false result equation shown in Figure 4 will yield measurement data. We vary the values of α, N_{bs}, p and β to test their effects. For the control treatment, pairs of p and P_F values output to a different file for each value of N_{bs}. For the experimental treatment, there are two sets of data: pairs of p and P_F values output to a different file for each value of β and pairs of p and P_F values output to a different file for each value of α.

4.4 Results

The following graphs plot the response of P_F with respect to p. In most cases, the graph of P_F is a sigmoid curve that reaches 0 and 1 as p approaches 0 and 1 respectively. This is consistent with how we expect a cumulative distribution function to behave. The exception to this is when $N_{\text{bs}} = 2$ in Figure 5.

Figure 5 presents the relationship of P_F to p in the Signalprints design for $p \in [0, 1]$ and $N_{\text{bs}} \in a_i = 2 + 4i$ where $i \in \{0, 1, 2, \ldots, 15\}$. The trend shows that the Signalprints design performs better for lower values of p and worse for higher values. The reason for this trend is that N_{bs} amplifies p. The more nodes there are in the network, the greater extent to which their incorrect vote probability will inform the result. In other words, the more data points there are in the

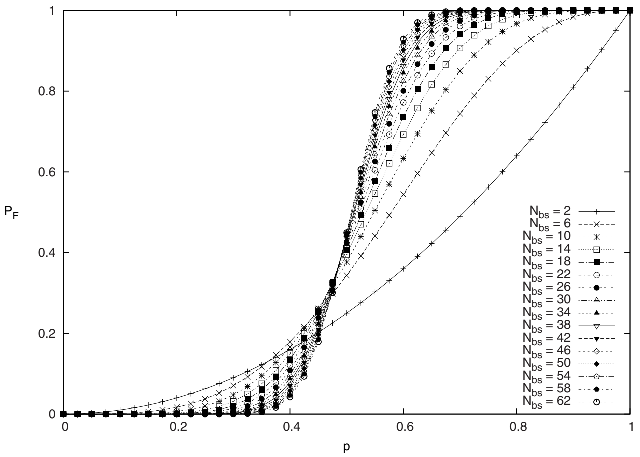

Fig. 5. P_F vs. p and N_{bs} under the Signalprints design

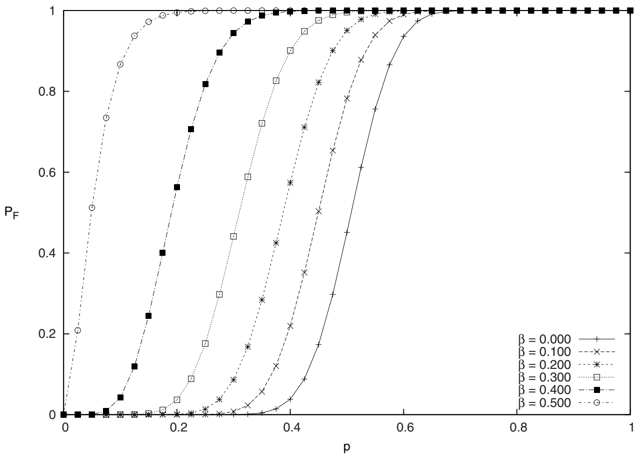

Fig. 6. P_F vs. p and β under the Nodeprints design

sample for the IDS, the more likely the system answer will adhere to the trend of individual voters.

Figure 6 conveys the relationship of P_F to p in the Nodeprints intrusion detection design for $p \in [0, 1]$ and $\beta \in \{0.000, 0.100, 0.200, 0.300, 0.400, 0.500\}$. The trend is for Nodeprints intrusion detection to perform better for lower values of β and worse for higher values. The reason behind this is that β models the node capture ratio. As more terminals are compromised, the more sense it makes not to incorporate them into the pool of voters.

Figure 7 presents the relatively small impact of α for $\beta = 0.100$. The trend is for Nodeprints intrusion detection to perform slightly better for lower values of α. The reason for this trend is that α amplifies β. The more terminals there

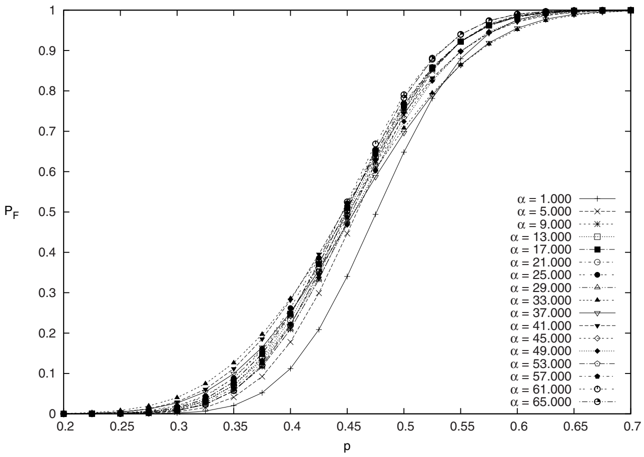

Fig. 7. P_F vs. p and α under the Nodeprints design

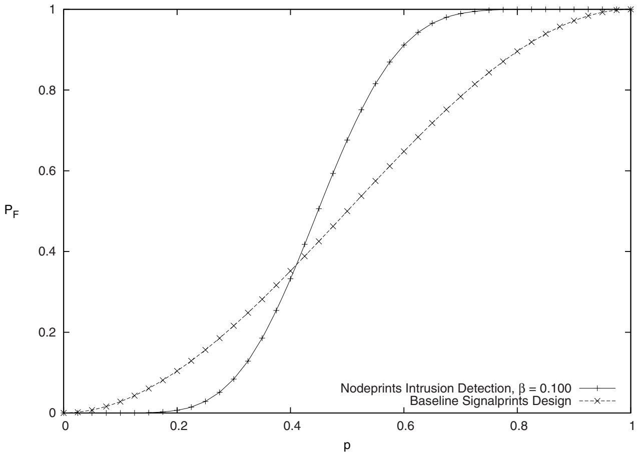

Fig. 8. Performance comparison of Nodeprints intrusion detection vs. baseline Signalprints design for $N_{\mathrm{bs}} = 3$

are in the network, the greater extent to which their capture ratio will introduce bad voters into the pool.

Figure 8 visualizes how Nodeprints intrusion detection offers an advantage over existing Signalprints design over a wide range of parameter values under $N_{\mathrm{bs}} = 3$ and $\beta = 0.100$. To correlate this with our model, $m = N_{\mathrm{bs}} = 3$ for the exiting Signalprints design and $m = N_{\mathrm{bs}} + \beta \cdot (N_{\mathrm{bs}} \cdot \alpha) + (1 - \beta) \cdot (N_{\mathrm{bs}} \cdot \alpha) = 15$ for Nodeprints. We observe that Nodeprints consistently outperforms Signalprints for $p \in [0, 0.400]$. It is reported [2] that any valid IDS must have its p value below 0.1. We see that for $p < 0.1$ our Nodeprints intrusion detection design significantly outperforms the existing Signalprints design.

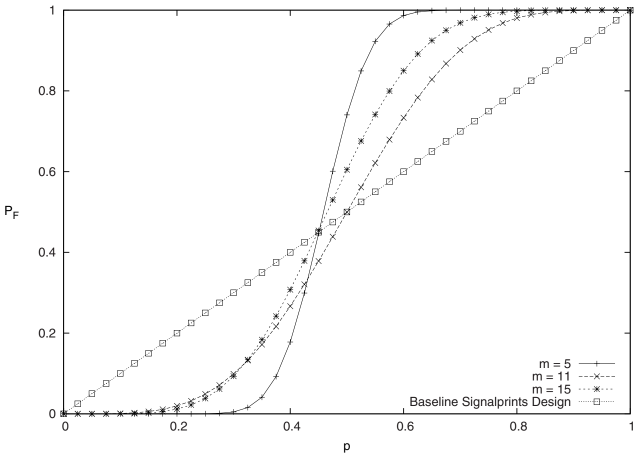

Fig. 9. Contrast Nodeprints with Signalprints with m varying for P_F vs. p

Lastly Figure 9 contrasts Nodeprints with Signalprints with m varying in the range of [5, 15]. We observe that as m increases, the cross-over value of p below which Nodeprints performs better than Signalprints increases, suggesting that Nodeprints will offer even more advantages over Signalprints when the node population is high under which the system can easily find many vote participants in a MH's neighborhood to determine whether the MH is compromised.

5 Conclusion

Signalprint-based intrusion detection contributes a novel design alternative to system designers putting together secure wireless networks. By testing the effect of system parameters such as α and β, system designers can identify situations where, when a cooperative IDS is a requirement, Nodeprints intrusion detection offers better performance than the existing Signalprints design. As a general guideline, smaller values of β and p are favorable to Nodeprints intrusion detection. Specifically, this investigation demonstrates that for $\beta \leq 0.100$ and $p \leq 0.400$, Nodeprints intrusion detection outperforms the existing Signalprints design. At the crossover point of around $p = 0.400$, the benefit of including additional metadata from untrusted terminals is overcome by the bias they impose on results.

A future investigation will consider the probability of a false result under Nodeprints intrusion detection as a function of time as terminals are identified as bad nodes by IDS and evicted from the system, and as a function of the node capture ratio. Also, future research will duplicate the results achieved in this paper in a simulation and/or experimental environment. Third, enhancing the yes/no mechanism to a yes/no/maybe or even finer granularity is worth investigating. Also, studying the trade associated with increased security and impact on network capacity or energy consumption is important. Finally, we have assumed the value of the p parameter is the same for all base stations and terminals. In heterogeneous environments this p value may vary among heterogeneous nodes, reflecting varying IDS capabilities of base stations and terminal nodes in the system. In the future we plan to test the effect of node heterogeneity on the false result probability.

References

1. Bratus, S., Cornelius, C., Kotz, D., Peebles, D.: Active Behavioral Fingerprinting of Wireless Devices. In: Proceedings of the First ACM Conference on Wireless Network Security, pp. 56–61 (2008)
2. Cho, J.H., Chen, I.R., Feng, P.G.: Effect of Intrusion Detection on Failure Time of Mission-Oriented Mobile Group Systems in Mobile Ad Hoc Networks. In: 14th IEEE Pacific Rim International Symposium on Dependable Computing, Taipei, Taiwan (December 2008)
3. Crotti, M., Dusi, M., Gringoli, F., Salgarelli, L.: Traffic Classification through Simple Statistical Fingerprinting. SIGCOMM Computer Communication Review, 5–16 (2007)
4. Desmond, L.C., Yuan, C.C., Pheng, T.C., Lee, R.S.: Identifying Unique Devices Through Wireless Fingerprinting. In: Proceedings of the First ACM Conference on Wireless Network Security, pp. 46–55 (2008)
5. Faria, D.B., Cheriton, D.R.: Detecting Identity-Based Attacks in Wireless Networks Using Signalprints. In: Proceedings of the 5th ACM Workshop on Wireless Security, pp. 43–52 (2006)
6. Frincke, D.: Balancing Cooperation and Risk in Intrusion Detection. ACM Transactions on Information and System Security, 1–29 (2000)
7. Le, T., Canovas, C., Clédière, J.: An Overview of Side Channel Analysis Attacks. In: Proceedings of the 2008 ACM Symposium on Information, Computer and Communications Security, pp. 33–43 (2008)
8. Pang, J., Greenstein, B., Gummadi, R., Seshan, S., Wetherall, D.: 802.11 User Fingerprinting. In: Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking, pp. 99–110 (2007)
9. Patwari, N., Kasera, S.K.: Robust Location Distinction Using Temporal Link Signatures. In: Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking, pp. 111–122 (2007)
10. Sheng, Y., Tan, K., Chen, G., Kotz, D., Campbell, A.: Detecting 802.11 MAC Layer Spoofing Using Received Signal Strength. In: INFOCOM 2008, pp. 1768–1776 (2008)