IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Sphinx: a Colluder-Resistant Trust Mechanism for Collaborative Intrusion Detection

CARLOS GARCIA CORDERO[1], GIULIA TRAVERSO[2], MEHRDAD NOJOUMIAN[3], SHEIKH MAHBUB HABIB[4], MAX MÜHLHÄUSER[5], JOHANNES BUCHMANN[6] AND EMMANOUIL VASILOMANOLAKIS[7]

[1]Technische Universität Darmstadt, Darmstadt, Germany (e-mail: garcia@tk.tu-darmstadt.de)
[2]Technische Universität Darmstadt, Darmstadt, Germany (e-mail: gtraverso@cdc.informatik.tu-darmstadt.de)
[3]Florida Atlantic University (USA) (e-mail: mnojoumian@fau.edu)
[4]Continental AG, Frankfurt, Germany (e-mail: sheikh.mahbub.habib@continental-corporation.com)
[5]Technische Universität Darmstadt, Darmstadt, Germany (e-mail: max@tk.tu-darmstadt.de)
[6]Technische Universität Darmstadt, Darmstadt, Germany (e-mail: buchmann@cdc.informatik.tu-darmstadt.de)
[7]Technische Universität Darmstadt, Darmstadt, Germany (e-mail: vasilomano@tk.tu-darmstadt.de)

Corresponding author: Carlos Garcia Cordero.

ABSTRACT The destructive effects of cyber-attacks demand more proactive security approaches. One such promising approach is the idea of Collaborative Intrusion Detection Systems (CIDSs). These systems combine the knowledge of multiple sensors (e.g., intrusion detection systems, honeypots or firewalls) to create a holistic picture of a monitored network. Sensors monitor parts of a network and exchange alert data to learn from each other, improve their detection capabilities and ultimately identify sophisticated attacks. Nevertheless, if one or a group of sensors is unreliable (due to incompetence or malice), the system might miss important information needed to detect attacks. In this article, we propose *Sphinx*, an evidence-based trust mechanism capable of detecting unreliable sensors within a CIDS. *Sphinx* detects, both, single sensors or coalitions of dishonest sensors that lie about the reliability of others to boost or worsen their trust score. Our evaluation shows that, given an honest majority of sensors, dishonesty is punished in a timely manner. Moreover, if several coalitions exist, even when more than 50% of all sensors are dishonest, dishonesty is punished.

INDEX TERMS clustering, collaborative intrusion detection, machine learning, mixture models, sensor reliability, trust management

I. INTRODUCTION

Recent cyber-attacks such as the Distributed Denial of Service (DDoS) attacks of the Internet of Things (IoT)-enabled Mirai botnet [1] highlight the need for novel cyber-defense mechanisms. Over the last years a lot of research has been conducted towards the notion of collaborative defense [2], [3]. The core idea behind this is, as the name implies, to utilize the knowledge of multiple monitoring entities (so-called sensors) to create a holistic picture of a monitored network.

Sensors (e.g., honeypots, intrusion detection systems, firewalls, etc.) identify attacks on a communication's network so as to mitigate disruptions. However, isolated sensors cannot effectively detect coordinated attacks, especially in large-scale networks, unless they collaborate [3]. As a result, Collaborative Intrusion Detection Systems (CIDSs) have been proposed as a line of defense suitable to modern network communications. A CIDS can identify sophisticated and coordinated attacks as follows. Upon discovering suspicious behavior, sensors can raise and share alarms with each other. Afterwards, by aggregating and correlating alert data, the CIDS is able to identify attacks that would otherwise be invisible to isolated sensors.

A major challenge in the field of CIDSs is dealing with the trustworthiness of sensors. That is, the overall accuracy of a CIDS can severely degrade when sensors are *compromised* and report false data, or when sensors are *unreliable* and either submit false data or no data at all. To cope with this, a computational trust mechanism can be utilized inside the CIDS. In this article, we propose a trust mechanism that

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access

identifies honest and dishonest sensors taking into account their reliability. *Honest sensors* are sensors with the common goal of sharing information as accurately as possible within the CIDS. Conversely, *dishonest sensors* may only share information that would advance their personal goals. Dishonest sensors may also tamper the information they share or collude with other dishonest sensors in the tampering process. A group of coordinated dishonest sensors that collude with the common goal of improving the trust of the sensors in their own group (to the eyes of everyone else), while worsening the trust of those outside the group, is known as a *coalition*.

We propose *Sphinx*[1], an evidence-based trust mechanism that uses the sensing reliability of participating sensors to detect dishonesty[2]. Beyond detecting one (single) dishonest sensor, we also consider the scenario of detecting coalitions. With *Sphinx*, we make the following contributions on top of the state of the art:

- We develop a trust mechanism for CIDSs that takes into account the reliability of a sensor as part of its trustworthiness (in addition to an evidence-based trust score).
- We detect both isolated malicious and incompetent sensors as well as groups of malicious sensors that form coalitions. In addition, we are agnostic with regard to the CIDS architecture. *Sphinx* works analogously in fully distributed as well as centralized CIDSs[3].
- Contrary to the state of the art, we relax the assumption that malicious sensors always behave consistently. We enable *Sphinx* to detect dishonest sensors that choose, with some probability, to act as honest sensors.

The reminder of the article is organized as follows. Related work is discussed in Section II. This is followed by Section III, where our proposal, *Sphinx*, is detailed. A detailed evaluation is presented in Section IV and conclusions can be found in Section V.

II. RELATED WORK

Trust mechanisms are referred to as evidence-based trust mechanisms when they rely on evidence derived from past interactions. More precisely, evidence can be derived from direct interactions between a trustor and a trustee. Direct interactions, however, may be rare in certain cases, e.g., newcomers in service marketplaces. Thus, evidence-based mechanisms also consider evidence derived from indirect interactions. That is, an entity provides another with evidence about its past interactions with a third entity. This is usually referred to as exchange of recommendations. In the case that both direct and indirect interactions are not available, one may rely on evidence derived from virtual cues, e.g., certifications or stereotypes. In this article, we are interested in computational trust models that consider the past evidence (via direct or indirect interactions) of a trustees' behavior to estimate the future trustworthiness of that trustee. In this Section, we examine diverse evidence-based trust mechanisms based on statistical and machine learning techniques. Furthermore, we highlight key trust mechanisms applied within CIDSs.

A. BAYESIAN TRUST MODELS

Bayesian trust models [5]–[9] leverage Bayesian probabilities [10] to estimate the future behavior (i.e., the trust score) of a trustee. The Beta probability density function is used in this models to estimate the future behavior based on evidence collected from past interactions. For instance, the reputation system proposed in [5] calculates trust scores following the Beta distribution. The system, however, is not able to filter out dishonest evidence, making the system ineffective when evidence is not honest. A more robust reputation system is introduced in [6]. This reputation system uses the honesty of the participants to build trust relationships. The main idea is to learn from the observation of others before learning from direct interaction. In other words, reputation ratings are incorporated into the view of others.

An extension of the Bayesian probabilistic model is the event-based trust mechanism proposed in [7], which handles so called event-structure frameworks [11]. This work provides a formal framework based on information divergence to measure the quality of probabilistic trust mechanisms. Furthermore, the trust-aware model introduced in [8], addressing service-oriented environments, formalizes a Bayesian service selection model focusing on monitoring and exploring service composition. The work shows how one can reward or punish services dynamically even with incomplete knowledge of the composition.

B. MACHINE LEARNING FOR TRUST MODELS

Nowadays, an increasing amount of evidence (or data) is generated by large-scale web applications, e.g., social media, e-commerce or recommender systems. Machine learning techniques are used by researchers to model complex scenarios by answering two fundamental questions in trust research. The first question being: *in the absence of past behavior, how can trustworthiness be estimated*? And second, *how can the dynamic behavior of a target be estimated from different interactions*?

To address the first question, stereotyping models, e.g., [12], use the trustor's past experience with other similar entities. These models harness trust-relevant features using machine learning techniques [13] (e.g., Linear Discriminant Analysis (LDA), Decision Tree (DT) or M5 model trees) to

[1]In ancient Greek mythology Sphinx was a creature that guarded the entrance to the city of Thebes asking travelers a riddle to allow them passage; hence allowing only trustworthy and/or knowledgeable entities to enter. Our system is inspired by Sphinx as it requires sensors to reliably and correctly answer requests.

[2]Note that the theoretical basis behind *Sphinx* is not only applicable within the context of CIDSs. In fact, we introduced the basis of the trust mechanism, that *Sphinx* is using, and demonstrated how it can be applied to the field of social secret sharing in the short paper [4]. The article at hand provides an extended version of the aforesaid work, along with an adaption to the CIDS context, less unrealistic assumptions, the support for smart attackers and last but not least a full fledged evaluation.

[3]For an introduction to the different CIDS network architectures see [3].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access

extract connections between potential interactions and past interactions.

To address the second question, Tang et al. [14] address the issue of dynamic behavior. The authors analyze the evolution of trust by investigating the online dynamics of users in review sites like Epinions[4]. It turns out that trust is strongly correlated to the similarity of the preference of the users. To capture their preference evolution, or dynamic trust, the authors use machine learning approaches such as latent factor models [15]. Moreover, in evidence-based trust mechanisms, evidence is often provided by different sources. Honesty of the source of information is key for reliable trust estimation and, thus, it is essential to determine whether the information source is unbiased or not. Existing evidence-based trust models use unsupervised approaches, like statistical deviation [16], to identify responses that are very different from others. The assumption here is that biased responses are a small subset of all responses. Furthermore, in large-scale open systems like social networks, the behavior of an entity with respect to others may vary so as to maximize profits. Approaches based on Hidden Markov Models [17] have also proven effective in detecting and correcting dynamic behavior.

Our article addresses the second question (of *how can the dynamic behavior of a target be estimated from different interactions*) by using machine learning techniques to design our system. We use unsupervised clustering algorithms to identify evidence created by dishonest participants. In contrast to related work, with the techniques of fitting mixture of Gaussians to clusters, we identify unreliable evidence submitted by colluding participants. This confers *Sphinx* the capability to downgrade the trustworthiness of groups of participants that have chosen to collude.

C. TRUST MANAGEMENT WITHIN CIDSs

In the early days of CIDS research, it was mostly assumed that every collaborating sensor was honest; all having the common goal of detecting coordinated attacks [3]. More implicitly, but equally important, it was also assumed that the sensing capabilities of collaborating sensors were reliable. That is, collaborating sensors could be trusted to reliably monitor (i.e., sense) the network point to which they were assigned. In recent years, the assumption that there were no insider threats has been taken more seriously, e.g., [18], [19]. The assumption that the sensing capabilities of sensors might not be reliable, however, has not been extensively addressed in related work. With *Sphinx*, we aim at proposing an approach to address this issue when sensing reliability cannot be directly obtained. Instead, we assume that this information can only be indirectly obtained by asking other sensors; which enables sensors to collude and provide dishonest reports.

Fung et al. have worked on many aspects of identifying insider threats within CIDSs. In [20], they propose a general framework to bestow CIDS sensors the ability to assess the trustworthiness of others from past interactions. Afterwards, in [21], they propose a trust mechanism based on the Dirichlet distribution that enables sensors to update their trustworthiness from outcomes of mutual interaction. This last work considers the existence of both malicious and incompetent sensors. Finally, in [18], they add the concept of acquaintances that, along with the previous Dirichlet-based model, can break or establish dynamic relationships to improve the performance of a CIDS. It is worth noting that in many of the contributions of Fung et al., through the usage of a message passing system, collusion cannot occur. In our work, instead, we cannot discard the possibility of collusion due to the fact that their message passing system cannot function in our scenario. This is because we assume that it is not possible to directly query all CIDS sensors. Instead, we need to rely on the opinions of others. This is a common case when multiple CIDS sensors belong to different groups, when the CIDS employs a fully distributed P2P architecture, or both.

Collusion resistant trust models have been proposed in different fields. Dwarakanath et al. proposed a collusion resistant mechanism that detects dishonest IoT devices communicating false trust scores [22]. Collusion is detected by comparing trust vectors reported by multiple devices using a cosine similarity metric. Although this collusion detection mechanism does not target CIDSs, it could be adapted to such systems.

III. *SPHINX*: A COLLUDER-RESISTANT TRUST MECHANISM

In this section, our colluder-resistant trust mechanism, namely *Sphinx*, is presented. First, the general framework and main assumptions are shown (Section III-A). Then, Section III-B, Section III-C, and Section III-D describe how *Sphinx* operates. In Table 1 we provide a reference summary of the most important notations used throughout this paper.

A. FRAMEWORK AND ASSUMPTIONS

Let us assume a CIDS having a set $\mathcal{S} = \{S_1, \ldots, S_n\}$ of n collaborating sensors. Trust scores $\tau_1^{(t)}, \ldots, \tau_n^{(t)} \in [0, 1]$ are assigned, respectively, to sensors $S_1, \ldots, S_n$ at time t. These trust scores convey information about how reliable the sensing capabilities of the sensors are. Sensors periodically interact with each other exchanging local knowledge[5]. After each interaction at time t, sensors evaluate the reliability of each other and update trust scores $\tau_1^{(t)}, \ldots, \tau_n^{(t)}$. To simplify notation, trust scores $\tau_1^{(t)}, \ldots, \tau_n^{(t)}$ at time t are simply denoted as $\tau_1, \ldots, \tau_n$.

Trust scores $\tau_1, \ldots, \tau_n$ indirectly measure a sensor's sensing reliability. In an environment where sensing reliability is rewarded, sensors might be interested in maximizing their trust scores. To maximize its trust score, a sensor can go

[4]http://epinions.com/

[5]This work assumes that if sensor S_i shares its local knowledge, all sensors $S_j \in \mathcal{S}$ with $j \neq i$ receive the knowledge unaltered.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access

S_i	sensor indexed by i
n	total number of sensors
τ_i	trust score of S_i at time t
τ_i' / τ_i''	evidence- / reliability- based trust score of S_i at time t
$\tau_i^{(t-1)}$	reputation of S_i at time $t-1$
$P_j^{(i)}$	Cartesian point related to how S_j rates S_i
$x_{P_j^{(i)}}$ / $y_{P_j^{(i)}}$	$x-$ / y-coordinate of $P_j^{(i)}$
$\sigma_j^{(i)}$	evidence submitted by S_j with respect to S_i
K	total number of cluster centers
$\mathcal{C}_1, \ldots, \mathcal{C}_K$	clusters or classes of credibility
$M_1, \ldots, M_K$	center points of clusters $\mathcal{C}_1, \ldots, \mathcal{C}_K$
$y_{M_1}, \ldots, y_{M_K}$	y-coordinate of $M_1, \ldots, M_K$
$\omega_j^{(i)}$	weight of data point $P_j^{(i)}$
$\pi_1, \ldots, \pi_K$	mixing coefficients of $M_1, \ldots, M_K$
$o_i^{(j)}$	trustworthiness gain or loss by S_i with respect to S_j
α_1	reward and penalty values used to calculate $o_i^{(j)}$
α_2	number of possible values $o_i^{(j)}$ can have
$F(\tau_i)$	weight balance between high and low trust scores τ_i
η	mixing coefficient of τ_i' and τ_i'' to create τ_i

TABLE 1: Summary of the notations used in this article.

through the hardships of ensuring high sensing availability and accuracy so that others give the sensor good ratings. However, sensors might also consider lying about the reliability of others so as to make the others look worse. Furthermore, those dishonest sensors might secretly choose to form coalitions to boost the trust scores of their members and decrease the trust scores of others outside the coalition. We consider that each sensor $S_1, \ldots, S_n$ may be either honest or dishonest, where dishonest sensors might belong or not to a coalition. We make the following three assumptions with respect to the behavior of honest and dishonest sensors.

- *Assumption 1*: Honest sensors report evidence as accurate as they can but make small mistakes that follow a Gaussian distribution[6].
- *Assumption 2*: Dishonest sensors submit tampered evidence following a Beta distribution[6] and can choose to submit accurate evidence to confuse the system following a Uniform distribution[6].
- *Assumption 3*: When dishonest sensors collude, they only belong to one coalition.

Sphinx aims at mitigating the effect of a coalition under the last three assumptions. To achieve this, *Sphinx* calculates the trust score τ_i using two measurements termed the evidence-based and reliability-based trust scores. The evidence-based trust score τ_i' for sensor S_i results from the evidence submitted by sensors S_j, for $j = 1, \ldots, n$ and $j \neq i$ (see Section III-B). The calculation of the evidence-based trust score is somewhat similar (see Section II) to what is done by Bayesian models, except that the evidence processed for the computation of the evidence-based trust score has different relevance depending on the reputation of the source. The reliability-based trust score τ_i'' depends on how reliable the evidence submitted by sensor S_i is with respect to sensor S_j, with $j \neq i$ (see Section III-C). In the calculation of this reliability-based trust score, unreliable evidence is detected and the submitter is discouraged to do so by decreasing its trustworthiness. In Section III-D, we describe how to merge these two values to obtain the final trust score τ_i.

[6]For a discussion of why this distribution is used see Section IV-A.

B. EVIDENCE-BASED TRUST SCORE

This section describes how the evidence-based trust score τ_i' for sensor S_i is computed, taking into account the evidence submitted by all the other sensors. The computation of the evidence-based trust score τ_i' is performed in an Euclidean space of dimension $D = 2$. For readability, we divide this computation into the following steps.

a: Collecting the Evidence

Each sensor S_j submits point $P_j^{(i)} = (x_{P_j^{(i)}}, y_{P_j^{(i)}})$ with respect to sensor S_i. The first coordinate of point $P_j^{(i)}$ is $x_{P_j^{(i)}} = \tau_j^{(t-1)} \in [0, 1]$, where $\tau_j^{(t-1)}$ is the reputation of the sensor S_j conducting the evaluation. In fact, being $\tau_j^{(t-1)}$ the trust score computed at time $t-1$, it can be seen as the reputation gained by sensor S_j up to that moment. The second coordinate of point $P_j^{(i)}$ is $y_{P_j^{(i)}} = \sigma_j^{(i)}$, where $\sigma_j^{(i)} \in [0, 1]$ is the evidence by which sensor S_j evaluates sensor S_i. In other words, $\sigma_j^{(i)}$ is the expectation that sensor S_j has with respect to the future behavior of S_i.

b: Representation of τ_i'

Since the evidence relative to the trustworthiness of sensor S_i is represented as a value between 0 and 1, the evidence-based trust score τ_i' is also a value between 0 and 1. The idea is to define the data set $\mathcal{P}^{(i)} = \{P_1^{(i)}, \ldots, P_{i-1}^{(i)}, P_{i+1}^{(i)}, \ldots, P_n^{(i)}\}$ of points submitted by sensor S_j, for $j = 1, \ldots, n$ and $j \neq i$. Afterwards, K-means clustering and Gaussian mixtures are used to extract the evidence-based trust score $\tau_i^{(i)}$ from the coordinate $y_{P_j^{(i)}}$ of each point in the data set $\mathcal{P}^{(i)}$.

c: Classes of Credibility

K-classes of evidence are distinguished with respect to their credibility using the K-means clustering algorithm. The points in the data set $\mathcal{P}^{(i)}$ are grouped into K clusters $\mathcal{C}_1, \ldots, \mathcal{C}_K$. Each point in the data set $\mathcal{P}^{(i)}$ is a tuple corresponding to the values "reputation of the rater" and the values "the submitted rate/evidence". Therefore, the clustering algorithm finds classes which take into account both values. The center points $M_1, \ldots, M_K$ of clusters $\mathcal{C}_1, \ldots, \mathcal{C}_K$ simplify these classes of credibility with fewer, yet more informative points. An analogous view is that coordinates $y_{M_1}, \ldots, y_{M_K}$ of center points $M_1, \ldots, M_K$ represent how trustworthy sensor S_i is believed to be by the sensors in clusters $\mathcal{C}_1, \ldots, \mathcal{C}_K$. The classes of credibility may be shaped in any form of disjoint sub-intervals. We may consider, for example, dividing

the interval $[0, 1]$, representing the reputation $\tau_j^{(t-1)}$ of sensor S_j, into K disjoint sub-intervals $[0, \frac{1}{K}), \ldots, [1 - \frac{1}{K}, 1]$. This way, since clustering algorithms group together points around the same area, clusters $\mathcal{C}_1, \ldots, \mathcal{C}_K$ represent sensors with a similar reputation rate with sensor S_i.

d: Assigning Weight $\omega_j^{(i)}$ to Point $P_j^{(i)}$

Each point $P_j^{(i)}$ is submitted by sensor S_j, which has a reputation $\tau_j^{(t-1)}$. We wish to use this reputation to weight point $P_j^{(i)}$. We define weight $\omega_j^{(i)}$ as:

$$\omega_j^{(i)} = \frac{F(\tau_j^{(t-1)})}{\sum_{j=1}^{n-1} F(\tau_j^{(t-1)})},$$

where $F : [0, 1] \to \mathbb{R}$ is a positive and increasing function over the interval $[0, 1]$, which assigns higher scores for larger trust score $\tau_j^{(t-1)}$. The purpose of this function is to define how to balance the influence of low-weighted reputation against high-weighted reputation. In other words, function $F(x)$ determines how many low-weighted reputations are needed to have enough influence to overcome high-weighted reputation. This is desirable as sensors with a high reputation might also submit incorrect evidence. If many sensors, even with a low reputation, submit different evidence in comparison to the highly reputable CIDS, their opinion also has an impact. A possible approach to define function $F(x)$ is to choose a known increasing function (such as the logarithmic function) and adjust the eccentricity according to the number of low-weighted reputations necessary to balance higher-weighted reputations. Note that $F(x)$ does not need to be continuous. In fact, another possible approach to define function $F(x)$ is to create a step function over disjoint sub-intervals of the interval $[0, 1]$. Thus, two trust scores within the same sub-interval are considered equivalent with respect to the reputation of the sensor they represent.

e: Computation of τ_i'

The evidence-based trust score τ_i' is computed as a weighted combination of coordinates $y_{M_1}, \ldots, y_{M_K}$ of the center points $M_1, \ldots, M_K$, respectively. Center points $M_1, \ldots, M_K$ are not equivalent: together with the classes of credibility they distinguish, they depend on the cardinality of their respective clusters. The idea is to associate values $\pi_1, \ldots, \pi_K$ to center points $M_1, \ldots, M_K$ in quantitative and qualitative manner. Values $\pi_1, \ldots, \pi_K$ are regarded as the mixing coefficients of a mixture $p(\mathcal{P}^{(i)})$ of K Gaussian distributions $\mathcal{N}_1(\mu_1, \sigma_1^2), \ldots, \mathcal{N}_K(\mu_K, \sigma_K^2)$. More precisely, the points within each cluster $\mathcal{C}_l$ can be seen as following a Gaussian distribution with $\mu_l = M_l$, for $l = 1, \ldots, K$. That is because the mean and the variance of a Gaussian distribution convey information about where the points are mostly concentrated and how they are spread, which is comparable to the information conveyed by the clusters. Weights $\omega_j^{(i)}$ are used to compute the mixing coefficients $\pi_1, \ldots, \pi_K$. In more detail, $\pi_l = \sum_{j=1}^{n_l} \omega_{l_j}^{(i)}$, where n_l is the cardinality of cluster $\mathcal{C}_l$ and $\omega_{l_j}^{(i)}$ is the weight assigned to point $P_{l_j}^{(i)} \in \mathcal{C}_l$, for $l = 1, \ldots, K$. Note that $\sum_{l=1}^{K} \pi_l = 1$ and thus the mixture $p(\mathcal{P}^{(i)})$ is a probability distribution. The weighted sum of means $\mu_1, \ldots, \mu_K$ represents the mean μ of the closets Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ approximating mixture $p(\mathcal{P}^{(i)})$. And this is what is aimed at: since the means $\mu_1, \ldots, \mu_K$ are the center points $M_1, \ldots, M_K$, we can now compute the evidence-based trust score τ_i' as the weighted sum of coordinates $y_{M_1}, \ldots, y_{M_K}$ according to the mixing coefficients $\pi_1, \ldots, \pi_K$. That is,

$$\tau_i' = \sum_{l=1}^{K} \pi_l \cdot y_{M_l} \in [0, 1].$$

In other words, the evidence-based trust score is computed as coordinate y_μ of the mean μ of the fitting Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$. One can argue that the evidence-based trust score τ_i' could be computed directly after clusters $\mathcal{C}_1, \ldots, \mathcal{C}_K$ were distinguished, without passing through the step of computing the mixture of Gaussians. This is, in fact, what one would practically do when computing τ_i'. However, we highlight that this computation is possible because the center points of the clusters model are the means of Gaussian distributions.

C. RELIABILITY-BASED TRUST SCORE

This section describes how the reliability-based trust score τ_i'' for sensor S_i is computed taking into account the reliability of the evidence submitted by sensor S_i itself with respect to all the other sensors. We recall that the reliability-based trust score is meant to distinguish submitted reliable evidence from unreliable evidence and to, respectively, encourage and discourage such submissions. Just as with the evidence-based trust score τ_i', the computation of the reliability-based trust score τ_i'' is performed in an Euclidean space of two dimensions $D = 2$. For readability, we divide this computation into four steps.

a: Evidence Collection

The process begins by collecting all evaluations, known as evidence, issued by each sensor. This evidence is defined as $\sigma_i^{(j)} \in [0, 1]$; that is, the coordinate $y_{P_i^{(j)}}$ of point $P_i^{(j)}$ submitted by the evaluator sensor S_i with respect to sensor S_j, for $j = 1, \ldots, n$ and $j \neq i$, where point $P_i^{(j)} = (x_{P_i^{(j)}}, y_{P_i^{(j)}})$ is defined in Section III-B.

b: Representation of τ_i''

Since the evidence relative to the reliability of the submissions of sensor S_i is represented as a value between 0 and 1, the reliability-based trust score τ_i'' is also a value between 0 and 1. The reliability of evidence $\sigma_i^{(j)}$ is measured with the distance $d(\tau_j', \sigma_i^{(j)})$, where τ_j' is the evidence-based trust score of sensor S_j. Distance $d(\tau_j', \sigma_i^{(j)})$ is at most 1. If distance $d(\tau_j', \sigma_i^{(j)})$ is close to 1, this is an indicator of the dishonesty of sensor S_i when rating sensor S_j; that is, it is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access

an indicator that $\sigma_i^{(j)}$ is an unreliable piece of evidence. On the contrary, if distance $d(\tau'_j, \sigma_i^{(j)})$ is close to 0, this is an indicator of the honesty of sensor S_i when rating sensor S_j; in other words, it is an indicator that $\sigma_i^{(j)}$ is a reliable piece of evidence.

c: Reliability Score

Ranges of reliability are designed to grant trustworthiness to sensor S_i when distance $d(\tau'_j, \sigma_i^{(j)})$ is small and vice versa. We aim at designing a mechanism that increases τ''_i when S_i is honest and decreasing τ''_i, in variable amounts, when S_i is dishonest. We achieve this by assigning a reliability score $o_i^{(j)}$ to each value $d(\tau'_j, \sigma_i^{(j)})$ of S_i for $j = 1, \ldots, n$ and $j \neq i$. The evidence $\sigma_i^{(j)}$ submitted by S_i is rated as $o_i^{(j)}$ depending on how close it is to τ'_j. The reliability score $o_i^{(j)}$ is a value in the set

$$\mathbf{O} = \{x : x = \alpha_1 + (-\alpha_1 \times n), n \in \{0, 1, 2, \ldots, (\alpha_2 - 1)\}\},$$

where $\alpha_1 \in \{0, 1\}$ specifies, both, a maximum reward and a series of penalty values; and $\alpha_2 \in \mathbb{Z}^+$ specifies the total number of elements, or steps, in $\mathbf{O}$. Intuitively, the first value of $\mathbf{O}$ is α_1, the second value is 0 and subsequent values are multiples of $-\alpha_1$, with a total of α_2 elements. The element $o \in \mathbf{O}$ is assigned to the distance $d(\tau'_j, \sigma_i^{(j)})$ at index $\lfloor d(\tau'_j, \sigma_i^{(j)}) \cdot \alpha_2 \rfloor$. With such an interpretation, if distance $d(\tau'_j, \sigma_i^{(j)})$ is between the range $\left[0, \frac{1}{\alpha_2}\right]$, the first element of $\mathbf{O}$, namely α_1, is assigned as $o_i^{(j)}$.

Because all submitted evidence is rated according to its reliability, sensors are encouraged to submit reliable evidence and discouraged to submit unreliable one. If a sensor submits unreliable evidence time after time, then it will progressively lose more and more trust. This is a countermeasure that discourages the submission of unreliable evidence, as the trustworthiness of the submitter decreases.

d: Computation of τ''_i

The reliability-based trust score τ''_i is computed from the reputation $\tau_i^{(t-1)}$ of sensor S_i at time $t-1$, taking into account the average reliability score of the $n-1$ scores $\sigma_i^{(j)}$ it submitted, for $j = 1, \ldots, n$ with $j \neq i$. That is,

$$\tau''_i = \tau_i^{(t-1)} + \frac{1}{n-1} \sum_{j=1, j\neq i}^{n} o_i^{(j)}.$$

In this way, the reliability-based trust score τ''_i is computed by increasing $\tau_i^{(t-1)}$ if the scores $\sigma_i^{(j)}$ are on average reliable or by decreasing $\tau_i^{(t-1)}$ if the scores $\sigma_i^{(j)}$ are unreliable on average. Note that the computation of the reliability-based trust score τ''_i is recursive. That is, the history of the behavior of sensor S_i in the previous rounds is taken into account by the term $\tau_i^{(t-1)}$. In fact, reputation is built upon consistent increments over a long period of time and is not greatly affected, both positively nor negatively, in one single recursion.

D. FINAL TRUST SCORE

Trust score τ_i is computed as a convex combination of τ'_i (Section III-B) and τ''_i (Section III-C). That is, the parameter $\eta \in [0, 1]$ is selected such that $\eta + (1 - \eta) = 1$ and trust score τ_i is:

$$\tau_i = \eta \cdot \tau'_i + (1 - \eta) \cdot \tau''_i. \quad (1)$$

The parameter η holds for the computation of each trust score τ_i, for $i = 1, \ldots, n$. It is chosen based on the requirements of the specific CIDS. In some situations, it might be more desirable to assign more weight to a sensor performing well rather than a sensor rating honestly and vice versa.

IV. EVALUATION

Sphinx is capable of identifying coalitions if less than 50% of all sensors collude in a single coalition. In certain conditions, when multiple and independent coalitions exist, *Sphinx* can identify dishonest participants even when more than 50% of all sensors are dishonest. This section gives evidence to support these claims in the form of different evaluation experiments.

A. EXPERIMENTAL SETUP

Experiments are performed in rounds. Rounds represent the basic time unit used in all experiments. At each round t, the evidence-based (Section III-B) and reliability-based (Section III-C) trust scores, $\tau'^{(t)}$ and $\tau''^{(t)}$, respectively, are calculated for each sensors participating in a CIDS. These two trust scores are combined together, according to Equation (1), to obtain the trust score $\tau^{(t)}$ of each server at round t. The evidence and reliability-based trust scores use the previous trust score $\tau^{(t-1)}$ to perform their calculations; therefore, making the whole process recursive.

All experiments assume that sensors behave in one of two different ways: either they are honest or dishonest. An honest sensor always rates others in accordance to its empirical observations. A dishonest sensor, on the other hand, rates others better or worse depending on some convenience factor. If a dishonest sensor is acting alone, without the support of others, it resorts to rating every other sensor worse than what it itself empirically observed. In doing so, the supposition is that his trust score would eventually become the best if the score of everyone else worsens. These single dishonest sensors are termed lone colluders.

The experiments take into account all assumptions stated in Section III-A. The following is a summary of these. Dishonest sensors are able to collaborate with others to form *coalitions*. Within one single CIDS, more than one coalition might be present. A coalition has the goal of increasing the trust of all its sensors while trying to reduce the trust of any outsider. All sensors of a coalition act according to the following three rules:

1) When rating other sensors in the same coalition, the rating is improved relative to the observed sensing reliability of the other sensor.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access

2) The sensors of a coalition rate others, not belonging to the same coalition, with a worsened rating relative to the other sensor's observed sensing reliability. With some probability, dishonest sensors can give honest ratings to try and fool the system.
3) Dishonest sensors can only belong to one coalition.

All non-deterministic experiments are repeated 50 times. Instead of showing aggregated graphs of all experiments, we choose to show one descriptive scenario that represents the experiments well. Once parameters of *Sphinx* are fixed, the algorithm is deterministic except for how dishonest sensors alter ratings. Dishonest sensors use a Beta distribution to select a value to add or subtract from a trust score before submission (depending on whether they are improving or worsening the score). This stochastic selection of values does not modify the trend of the experiments, therefore, making the variance of the results small.

1) Parameters of the Rating System

Dishonest sensors improve or worsen ratings according to a Beta distribution. Whenever a dishonest sensor S_j needs to submit a rating, or evidence, for S_i, it does so using:

$$x \sim Beta(a, b) \tag{2}$$

$$\sigma_j^{(i)} = \tau_i \pm x, \tag{3}$$

where a and b are chosen parameters for a Beta distribution according to one of the combinations shown in Figure 1. In Equation (2), a sample x of the chosen Beta distribution is obtained. This sample is either added or subtracted, in Equation (3), depending on whether the trust score τ_i is improved or worsened. We choose the Beta distribution as it models dishonest members that alter evidence conservatively most of the time and aggressively a few times with low probability.

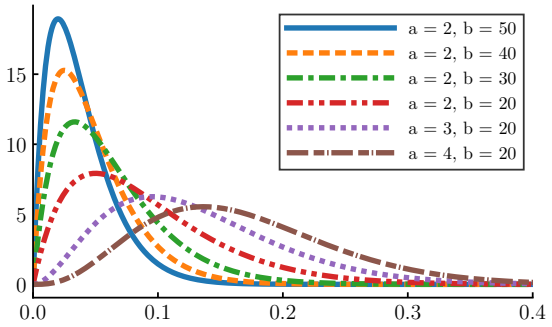


FIGURE 1: Family of Beta distributions that specifies how much ratings are improved or worsen by dishonest participants in a CIDS.

In contrast to a dishonest sensor, honest sensor S_j submits ratings, or evidence, of another sensor S_i using a Gaussian model; that is,

$$y \sim \mathcal{N}(0, std) \tag{4}$$

$$\sigma_j^{(i)} = \tau_i + y. \tag{5}$$

A sample y from a Gaussian distribution with standard deviation std is obtained. This sample is then added to the true trust score τ_i to account for the potential inaccuracies in the empirical observations S_j might have had. In all experiments, std is chosen such that $std < \mathbb{E}[Beta(a, b)]$. If this is not the case, the samples of the Gaussian distribution used by honest sensors overlaps greatly with the samples of the Beta distribution used by dishonest sensors, making honest and dishonest behaviors almost indistinguishable.

2) Parameters of *Sphinx*

The calculation of trust scores depend on some user-supplied parameters. The evidence-based trust score τ' depends on two parameters: a score weight function $F(x)$ and the number of cluster centers K (see Section III-B0c). The reliability-based trust score τ'' relies on two parameters: a reward parameter α_1 and the number of penalty subdivisions α_2 (see Section III-C0c). The combination of both evidence and reliability-based trust scores into the final trust τ depends on the mixing coefficient η (see Section III-D). In most experiments, all these parameters are fixed to a single set to demonstrate how generalizable a single set of parameters can be.

B. EXPERIMENTS

We conduct a series of experiments that demonstrate how the trust scores of lone colluders and coalitions are penalized. Unless explicitly indicated, all experiments use the following parameters:

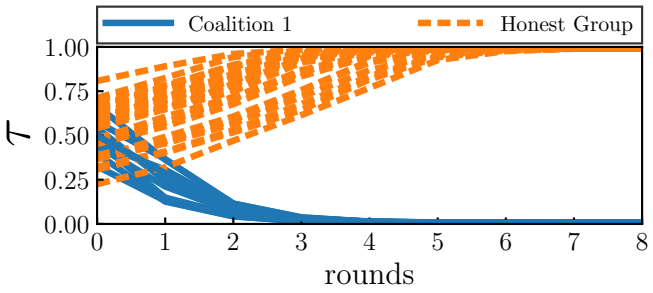
$$F(x) = \begin{cases} 1 & \text{if } 0.00 \leq \tau \leq 0.25 \\ 2 & \text{if } 0.25 < \tau < 0.50 \\ 3 & \text{if } 0.50 < \tau \leq 0.75 \\ 4 & \text{if } 0.75 < \tau \leq 1.00 \end{cases},$$

$K = 2$, $\alpha_1 = 0.20$, $\alpha_2 = 25$ and $\eta = 0.30$. For the generation of ratings by honest and dishonest sensors, we choose the parameters $a = 3, b = 20$ for the Beta distribution in Equation (2) and the parameter $std = 0.05$ for the Gaussian distribution in Equation (4). The chosen parameters of the Beta distribution model a distribution with an expected value of $\mathbb{E}[Beta(3, 20)] = 0.13$. This implies that, most of the time dishonest sensors increase (or decrease) the rating of others by 0.13 points. With this particular Beta distribution, the increase (or decrease) can range from small values close to 0.0 up to the high value 0.35 (with low probability). This models dishonest sensors that choose to be conservative most times but sporadically aggressive.

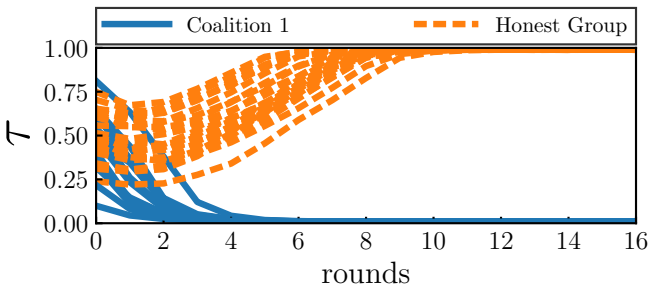
1) Experiment 1: Detecting Single Large Coalitions

Sphinx can detect single large coalitions as long as the number of sensors of that coalition do not exceed the number

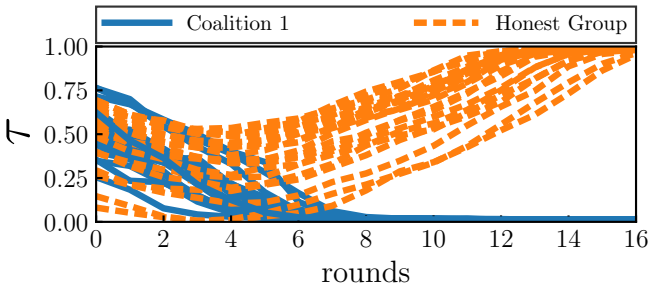
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access



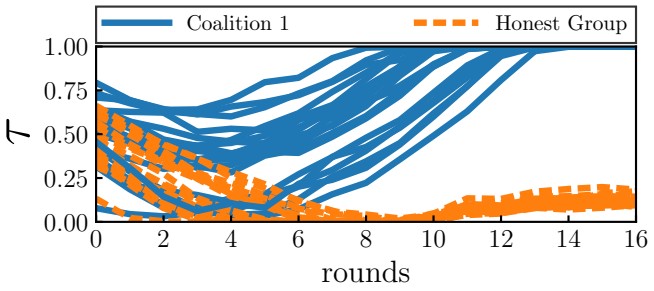
(a) 40 sensors in total, 25% of all sensors are dishonest. The colluding sensors are easily identified; their trust rapidly drops to zero.



(b) 40 sensors in total, 37.5% of all sensors are dishonest. In the first two rounds, the coalition can negatively affect the honest sensors. After the trust of the coalition drops to zero, the honest sensors only increase their trust.



(c) 40 sensors in total, 47.5% of all sensors are dishonest. Same effect as when considering 37.5% of dishonest sensors. With more dishonest sensors, the effect last longer until about round six.



(d) 40 sensors in total, 50% of all sensors are dishonest. The dishonest sensors overwhelm the honest sensors and they easily drop the rating of the honest sensors to low values.

FIGURE 2: Change of trust with every new calculation (round) of *Sphinx*. Four different scenarios are evaluated: when 25%, 37.5%, 47.7% and 50.0% of the collaborating sensors form a dishonest coalition.

of honest sensors. This is illustrated in Figure 2. The x-axis of each plot shows the number of rounds or, in other words, the number of times our recursive trust mechanism is calculated. Round zero is the case where *Sphinx* has not yet been run; therefore, it represents the initial bootstrapped trust of each sensors. All sensors are bootstrapped with an initial trust score following the Gaussian distribution $\mathcal{N}(0.50, 0.15)$. On the y-axis, the trust scores τ are shown.

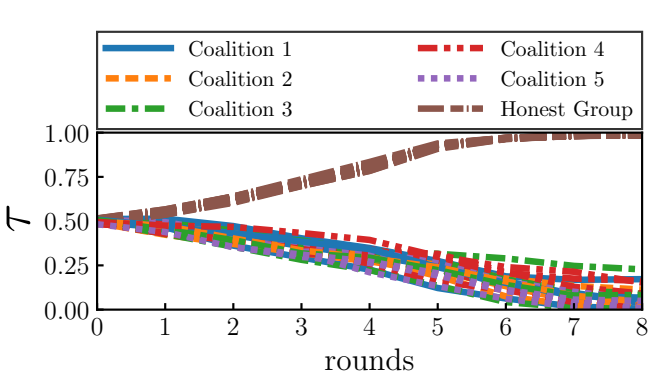
In each of the four plots of Figure 2, setups of different dishonest and honest participants are tested. Figure 2a shows that it is easy to detect coalitions with 25.0% of dishonest sensors. Figures 2b and 2c show that, although there is some negative influence from the coalition for a few rounds, the trust of the coalition falls to zero. This is taking into account the fact that in Figure 2c 47.5% of all sensors (19 out of 40) are dishonest. If 50.0% of all sensors are part of a coalition, however, the trust model fails to punish dishonesty as the results in Figure 2d show.

2) Experiment 2: Detecting Multiple Coalitions

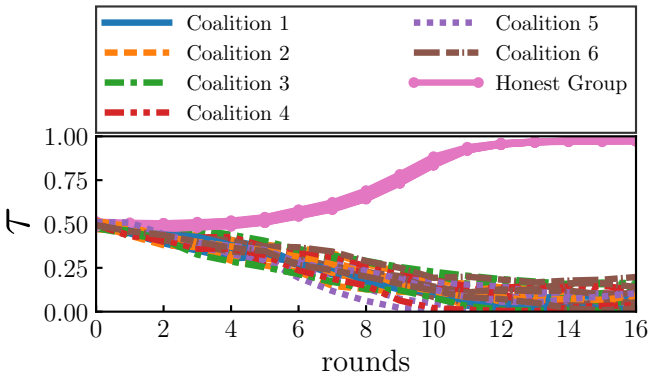
Sphinx is capable of recognizing and punishing independent coalitions. With multiple coalitions, even if more than 50.0% of the total sensors are dishonest, honesty is successfully rewarded while dishonesty is punished. In Figure 3, we can appreciate how our methodology performs when dishonest participants are moderately dishonest using a Beta distribution with parameters $a = 3$ and $b = 20$. From a total of 50 sensors, a varying amount of dishonest sensors in different coalitions are shown. In Figure 3a, five coalitions are tested, each having 5 sensors, amounting to 50% of all sensors. Similarly in Figure 3b, six coalitions with 5 sensors each, representing 60% of all sensors, are tested. Honesty is successfully rewarded while dishonesty is punished in both scenarios.

When dishonest sensors use the Beta distribution with parameters $a = 3$ and $b = 20$, adding more coalitions (of 5 sensors) would result in an ecosystem where no individual or coalition can increase its trust score beyond 0.25. The dishonest ratings effectively deny the possibility of gaining trust. If the dishonest sensors are less conservative and modify their ratings using a more aggressive Beta distribution with parameters $a = 2$ and $b = 10$, honesty is still recognized and rewarded as shown in Figure 4. During the early rounds of our methodology, the trust score of everyone is heavily decremented. After the trust scores have settled to low values, honesty can be recognized again. In round 12, dishonest sensors no longer have enough trust and their evidence submissions stop having much weight. It is in this scenario that honesty can slowly build up trust once again.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access



(a) 50 sensors and 5 coalitions of 5 sensors each. 50% of all sensors are dishonest. Although 50.0% of all sensors are dishonest, the fact that no coalition has more than the total number of honest sensors, dishonesty can still be successfully punished. The trust score of all sensors in all coalitions rapidly drops to zero. The trust score of the honest sensors rapidly increases.



(b) 50 sensors and 6 coalitions of 5 sensors each. 60% of all sensors are dishonest. Dishonesty is correctly identified and punished. However, honest sensors are also punished in the beginning and then start to be rewarded. Notice that double the rounds were needed to detect dishonesty as when 50% of the sensors were dishonest.

FIGURE 3: Change of trust with every new calculation (round) of *Sphinx*. Four different scenarios are evaluated: when 25%, 37.5%, 47.7% and 50.0% of the collaborating sensors form a dishonest coalition.

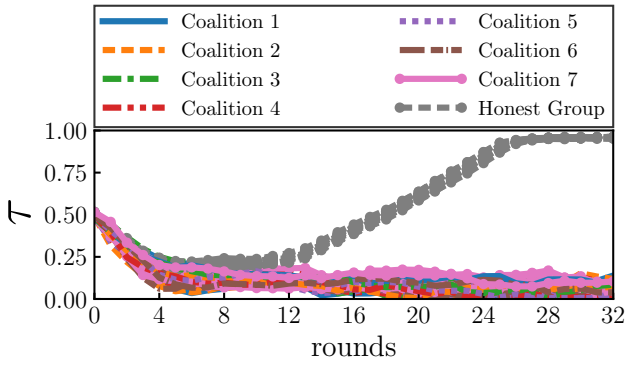


FIGURE 4: 50 sensors and 7 coalitions of 5 sensors each. 70% of all sensors are dishonest. If the dishonest sensors are less conservative, honest sensors are eventually recognized.

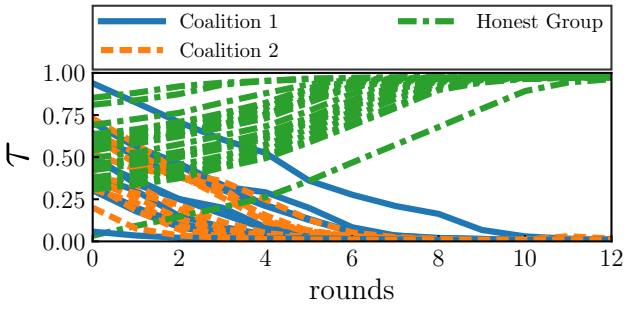


FIGURE 5: Evolution of the trust scores (τ) of two coalitions, each with 10 sensors, and 20 honest sensors when the trust scores are initially dispersed. The honest sensor with the lowest score (of 0.08) reaches a trust of 1.00 after 11 rounds.

3) Experiment 3: The Effects of Disperesed Intitial Trust Scores

The previous experiment assumed that all sensors started with a trust score close to 0.5. Initializing the trust scores over the range $[0, 1]$ has no negative influence on the capabilities of *Sphinx* to detect dishonesty. Taking into account 20 honest sensors and two coalitions of 10 sensors each (for a total of 20 dishonest sensors), Figure 5 shows how honest sensors with a low starting trust score are able to reach high trust scores. Similarly, dishonest sensors starting with high trust scores get their score reduced to zero given enough rounds. Shown in the figure, the honest sensor with the lowest starting trust score (of 0.08) is able to reach a maximum score in eleven rounds. The dishonest sensor with the highest initial trust score (of 0.95) is reduced to a score of zero after ten rounds.

4) Experiment 4: Sensibility of Dishonesty

Starting with the trust score of each server initialized with the Gaussian distribution $\mathcal{N}(0.5, 0.2)$, we examine how changing the sensibility of dishonesty affects the capability of identifying coalitions. In our experiments, we observed that honesty can be identified when $std < \mathbb{E}[Beta(a, b)]$ (see Section IV-A1). In our setup, we assume that $std = 0.05$, i.e., honest sensors approximate the real score τ_i of sensor S_i with $\hat{\tau}_i$ such that $\hat{\tau}_i = \mathcal{N}(\tau_i, 0.05)$. From the illustrated Beta distributions in Figure 1, it is possible to detect dishonest sensors that act according to a Beta distribution parameterized with $a = 2$ and $b = 30$ (where $\mathbb{E}[\beta(2, 30)] = 0.062$) and below. Conversely, if dishonesty is modeled with the Beta distribution $\beta(2, 40)$ or $\beta(2, 50)$, dishonesty cannot be identified as it is easy to confuse with (honest) mistakes.

Figure 6 shows the average trust scores τ with standard deviations after executing 12 rounds of our trust mechanism.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access

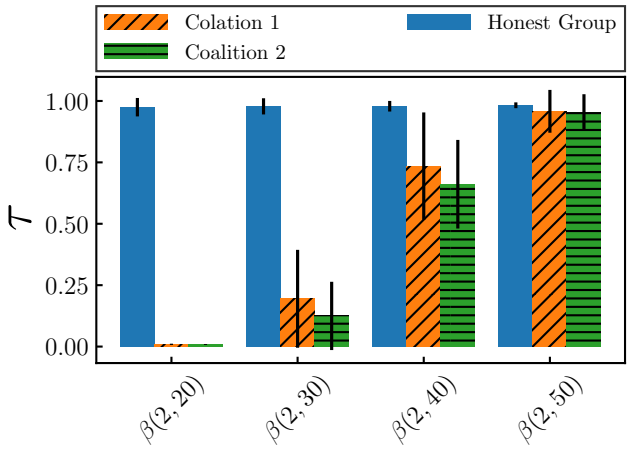


FIGURE 6: Average trust score τ of honest and dishonest sensors when the sensibility of dishonesty is varied after 12 rounds of executing our trust mechanism. From left to right, more subtle Beta distributions are used by dishonest sensors to improve or worsen the trust score of others. When using $\beta(2, 30)$ or $\beta(2, 20)$, the trust scores of both coalitions are kept in line. Using more subtle Beta distributions results in the trust scores of the coalitions also increasing (without affecting the honest participants).

When the coalitions are dishonest according to the Beta distribution $\beta(2, 20)$ and $\beta(2, 30)$, the average trust score of the dishonest participants is kept low. In all repeated experiments, the average trust score of all coalitions would tend towards 0.0. This is, however, not the case when coalitions act according to the Beta distributions $\beta(2, 40)$ or $\beta(2, 50)$. In both cases, the average trust scores cannot be kept low and have a tendency to increase towards 1.0. This is due to the fact that with such Beta distributions, it is not possible to distinguish dishonesty from honest mistakes. Note that the trust scores of the honest sensors are not negatively affected this way.

5) Experiment 5: Dealing with Smarter Dishonest Sensors

In this scenario, we describe the effects on honest and dishonest sensors when those being dishonest choose to honestly submit evidence according to some probability p. In Figure 7, we illustrate four scenarios that take into account 40 sensors, different ratios of dishonest sensors, and different values of p. Figures 7a and 7b duplicate the conditions of the experiments illustrated in Figure 2a but incorporate p. In Figure 7a, each sensor of the coalition chooses to be honest with a probability of 20% ($p = 0.2$). In contrast to the results obtained when $p = 0$ (cf. Figure 2a), the coalition's trust scores stay slightly higher in round two but almost collapse by round three. With $p = 0.5$, as illustrated in Figure 7b, the coalition's trust scores are more slowly punished. In the last two scenarios, the honest sensors' trust scores are barely affected.

Figures 7c and 7d duplicate the conditions of the experiments shown in Figure 2c but use different values of p. As shown in Figure 7c, with $p = 0.2$, the coalition is not successful in keeping their trust scores relevant. By round eight, all trust scores collapse. Surprisingly, the honest sensors' trust scores are positively rewarded by the choices of the coalition. The honest sensors' trust scores do not decrease as they did in Figure 2c and are maximized by round 11. In the last experiment, shown in Figure 7d, sensors in the coalition choose to be honest 50% of the time. In this scenario, the coalition's trust scores stay relevant longer but, with enough rounds, eventually collapse. However, the honest sensors' trust scores are positively affected and reach their maximum by round eight.

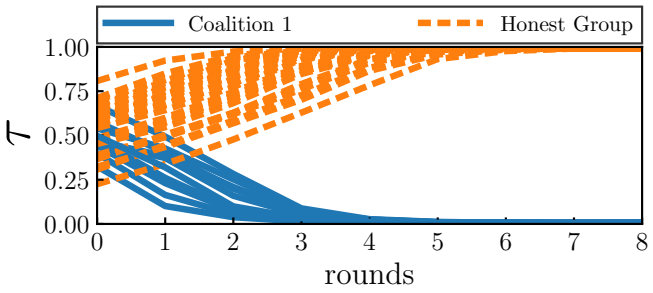
Overall, dishonest sensors that choose to act honestly with some probability p must trade between staying relevant longer and rewarding honest members to increase their score even faster. In the previously described scenario, we found a turning point when $p = 0.65$, for which the results are shown in Figure 8. At this turning point, sensors in coalition 1 choose to be honest 65% of the time and manage to keep their trust scores almost constant for all 32 rounds. If the coalition wishes to get the trust scores τ of their sensors to rapidly increase (and not be left behind by the honest sensors), they would need to be honest 80% of the time. This would enable them to give some dishonest ratings without being heavily punished but would go against the goal of reducing the honest sensors' trust scores.

V. CONCLUSION AND FUTURE WORK

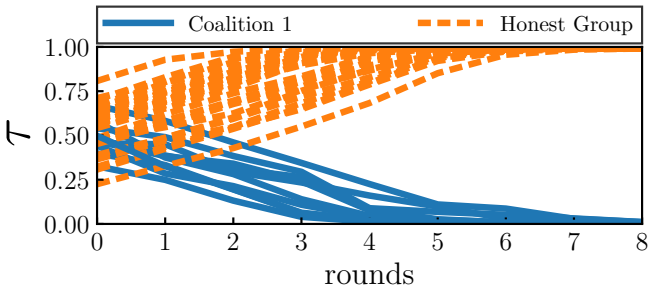
In this article, *Sphinx* a colluder-resistant trust mechanism for CIDSs that is capable of identifying and punishing coalitions of dishonest CIDS sensors is presented. *Sphinx* uses unsupervised machine learning techniques to collect and process the evidence submitted by sensors. Through these methods, it is possible to establish a trust score for each CIDS sensor based on its trustworthiness as well as the reputation of its submitted evidence. Using this novel approach, it is possible to detect unreliable evidence and effectively penalize dishonest sensors by reducing their trust even if they are part of a coalition. In the evaluation section, it was shown how *Sphinx* detects single large coalitions as long as the majority of the CIDS sensors are honest. When there are multiple independent coalitions, even when the large majority is dishonest, *Sphinx* is still able to identify and punish dishonesty. The only necessary condition is that the largest independent coalition must be smaller than the number of honest sensors. For instance, in a scenario with two coalitions of 10 sensors each (a total of 20 dishonest sensors) and 15 honest sensors, dishonesty can still be detected.

As future work, *Sphinx* can be extended to compute the trust scores according to additional criteria, such as the amount of past interactions. Also, in this case, clustering algorithms can be used to process all this information and define proper credibility classes of evidence. Furthermore, we plan to design a new bootstrapping procedure that improves upon [12] so as to better address the scenario of trust within CIDSs.

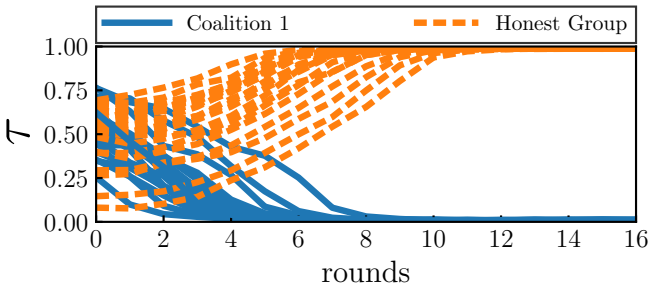
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access



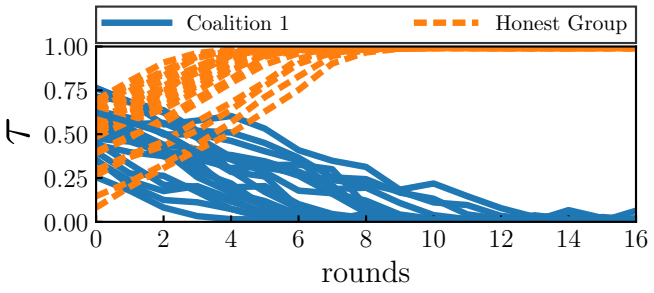
(a) 40 sensors, 25% dishonest, 20% probability of dishonest sensors being honest. When colluders are 20% of the times honest, honest sensors increase their trust scores faster. Dishonest sensors are still identified.



(b) 40 sensors, 25% dishonest, 50% probability of dishonest being sensors honest. Colluders are correctly identified, albeit in later rounds.



(c) 40 sensors, 47.5% dishonest, 20% probability of dishonest sensors being honest. When sensors in coalition 1 rate honestly with a 20% probability, instead of harming the honest sensots, the coalition boosts the trust scores of the honest members without gaining enough trust.



(d) 40 sensors, 47.5% dishonest, 50% probability of dishonest sensors being honest. With a 50% probability of rating honestly, coalition 1 further boosts the scores of the honest sensors. Their trust scores, however, are more slowly reduced.

FIGURE 7: *Sphinx*'s performance when dishonest sensors are smarter, i.e., they act honestly with some probability. The top and bottom rows replicate the conditions used in the experiments shown in Figures 2a and 2c, respectively, but consider smarter dishonest sensors.

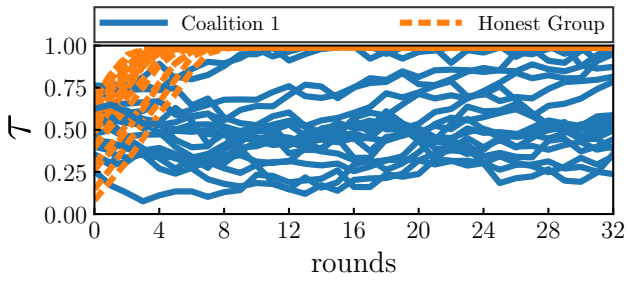


FIGURE 8: 40 sensors, 47.5% dishonest, 65% probability of dishonest sensors being honest. The coalition's trust scores almost stay constant throughout 32 rounds of running *Sphinx*. The coalition can be identified by the fact that their trust scores do not increase as those of the honest sensors.

ACKNOWLEDGMENTS

The authors would like to thank Guido Salvaneschi, Moritz Horsch, Alex Palesandro and Denis Butin for their input and constructive discussions. This work has been co-funded by the DFG as part of the projects "Scalable Trust Infrastructures" and "Long-Term Secure Archiving" within the CRC 1119 CROSSING and the European Union's Horizon 2020 research and innovation program under Grant Agreement No 644962. The research leading to these results has also received funding from the European Union's Horizon 2020 Research and Innovation Program, PROTECTIVE, under Grant Agreement No 700071. We acknowledge support by the German Research Foundation and the Open Access Publishing Fund of Technische Universität Darmstadt.

REFERENCES

[1] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis et al., "Understanding the mirai botnet," in USENIX Security Symposium, 2017, pp. 1092–1110.

[2] G. Meng, Y. Liu, J. Zhang, A. Pokluda, and R. Boutaba, "Collaborative security: A survey and taxonomy," ACM Computing Surveys (CSUR), vol. 48, no. 1, p. 1, 2015.

[3] E. Vasilomanolakis, S. Karuppayah, M. Muehlhaeuser, and M. Fischer, "Taxonomy and Survey of Collaborative Intrusion Detection," ACM Computing Surveys (CSUR), vol. 47, no. 4, pp. 1–35, 2015.

[4] G. Traverso, C. G. Cordero, M. Nojoumian, R. Azarderakhsh, D. Demirel, S. M. Habib, and J. Buchmann, "Evidence-Based Trust Mechanism Using Clustering Algorithms for Distributed Storage Systems," in 15th Annual Conference on Privacy, Security and Trust (PST), aug 2017.

[5] A. Josang and R. Ismail, "The beta reputation system," in Proceedings of the 15th bled electronic commerce conference, vol. 5, 2002, pp. 2502–2511.

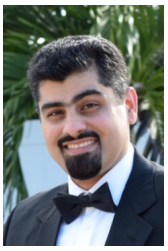
[6] S. Buchegger and J. Boudec, "A robust reputation system for peer-to-peer and mobile ad-hoc networks," in Workshop on the Economics of Peer-to-Peer Systems, 2004.

[7] M. Nielsen, K. Krukow, and V. Sassone, "A bayesian model for event-based trust," Electronic Notes in Theoretical Computer Science, vol. 172, pp. 499–521, 2007.

[8] C.-W. Hang and M. P. Singh, "Trustworthy service selection and composition," ACM Trans. Auton. Adapt. Syst., vol. 6, no. 1, pp. 5:1–5:17, Feb. 2011.
[9] S. Ries, "Extending bayesian trust models regarding context-dependence and user friendly representation," in Proceedings of the 2009 ACM Symposium on Applied Computing (SAC), Honolulu, Hawaii, USA, 2009, pp. 1294–1301.
[10] W. M. Bolstad, Introduction to Bayesian Statistics. John Wiley & Sons, Inc, 2004.
[11] M. Nielsen, G. Plotkin, and G. Winskel, "Petri nets, event structures and domains, part i," Theoretical Computer Science, vol. 13, no. 1, pp. 85 – 108, 1981.
[12] X. Liu, A. Datta, K. Rzadca, and E.-P. Lim, "Stereotrust: A group based personalized trust model," in Proceedings of the 18th ACM Conference on Information and Knowledge Management, ser. CIKM '09. New York, NY, USA: ACM, 2009, pp. 7–16.
[13] X. Liu, G. Tredan, and A. Datta, "A generic trust framework for large-scale open systems using machine learning," Computational Intelligence, vol. 30, no. 4, pp. 700–721, 2014.
[14] J. Tang, H. Gao, H. Liu, and A. Das Sarma, "etrust: Understanding trust evolution in an online world," in Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '12. New York, NY, USA: ACM, 2012, pp. 253–261.
[15] C. M. Bishop, Pattern recognition and machine learning, 5th Edition, ser. Information science and statistics. Springer, 2007.
[16] H. W. Lauw, E.-P. Lim, and K. Wang, "Bias and controversy: Beyond the statistical deviation," in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '06. New York, NY, USA: ACM, 2006, pp. 625–630.
[17] M. Moe, B. Helvik, and S. Knapskog, "Comparison of the beta and the hidden markov models of trust in dynamic environments," Trust Management III, pp. 283–297, 2009.
[18] C. J. Fung, J. Zhang, I. Aib, and R. Boutaba, "Dirichlet-based trust management for effective collaborative intrusion detection networks," IEEE Transactions on Network and Service Management, vol. 8, no. 2, pp. 79–91, 2011.
[19] M. Gil Pérez, F. Gómez Mármol, G. Martínez Pérez, and A. F. Skarmeta Gómez, "RepCIDN: A reputation-based collaborative intrusion detection network to lessen the impact of malicious alarms," Journal of Network and Systems Management, vol. 21, no. 1, pp. 128–167, 2013.
[20] C. J. Fung, O. Baysal, J. Zhang, I. Aib, and R. Boutaba, "Trust management for host-based collaborative intrusion detection," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 5273 LNCS, pp. 109–122, 2008.
[21] C. J. Fung, J. Zhang, I. Aib, and R. Boutaba, "Robust and scalable trust management for collaborative intrusion detection," 2009 IFIP/IEEE International Symposium on Integrated Network Management, IM 2009, pp. 33–40, 2009.
[22] R. Dwarakanath, B. Koldehofe, Y. Bharadwaj, T. A. B. Nguyen, D. Eyers, and R. Steinmetz, "Trustcep: Adopting a trust-based approach for distributed complex event processing," in 2017 18th IEEE International Conference on Mobile Data Management (MDM). IEEE, 2017, pp. 30–39.

CARLOS GARCIA CORDERO is a security researcher versed in the fields of network security, artificial intelligence, programming languages, compilers, machine learning and computer graphics. He is currently pursuing a PhD in network security and machine learning in Germany. Besides his academic experience, Carlos has worked in the security, video game and software development sectors. He further describes himself as an enthusiastic programmer, mathematician, musician and thinker.

GIULIA TRAVERSO is a Ph.D. candidate in cryptography at the Technische Universität Darmstadt (Germany), under the supervision of Prof. Johannes Buchmann at CDC group. Her research focuses on the long-term security of distributed storage systems. She joined CDC in 2015 after receiving her B.SC. and M.Sc. degrees in mathematics at the University of Trento (Italy) and after an internship at IdQuantique in Geneva (Switzerland), a company producing quantum devices.



PROF. DR. MEHRDAD NOJOUMIAN is an assistant professor at the Florida Atlantic University (USA). He received his Ph.D. degree from the Cheriton School of Computer Science at University of Waterloo and his M.Sc. degree from the School of Electrical Engineering and Computer Science at University of Ottawa. His research interests lie on trust management, autonomous systems, applied cryptography, security and privacy, game theory, and any interdisciplinary research on the intersection of computer science and economics.

PROF. DR. JOHANNES BUCHMANN received a PhD from the Universität zu Köln, Germany, in 1982. From 1985 to 1986 he was a PostDoc at the Ohio State University on a Fellowship of the Alexander von Humboldt Foundation. From 1988 to 1996 he was a professor of Computer Science at the Universität des Saarlandes in Saarbrücken. Since 1996 he has been a professor of Computer Science and Mathematics at TU Darmstadt. From 2001 to 2007 he was Vice President Research of TU Darmstadt. In 1993, he received the Leibniz-Prize of the German Science Foundation (DFG) and in 2017 the Konrad Zuse Medal of the German Informatics Society (GI). He is a member of the German Academy of Science and Engineering acatech and of the German Academy of Science Leopoldina.

DR. EMMANOUIL VASILOMANOLAKIS is a senior researcher in the Technische Universität Darmstadt in Darmstadt, Germany. His research interests include collaborative intrusion detection, honeypots, botnet monitoring and alert data correlation. He received a PhD from the Technische Universität Darmstadt in 2016 for his dissertation "On Collaborative Intrusion Detection". Heretofore, he received his diploma (Dipl.-Inform.) and MSc from the University of the Aegean (Greece) in 2008 and 2011 respectively. Emmanouil has published in major scientific conferences, workshops and journals on topics related to the field of cybersecurity (including ACM Computing Surveys, IEEE CNS, IEEE ICC, RAID and Blackhat). Lastly, he worked as a researcher for AGT International, on the field of IoT security, from 2014–2015. Emmanouil is a member of IEEE as well as of the Honeynet Project.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2880297, IEEE Access

DR. SHEIKH MAHBUB HABIB holds a position as Automotive Security & Privacy Specialist under the Security and Privacy Competence Center (SCC) of Continental AG Germany. Before joining Continental, he was leading the SPIN (Smart Protection in Infrastructures and Networks) area in the Telecooperation (TK) Lab of Technische Universität Darmstadt (TU Darmstadt). He holds a doctoral degree (Dr. rer. nat.) in Computer Science focusing on IT Security from TU Darmstadt Germany. He holds a M. Sc. degree in Computer Science and Engineering from Chalmers University of Technology Sweden and a B.Sc. degree in Computer Science and Engineering from Khulna University of Engineering and Technology Bangladesh. During his academic duty, he was involved in acquiring several Cybersecurity research projects, namely CROSSING, PAT, and PROTECTIVE, funded by the German Research Foundation (DFG) and European Commission (EC). His research interests are computational trust models, trust management, mobile application security & privacy, and vehicular security & privacy. Dr. Habib is serving as a reviewer in several top journals like IEEE TIFS, IEEE TCC, ACM TOPS, and ACM CSUR. He is currently appointed as "Publicity Chair" in the IFIP WG 11.11 (Trust Management).

PROF. DR. MAX MÜHLHÄUSER is head of the Telecooperation Lab at Technische Universität Darmstadt, Informatics Dept. His Lab conducts research on smart ubiquitous computing environments for the "pervasive Future Internet" in three research fields: middleware and large network infrastructures, novel multimodal interaction concepts, and human protection in ubiquitous computing (privacy, trust, and civil security). He heads or co-supervises various multilateral projects, e.g., on the Internet-of-Services, smart products, ad-hoc and sensor networks, and civil security; these projects are funded by the National Funding Agency DFG, the EU, German ministries, and industry. Max is heading the doctoral school Privacy and Trust for Mobile Users and serves as deputy speaker of the collaborative research center MAKI on the Future Internet and of the center for infrastructure-less emergency response NICER. Max has also led several university wide programs that fostered E-Learning research and application. Max has over 30 years of experience in research and teaching in areas related to Ubiquitous Computing (UC), Networks and Distributed Systems, E-Learning, and Privacy&Trust. He held permanent or visiting professorships at the Universities of Kaiserslautern, Karlsruhe, Linz, Darmstadt, MontrŽal, Sophia Antipolis (Eurecom), and San Diego (UCSD). In 1993, he founded the TeCO institute (www.teco.edu) in Karlsruhe, Germany, which became one of the pace-makers for Ubiquitous Computing research in Europe. Max regularly publishes in Ubiquitous and Distributed Computing, HCI, Multimedia, E-Learning, and Privacy&Trust conferences and journals and authored or co-authored more than 400 publications. He was and is active in numerous conference program committees, as organizer of several annual conferences, and as member of editorial boards or guest editor for journals like Pervasive Computing, ACM Multimedia, Pervasive and Mobile Computing, Web Engineering, and Distance Learning Technology.