53rd IEEE Conference on Decision and Control
December 15-17, 2014. Los Angeles, California, USA

Physical Layer Methods for Privacy Provision in Distributed Control and Inference

Shalabh Jain, Tuan Ta and John S. Baras

Abstract— **Distributed control, decision and inference schemes are ubiquitous in many current technological systems ranging from sensor networks, collaborative teams of humans and robots, and information retrieval systems. Privacy, both location and identity, is critical for many of these systems and applications. The principal thesis investigated in this paper is that the utilization of physical layer methods and implementation techniques substantially strengthens privacy in the associated algorithms and systems. In fact it is argued that without the utilization of such physical layer methods it may be expensive to have provable levels of security in these systems. We analyze the performance of such physical layer techniques. We then utilize these techniques to provide provable privacy in distributed control, decision and inference algorithms. We demonstrate the results in context of distributed Kalman filtering. We develop useful metrics to measure privacy in these distributed systems. We investigate quantitatively the effects of privacy loss on the performance of the systems.**

I. INTRODUCTION

Advances in VLSI technology has led to a significant reduction in the size of wireless devices. This has led to a new paradigm of large scale systems comprising of small and distributed communicating devices. Such distributed networks have found applications in several aspects of human lives. This includes critical cyberphysical systems for monitoring and regulation of power grids, large scale intrusion detection systems or efficient systems for information retrieval.

The objective of such systems can broadly be classified as distributed processing of information for consensus or monitoring the state of the system. Significant research effort has been directed towards this, e.g. [1], [2]. However, the critical nature of the recent applications has introduced adversarial intent into such systems. Thus, recent research thrust has been towards modeling such systems with adversaries [3], [4]. The works in this direction can be broadly classified as systems that establish notions of trust [5], or design of robust distributed algorithms [6], or systems relying on cryptographic primitives [7].

In this paper, we investigate the direction of establishing trust in the network to exclude adversarial nodes. One promising idea for designing such a system is to form a trust hierarchy based on the quality of inputs from different nodes. Several works, [8], [9], argue the need for one level of the hierarchy acting as a reference to achieve consensus. Such reference nodes comprise the 'trusted core'. Other nodes use the trusted core as an anchor to evaluate the quality of received input. Along with robustness of the measurements from these nodes, it is required that nodes of the trusted core are reachable and the integrity of the measurements upon traversal through the network is verifiable. This typically requires the nodes to utilize a cryptographic framework. However for low power nodes, the cryptographic overhead may be undesirable. Additionally, such a framework introduces the complexity of key management in ad-hoc networks.

We argue that these requirements of a trusted core may alternately be fulfilled by ensuring privacy of the identity of trusted nodes. In the absence of knowledge of the identity of the trusted nodes, an adversary can at best choose a random subset of nodes to attack. Due to the sparsity of the trusted core, the number of trusted nodes that may be attacked by random selection is small enough to prevent significant disruption. In this paper, we utilize the physical layer authentication scheme from [10] to tag trusted messages from the trusted core. Using the stealth properties of the tagging scheme, we argue that the privacy of the trusted core is maintained. Further, we discuss the impact of the size of the trusted core on distributed Kalman filtering problem. We verify our assertions via MATLAB simulations.

The paper is organized as follows. In Section II, we highlight the requirements from a trusted core. In Section III, we describe the scheme of [10]. We describe our system and its security properties in Sections IV and V. In Section VI we illustrate the performance of the system via simulations.

II. APPLICATIONS OF TRUSTED CORE

We present the requirements of the trusted core as envisaged in some prior works [8], [9].

A. Distributed Kalman Filtering (DKF)

In the DKF application considered in [8], it was shown that a malicious adversary may drive the system to reach consensus to a false state. However, introduction of weights while computing the Kalman coefficient updates, where the weights represent the trust in a node may guarantee correct operation even in the face of Byzantine adversaries. The weights are derived using the reference state from a set of nodes assumed to be correct. The assumption here is that the reference is correctly generated (i.e. node is not compromised) and is propagated without modification (i.e. message integrity verification).

B. Trusted Core Properties

As evident from the example, availability of nodes and integrity of measurements during propagation are two fundamental features required from the trusted core. Typically,

Research supported partially by US Air Force Office of Scientific Research MURI grant FA9550-10-1-0573, and by National Science Foundation grants CNS 1018346 and CNS 1035655.

The authors are with Institute for Systems Research, and the Department of Electrical and Computer Engineering at the University of Maryland, College Park, MD 20742 {shalabh, tta, baras}@umd.edu

the integrity of the nodes can be guaranteed using hardware based checks. However, there is little that can be done to prevent the adversary performing co-located jamming of the wireless medium. Thus even under weak adversarial assumptions, it is difficult to guarantee availability.

Integrity during propagation can be ensured by using message integrity checks using low complexity crypto primitives. However, this requires the overhead of key management. It also introduces significant computational overhead and an increase in transmission bandwidth, both of which may be undesirable for the low power networks we consider.

A more subtle requirement may be to prevent leakage of the data from the trusted core as it may allow the adversary to adapt its behavior. This can be prevented by encrypting the entire message. This however, consumes significant power.

The properties described here may alternately be achieved by concealing the identity of the trusted nodes from an adversary. In the absence of knowledge about the specific location or identity of a trusted node, an adversary can neither jam its transmissions or corrupt its observations. Thus privacy of the identity and location of the nodes of a trusted core is sufficient for its functionality.

III. PHYSICAL LAYER METHOD

We utilize the physical layer authentication scheme in [10] to ensure privacy of the trusted core. Here we briefly present important aspects of their scheme and notation relevant to our discussion. For details, constraints and performance metrics of the system, the reader is referred [10].

Consider a system where the sender wishes to transmit a signal $\mathbf{s} = \{s_1, s_2, \ldots, s_L\}$ to the receiver with some additional information $\mathbf{t}$ to authenticate the sender. Let k be the shared key between the sender and the receiver. The sender generates the authentication tag as $\mathbf{t} = g(k, \mathbf{s})$. $g(\cdot)$ represents a 'secure' tagging scheme (e.g: keyed hash function). The sender superimposes the tag on the signal waveform to transmit $\mathbf{x} = \rho_s \mathbf{s} + \rho_t \mathbf{t}$, where $\rho_s, \rho_t \in (0, 1)$ represent the power allocation to the signal and tag.

Assume a Rayleigh block fading (slow fading) channel. The channel for the transmitted block is denoted by $h \sim CN(0, \sigma_h^2)$. CN denotes a circularly symmetric complex Gaussian variable. The receiver observes the block $\mathbf{y} = h \cdot \mathbf{x} + \mathbf{w}$, where $\mathbf{w} = \{w_1, \ldots w_L\}$ and $w_k \sim CN(0, \sigma_w^2), \forall k$. Using the estimation techniques highlighted in [10], the receiver recovers the transmitted signal $\hat{\mathbf{s}}$ and the expected tag $\hat{\mathbf{t}} = g(k, \hat{\mathbf{s}})$. The receiver authenticates the sender by verifying the presence of the tag in the residue

$$\mathbf{r} = \frac{1}{\rho_t}(\hat{\mathbf{x}} - \rho_s \hat{\mathbf{s}}). \tag{1}$$

The receiver obtains the test statistic τ by applying a matched filter to the residue with the estimated tag, $\tau = \hat{\mathbf{t}}^H \mathbf{r}$. The receiver performs a threshold test with hypotheses

$$\begin{aligned} H_0 &: \hat{\mathbf{t}} \text{ is not present in } \mathbf{r} \\ H_1 &: \hat{\mathbf{t}} \text{ is present in } \mathbf{r}. \end{aligned} \tag{2}$$

Assuming perfect channel estimation ($\hat{h} = h$) and tag estimation ($\hat{\mathbf{t}} = \mathbf{t}$), the statistic for the tagged and non tagged scenarios are

$$\begin{aligned} \tau|H_1 &= |\mathbf{t}_i|^2 + v, \\ \tau|H_0 &= \left(\frac{1-\rho_s}{\rho_t}\right) \mathbf{t}^H \mathbf{s} + v, \end{aligned} \tag{3}$$

where, conditioned on $\mathbf{t}$, $v \sim \mathcal{N}(0, L\sigma_w^2/\rho_t^2 |h|^2)$. Additionally, $E[\tau|H_0] = 0$, since we assume $E[\mathbf{s}^H \mathbf{t}] = 0$. Thus the receiver performs simple threshold test as

$$\tau \underset{H_0}{\overset{H_1}{\gtrless}} \tau_{th}.$$

We emphasize that due to the low power of the tag, a node aware of its structure can verify its existence. However, a node without knowledge of the tag will not be able to check if a message contains it. This property is critical as it guarantees privacy in the scheme.

IV. SYSTEM DESCRIPTION

Consider an ad-hoc network of N distributed mobile nodes $M = \{M_1, \ldots M_N\}$. Each node is a low power device equipped with sensors of varying capabilities. The goal of the network is to collaborate to achieve an objective such as tracking an object or estimating the state of the system. This may be done via methods illustrated in Section II.

Assume a subset of N_t nodes to constitute the trusted core TC. A node may be part of the trusted core due to a higher degree of fault tolerance (trusted hardware), or robust measurements (better sensors) or simply due to a social hierarchy (platoon commanders). We assume each node to be aware of its capabilities (i.e. membership to the trusted core). However, regular nodes do not have any global view of the trusted core, nor any apriori knowledge of the identity of the trusted nodes.

This is a typical scenario as nodes may be continuously added to the network, several of which may be a part of the trusted core. Additionally, in a network, the configuration of the trusted core may differ for different tasks. For example, the trusted core for a sensing task may consist of nodes with a better sensor. For a coordinated movement task the trusted core may consist of nodes with GPS locators. In such dynamic scenarios, the distribution of a list of trusted nodes will induce significant overhead.

We assume the existence of a pre-shared key k by nodes of the network. Further, we consider that nodes strictly adhere to the collaboration protocol unless they have been compromised or in certain cases of arbitrary failures. As power of the nodes is limited, we do not consider the use of cryptographic methods for covertness, authentication or integrity protection.

A. Adversarial Model

Consider a set A of N_c adversaries. The goal of the adversaries is to defeat the network objective. In the case of state estimation, it may be to provide false state information. In a global consensus, this may be to force the network to converge to an incorrect value or not converge at all.

We limit our analysis to an external adversary, i.e: the adversary does not possess the group key k. The adversary compromises the network by disabling a subset of the nodes

via jamming or injecting spurious measurements by impersonating genuine nodes. In the latter case, the adversary may impersonate nodes it has disabled. We assume the adversary can compromise at most $N_c < N$ nodes.

As discussed in Section II, the presence of a trusted core makes the system robust to adversarial behavior. Thus maximum impact of an attack would be caused by reducing the size of the trusted core or impersonating the nodes in the trusted core. Thus we may assume that the primary objective of the adversary is to identify the nodes in the trusted core and disable and impersonate them.

In scenarios where the adversary is capable of capturing a node, we assume that the group key k and certain operations of the collaborative algorithm are not leaked. This may be ensured by delegating such operations and key storage to secure hardware modules present on the nodes. Architectures involving TrustZone or the Trusted Platform Module (TPM) can typically guarantee integrity of small operations and areas of the memory even in compromised nodes.

B. Message Tagging

We consider the nodes in the trusted core to utilize the tag described in Section III to identify themselves to the rest of the network. The nodes generate the tag as follows

$$t = \textit{HMAC}_k(ID, TS),$$

where $\textit{HMAC}_k(\cdot)$ denotes a message authentication code on a message using the private key k. ID denotes the identity of the transmitting node and TS denotes a timestamp embedded in the message (or sequence number). We allocate a small amount of power, $\rho_t^2 \in [0.001, 0.05]$, to the tag.

Similar to the procedure in Section III, the receiver extracts ID and TS from the message, and computes the test statistic to decide whether the message received is from a member of the trusted core.

The security properties of an HMAC ensure that an adversary cannot generate the expected tag without without knowledge of the key k. Furthermore, any tag t' generated with an assumed key k' will be uncorrelated to the original tag t. Intuitively, this guarantees that given a set of observed messages, the adversary cannot identify messages tagged by the trusted core, thus preserving the identity and location of trusted nodes.

Note: Using a pre-shared key, such privacy can be trivially achieved by encrypting all transmitted messages. An adversary unable to decode the packets cannot identify the trusted nodes. However, this requires encryption of *all* messages transmitted by *all* nodes, irrespective of whether they belong to the trusted core. This incurs significant overhead by the sender and receiver. In our scheme, we only tag messages originating from the trusted core, which would be small compared to the size of the network.

C. System Example

To demonstrate our scheme we utilize the problem formulation in [8]. The network is used for state estimation of a linear random process

$$\mathbf{x}(k+1) = A\mathbf{x}(k) + \mathbf{w}(k), \tag{4}$$

where $\mathbf{x} \in \mathbb{R}^m$ is the system state and $\mathbf{w}(k) \in \mathbb{R}^m$ is the state noise, $\mathbf{w}(k) \sim \mathcal{N}(0, Q)$. Each node has a linear sensor model

$$\mathbf{y}_i(k) = C_i\mathbf{x}(k) + \mathbf{v}_i(k),$$

where $\mathbf{y}_i(k) \in \mathbb{R}^{p_i}$ is the observation at node i and $\mathbf{v}_i(k) \in \mathbb{R}^{p_i}$ is the observation noise, $\mathbf{v}_i(k) \sim \mathcal{N}(0, V_i)$. We use the weighted DKF algorithm in [8] for estimation. The weights represent trust value for the reporting node inferred by the computing node. The trust for node j as inferred by node i is denoted as t_{ij}.

Let the graph $\mathcal{G} = (M, \mathcal{E})$ denote the topology of the network where M denotes the set of all nodes and $\mathcal{E}$ denotes the set of all edges ($e_{ij} \in \mathcal{E}$ if node i is within the communication range of node j). For our scenario, we assume the links to be symmetric. Define the neighborhood $\mathcal{N}_i$ of a node i as $\mathcal{N}_i = \{j \mid e_{ij} \in \mathcal{E}\}$.

During each iteration, a node i receives state information from its neighborhood. For each packet received, it extracts the residue, r_j, using (1) and generates the expected tag $\hat{\mathbf{t}}_j$ using the identity of the node j and the timestamp TS retrieved from the packet. It computes the test statistic $\tau_{ij} = \hat{\mathbf{t}}_j^H r_j$. It updates the global trust for other nodes as

$$\forall j \in M, t_{ij} = \frac{1}{|\mathcal{N}_i|}\left(\sum_{\{k\in\mathcal{N}_i|\tau_{ik}<\tau_{th}\}} t_{ik}t_{kj} + \sum_{\{k\in\mathcal{N}_i|\tau_{ik}\geq\tau_{th}\}} t_{max}t_{kj}\right), \tag{5}$$

where τ_{th} denotes the decision threshold to determine validity of the tag and t_{max} is the trust value for the trusted core. Typically, $t_{max} = 1$.

Occasionally, due to a false positive in the test for the tag, a non-trusted node will be assigned a higher weight. The probability of such an event may be computed as

$$P_{fa} = \mathbb{P}[\tau_{ij} > \tau_{th} | H_0].$$

Based on the application and tolerable errors, this can be adjusted by varying the τ_{th}.

V. SYSTEM SECURITY

We now consider the privacy guarantees provided by the tagging scheme. Intuitively, security of our scheme follows from stealth of the tag in [10]. Without apriori knowledge of the tag, an adversary cannot distinguish between a trusted node and a regular node. Thus, the best the adversary can do is to randomly select the nodes to attack.

A. Privacy Definition

We define the privacy of the system as a function of the adversary being able to distinguish between variations of the trusted core by observing the packets exchanged.

Definition 1: Consider an adversary $\mathcal{A}$ of N_c strength, i.e. the adversary compromises N_c nodes. Let the size of the trusted core be N_t. The adversary $\mathcal{A}$ upon observing a set of packets exchanged in the network, $\mathcal{P}(\cdot)$, dependent on the trusted core configuration S_t, chooses a set $S_c \subset$

M to attack. The loss in privacy $\mathcal{L}_{priv}$ due to adversarial observations is defined as

$$\mathcal{L}_{priv} = \max_{\{\mathcal{A},S_t,S_c\}} |\mathbb{P}[\mathcal{A}(\mathcal{P}(S_t)) = S_c] - \mathbb{P}[\mathcal{A}(\mathcal{P}(S_t')) = S_c]|, \quad (6)$$

where $S_t, S_t' \subset M$ are two sets of nodes representing the trusted core such that $S_t \neq S_t'$ and $|S_t| = |S_t'| = N_t$. Further, $|S_c| = N_c$.

The loss metric $L_{priv} \in [0, 1]$ quantifies the strength of the system against an adversary. Let $\mathcal{A}^*$ be the optimal adversary, i.e. one that is to maximally identify the trusted core

$$\mathcal{A}^* = \arg\max_{\mathcal{A}} |\mathcal{A}(\mathcal{P}(S_t)) \cap S_t|$$

Theorem 1: Consider a system with uniform distribution of trusted nodes, such that the loss of privacy $\mathcal{L}_{priv} \leq \epsilon$. Then for the adversary $\mathcal{A}^*$ with bounded strength N_c,

$$\frac{\mathbb{E}\left[|\mathcal{A}^*(\mathcal{P}(S_t)) \cap S_t|\right]}{\mathbb{E}[|\mathcal{A}_U() \cap S_t|]} \leq (1 + \epsilon \cdot \mathcal{O}(poly(N))), \quad (7)$$

where $\mathcal{A}_U()$ denotes an adversary which picks a uniform set of size N_c to be attacked.

Proof: In Appendix ∎

This illustrates that if the loss of privacy is small, no adversary can do better than random sampling.

B. Adversary Strategy

We consider an adversary capable of observing all network communication. This enables the adversary to obtain all the residues, following the procedure of a regular receiver. Since the adversary cannot perform matched filtering, due to the absence of the key k, its strategy is limited to detecting tags by by performing statistical inference tests on the residue for anomalies. Consider the residues obtained by the adversary.

$$\mathbf{r}_i = \mathbf{t}_i + \frac{\hat{h}_i^*}{\rho_t |\hat{h}_i|^2} \mathbf{w}_i, \quad i = 1, 2, \ldots, .$$

Depending on the test statistic, the index may denote node or time. An adversary may perform correlation on residues to obtain more robust test statistics, e.g:

$$r_{12} = \mathbf{r}_1^H \mathbf{r}_2 = \mathbf{t}_1^H \mathbf{t}_2 + \frac{\hat{h}_2^*}{\rho_t |\hat{h}_2|^2} \mathbf{t}_1^H \mathbf{w}_2 + \frac{\hat{h}_1^*}{\rho_t |\hat{h}_1|^2} \mathbf{w_1}^H \mathbf{t}_2 + \frac{\hat{h}_1^* \hat{h}_2}{\rho_t^2 |\hat{h}_1|^2 |\hat{h}_2|^2} \mathbf{w}_1^H \mathbf{w}_2. \quad (8)$$

We exemplify a few common statistics the adversary may use to perform the goodness-of-fit tests (e.g. Kolmogorov-Smirnov test) for distinguishing nodes

E1. Comparing series of residues with white Gaussian noise (channel noise) ($\mathbf{r}_1$ vs. noise)
E2. Comparing series of residues from two different nodes to isolate individual trusted nodes ($\mathbf{r}_1$ vs. $\mathbf{r}_2$).
E3. Correlating residues from pairs of nodes for comparison ($\mathbf{r}_{12}$ vs. $\mathbf{r}_{34}$).
E4. Generating a random tag, correlating the residue against the tag and comparing with white Gaussian noise (channel noise) ($\mathbf{r}_{12}$, where $\mathbf{w}_2 = 0$, vs. noise)

We discuss examples of the tests and the results via simulation in Section VI. Here we quantify the privacy loss (6) for a simple example.

Single Adversary Example: Consider the scenario of a single adversary, i.e: $N_c = 1$, and a single trusted node $N_t = 1$. The adversary performs experiment (E1), i.e: Lilliefors test [11] for every residue using channel statistics, to obtain a decision; false if the residue follows Gaussian distribution, true otherwise. The adversary considers all the nodes which yield a true decision and randomly selects a node. Denote the probability of detection for the test as (α) and the probability of false alarm as (β). Considering that a tagged signal, has normal distribution with slightly different variance from the channel noise, we argue that α would be small. To compute (6), we obtain 3 scenarios, i.e:

A1. Trusted node not selected ($S_c \cap S_t = S_c \cap S_t' = \varnothing$)
A2. $S_c \cap S_t = S_c \cap S_t'$
A3. Trusted node selected for one case only ($S_c \cap S_t = S_t, S_c \cap S_t' = \varnothing$)

Clearly, the difference in the probabilities for (A1) is 0, and (A2) is an impossible event as $S_t = S_t'$ is a contradiction. Thus the L_{priv} is obtained by considering (A3). Thus

$$\mathbb{P}[S_c = S_t] = \frac{1}{N}\left[(1-\alpha)(1-\beta)^{N-1} + \frac{\alpha}{\beta}(1-(1-\beta)^N)\right].$$

The adversary is incorrect when the trusted node fails to be detected and one of the incorrect nodes is flagged, i.e:

$$\mathbb{P}[S_c \cap S_t' = \varnothing] = \frac{1}{N}(1-\alpha)(1-\beta)^{N-1} + \frac{1}{N-1}(1-(1-\beta)^{N-1}) - \frac{\alpha}{\beta}\frac{1}{N(N-1)}\left(1-(1-\beta)^N - N\beta(1-\beta)^{N-1}\right).$$

Thus we obtain

$$\mathcal{L}_{priv} = \left|\frac{1}{N-1}(1-(1-\beta)^{N-1})\left(1-\frac{\alpha}{\beta}\right)\right|.$$

It can be observed that for a small value of α, as ensured by our design, the loss of privacy is low. Thus the adversary would not perform much better than random selection. As a special case, if we can ensure that the α is equivalent to false positive of the statistical test used by the adversary, i.e. $\alpha \approx \beta$, we obtain no loss of privacy.

Though the argument above is for a simple adversary, it highlights the gain obtained by our scheme. We discuss via simulations the gain for more complex adversarial scenarios.

VI. SIMULATION RESULTS

We verify our assertions via MATLAB simulations. First we highlight the influence of adversarial behavior in the absence of our scheme. We then present the performance of our scheme to mitigate adversarial behavior.

A. Consequences of compromised trusted core

We illustrate the influence of an adversary that is able to violate the security assumptions of the trusted core. Consider a sensor network to track an object moving in the 2-D plane. The network size is $N = 100$. The communication neighborhood is determined using the unit disk model. The target trajectory follows (4), with A = [1 -0.02; 0.02 1]. Each sensor can only sense one dimension of the target's position, i.e. half sense the x-direction (C = [1 0; 0 0]) and rest, the

y-direction (C = [0 0; 0 1]). The size of the trusted core is $N_t = 20$.

We model an adversary capable of tampering with the measurements of the compromised sensors. At iteration k, the adversary compromises the observation of the ith sensor by adding an offset $\mathbf{a}_i(k)$ to the measurement, i.e. $\mathbf{y}_i^a(k) = \mathbf{y}_i(k) + \mathbf{a}_i(k)$. The number of compromised nodes is set to $N_c = 45$. We consider $\mathbf{a}_i(k) = [10, 10], \forall\ (i, k)$,

Consider the scenario where an adversary successfully identifies a subset of the trusted core. We consider the simple case where the identified trusted core nodes are jammed. This effectively reduces the size of the trusted core. The compromise of a node not in the trusted core involves alteration of the measurements, as described above. We simulate the scenario with a varying percentage of the trusted core compromised by the adversary.

The performance degradation in tracking error due to decrease in size of the trusted core can be clearly observed in Fig. 1. However, even with a few trusted nodes remaining, the error significantly improves over traditional methods. This is due to the malicious nodes being detected and assigned lower weights based on their trust values.

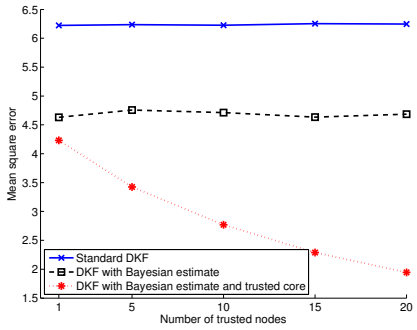

Fig. 1. MSE of good nodes with varying number of compromised trusted nodes (jamming).

Next we simulate the adversary that identifies the trusted nodes (as before) and tampers with their measurements by adding an offset. Figure 2 shows the severe performance degradation of the network when the trusted nodes provide malicious data. Furthermore, due to malicious measurements being given higher weights, the performance for several cases is worst than the traditional methods.

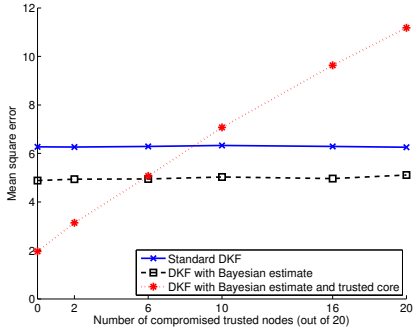

Fig. 2. MSE of good nodes with varying number of compromised trusted nodes (measurement offset).

This clearly signifies the importance of concealing the identity of the trusted core. Compromise of privacy of even a fraction of the trusted core can lead to significant decrease in performance.

B. Performance and security of embedded tags

We now present the performance of the tagging scheme for preserving privacy and avoiding the situations above.

1) Robustness: It is critical to detect the presence of the tags accurately. A weak scheme can be a cause for denial-of-service even in the absence of an external adversary. Using parameters of [10], in Figure 3, we plot the authentication probabilities in various channel conditions.

We allocate 1.5% of the signal power to the tag, i.e. $\rho_t^2 = 0.015$. We use $\mathcal{P}_{fa} = 0.01$ to determine the threshold τ_{th} We can see that even in poor channel conditions, the probability of correctly detecting the tag is high.

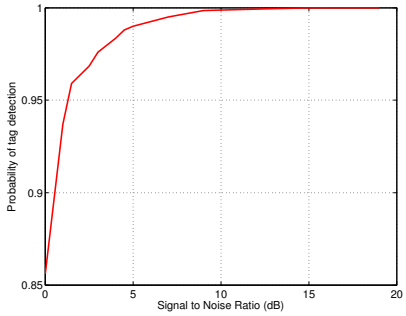

Fig. 3. Tag authentication probabilities under various channel conditions.

2) Stealth: The most important property of the tags is the inability of the adversary to detect them without knowledge of the secret key k. As discussed in Section V-B, th adversary can perform statistical inference tests on individual residues or correlated residues.

First, we consider the case where the adversary performs Lilliefors test on the residue to observe deviation from Gaussian distribution. We simulate the system using a tag to noise ratio of -10 dB. Lilliefors test with a 1% confidence (prob. false positive) returns negative with average p-value 0.37. Therefore the adversary does not have enough statistical confidence to discern between residue with tag or just noise.

Next, we consider correlation (time, space) based scenarios from Sec. V-B. We simulate identification of pairs of nodes in the trusted core by correlating residues from every pair and performing Kolmogorov - Smirnov test.

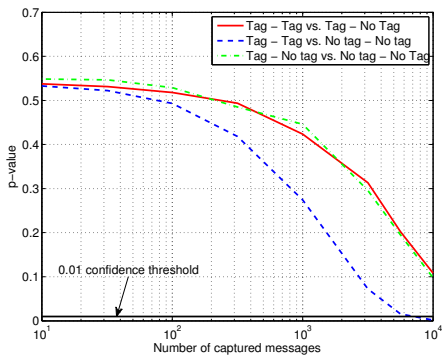

Fig. 4. Adversary's false positive confidence for different pairs of correlation data.

From Fig. 4 we observe that the adversary achieves maximum distinguishability when comparing correlation data from two residues with tag against correlation data from two residues without tag. However, the number of observations required to obtain a statistically significant deviation is extremely large and impractical.

As an example, we perform these tests in the context of an adversary of size $N_c = 10$ with $N_t = 1$. We observe the number of times (T) the adversary was able to compromise the trusted node in 10000 iterations.

1) Perform Lilliefors test on the residues. Select N_c nodes whose residues have the lowest p-values. $T = 1018$
2) Correlate the residues with a generated tag. Select N_c nodes whose correlations are highest. $T = 1099$
3) Select N_c nodes randomly. $T = 1058$

Thus it can be deduced that in the proposed tagging system, the evidence generated by the adversary is insufficient to gain any advantage over purely random selection of nodes to attack.

VII. CONCLUSIONS

In this paper we discussed the role and significance of a trusted core in distributed control systems. We proposed privacy as a means to fulfill the the security assumptions for the TC. Further we defined a privacy metric and characterized the adversarial advantage relative to the metric. We utilized a low-cost physical layer technique to provide identity privacy and security in wireless networked systems. For certain scenarios, we characterized the privacy loss of our scheme both analytically and via simulations.

APPENDIX

An equivalent way to represent a set $S \subset M$ is a length-N binary vector where the i^{th} index is 1 if node $i \in S$, and 0 otherwise. Let $\mathcal{T}$ be the set of all vector representations of a trusted core. $\mathcal{T}$ is the set of length-N binary vectors with Hamming weight N_t. Similarly, let $\mathcal{C}$ be the set of all vector representations of an adversary's attack selection. $\mathcal{C}$ is the set of length-N binary vectors with Hamming weight N_c. We use $\mathbf{T}$ and $\mathbf{C}$ to denote the random vectors representing the trusted core and adversary's selection. $\mathbf{t}$ and $\mathbf{c}$ denote realizations of those random vectors. For a system with loss of privacy $\mathcal{L}_{\text{priv}} = \epsilon$,

$$\left|\mathbb{P}[\mathcal{A}(\mathcal{P}(\mathbf{t})) = \mathbf{c}] - \mathbb{P}[\mathcal{A}(\mathcal{P}(\mathbf{t}')) = \mathbf{c}]\right| \leq \epsilon, \forall \mathcal{A} \tag{9}$$

Theorem 1 can be equivalently stated as

$$\max_{\mathcal{A}} \frac{\mathbb{E}[\mathcal{A}(\mathcal{P}(\mathbf{T}))^T \mathbf{T}]}{\mathbb{E}[\mathcal{A}_U()^T \mathbf{T}]} \leq (1 + \epsilon \cdot \mathcal{O}(\text{poly}(N))) \tag{10}$$

To prove Theorem 1 we will need the following lemmas

Lemma 1:

$$\mathbb{E}[\mathcal{A}_U()^T \mathbf{T}] = \frac{N_c N_t}{N}. \tag{11}$$

Proof: Since the trusted nodes are distributed uniformly and $\mathcal{A}_U(\cdot)$ is uniform and independent of $\mathbf{T}$,

$$\begin{aligned}\mathbb{E}[\mathcal{A}_U()^T \mathbf{T}] &= \mathbb{E}[\mathcal{A}_U()]^T \mathbb{E}[\mathbf{T}] \\ &= \frac{N_c}{N}\frac{N_t}{N}\mathbf{1}^T\mathbf{1} = \frac{N_c N_t}{N}.\end{aligned}$$

■

Lemma 2:

$$\sum_{\mathbf{c}\in\mathcal{C}} \mathbf{c} = \binom{N}{N_c}\frac{N_c}{N}\mathbf{1}, \quad \sum_{\mathbf{t}\in\mathcal{T}} \mathbf{t} = \binom{N}{N_t}\frac{N_t}{N}\mathbf{1}. \tag{12}$$

Proof: It is sufficient to prove the first equality as the second equality follows similarly. Since $\mathcal{C}$ is the set of all length-N binary vectors with Hamming weight N_c, $|\mathcal{C}| = \binom{N}{N_c}$. Thus, the sum Hamming weight of all vectors of $\mathcal{C}$ is $N_c|\mathcal{C}|$. This is also the sum of all elements of $\sum_{\mathbf{c}\in\mathcal{C}} \mathbf{c}$. Since the elements of this summation vector are the same, each element is $\frac{N_c|\mathcal{C}|}{N} = \binom{N}{N_c}\frac{N_c}{N}$. ■

Proof: [Proof of Theorem 1]

$$\begin{aligned}&\mathbb{E}[\mathcal{A}(\mathcal{P}(\mathbf{T}))^T \mathbf{T}] \\ &= \sum_{\mathbf{t}\in\mathcal{T}} \mathbb{P}[\mathbf{T} = \mathbf{t}]\mathcal{A}(\mathcal{P}(\mathbf{t}))^T \mathbf{t} = \frac{1}{|\mathcal{T}|}\sum_{\mathbf{t}\in\mathcal{T}} \mathcal{A}(\mathcal{P}(\mathbf{t}))^T \mathbf{t} \\ &= \frac{1}{|\mathcal{T}|}\sum_{\mathbf{t}\in\mathcal{T}}\sum_{\mathbf{c}\in\mathcal{C}} \mathbb{P}[\mathcal{A}(\mathcal{P}(\mathbf{t})) = \mathbf{c}]\mathbf{c}^T\mathbf{t} \\ &\leq \frac{1}{|\mathcal{T}|}\sum_{\mathbf{t}\in\mathcal{T}}\sum_{\mathbf{c}\in\mathcal{C}} \left(\mathbb{P}[\mathcal{A}(\mathcal{P}(\mathbf{t}_0)) = \mathbf{c}] + \epsilon\right)\mathbf{c}^T\mathbf{t} \\ &= \frac{1}{|\mathcal{T}|}\sum_{\mathbf{t}\in\mathcal{T}}\sum_{\mathbf{c}\in\mathcal{C}} \mathbb{P}[\mathcal{A}(\mathcal{P}(\mathbf{t}_0)) = \mathbf{c}]\mathbf{c}^T\mathbf{t} + \epsilon\frac{1}{|\mathcal{T}|}\sum_{\mathbf{t}\in\mathcal{T}}\sum_{\mathbf{c}\in\mathcal{C}} \mathbf{c}^T\mathbf{t} \\ &= \frac{1}{|\mathcal{T}|}\sum_{\mathbf{c}\in\mathcal{C}} \mathbb{P}[\mathcal{A}(\mathcal{P}(\mathbf{t}_0)) = \mathbf{c}]\mathbf{c}^T\sum_{\mathbf{t}\in\mathcal{T}}\mathbf{t} + \epsilon\frac{1}{\binom{N}{N_t}}\binom{N}{N_c}\binom{N}{N_t}\frac{N_c N_t}{N} \\ &= \frac{1}{|\mathcal{T}|}\sum_{\mathbf{c}\in\mathcal{C}} \mathbb{P}[\mathcal{A}(\mathcal{P}(\mathbf{t}_0)) = \mathbf{c}]\binom{N}{N_t}\frac{N_t}{N}\mathbf{c}^T\mathbf{1} + \epsilon\binom{N}{N_c}\frac{N_c N_t}{N} \\ &= \frac{1}{|\mathcal{T}|}\sum_{\mathbf{c}\in\mathcal{C}} \mathbb{P}[\mathcal{A}(\mathcal{P}(\mathbf{t}_0)) = \mathbf{c}]\binom{N}{N_t}\frac{N_t}{N}N_c + \epsilon\binom{N}{N_c}\frac{N_c N_t}{N} \\ &= \frac{1}{\binom{N}{N_t}}\binom{N}{N_t}\frac{N_t}{N}N_c + \epsilon\binom{N}{N_c}\frac{N_c N_t}{N} = \frac{N_c N_t}{N}\left(1 + \epsilon\binom{N}{N_c}\right)\end{aligned}$$

Where $\mathbf{t}_0$ is a particular trusted core configuration. The inequality makes use of the privacy guarantee of the system. Since for bounded N_c, $\binom{N}{N_c} < N^{N_c} = \mathcal{O}(\text{poly}(N))$, we have the proof. ■

REFERENCES

[1] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, Jun 2003.

[2] M. Mahmoud and H. Khalid, "Distributed Kalman filtering: a bibliographic review," *IET Control Theory Applications*, vol. 7, no. 4, pp. 483–501, March 2013.

[3] H. J. LeBlanc and X. D. Koutsoukos, "Low complexity resilient consensus in networked multi-agent systems with adversaries," in *Proc. Intl. Conf. on Hybrid Sys. Comp. and Control*, 2012, pp. 5–14.

[4] A. Melin, E. Ferragut, J. Laska, D. Fugate, and R. Kisner, "A mathematical framework for the analysis of cyber-resilient control systems," in *Proc. Intl. Symposium on Resilient Control Sys*, Aug 2013.

[5] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. D. Keromytis, "The Role of Trust Management in Distributed Systems Security," in *Secure Internet Programming*, 1999, pp. 185–210.

[6] S. Sundaram and C. Hadjicostis, "Distributed Function Calculation via Linear Iterative Strategies in the Presence of Malicious Agents," *IEEE Trans. on Automatic Control*, vol. 56, no. 7, pp. 1495–1508, Jul 2011.

[7] A. Harrington and C. Jensen, "Cryptographic Access Control in a Distributed File System," in *Proc. of ACM Symposium on Access Control Models and Technologies*, 2003, pp. 158–165.

[8] S. Zheng, T. Jiang, and J. Baras, "Robust state estimation under false data injection in distributed sensor networks," in *Proc. IEEE Global Telecommunications Conf.*, Dec 2010, pp. 1–5.

[9] K. Somasundaram and J. Baras, "Performance improvements in distributed estimation and fusion induced by a trusted core," in *Proc. International Conf. on Information Fusion*, July 2009, pp. 1942–1949.

[10] P. Yu, J. Baras, and B. Sadler, "Physical-layer authentication," *IEEE Trans. on Information Forensics and Security*, vol. 3, no. 1, pp. 38–51, Mar. 2008.

[11] H. Lilliefors, "On the KolmogorovSmirnov test for normality with mean and variance unknown," *Journal of the American Statistical Association*, vol. 62, pp. 399–402, June 1967.