Preserving Data Integrity in IoT Networks under Opportunistic Data Manipulation

Shameek Bhattacharjee,
Mehrdad Salimitari, Mainak Chatterjee
University of Central Florida
Orlando, FL
Email: {shameek, mehrdad, mainak}@cs.ucf.edu

Kevin Kwiat
Consultant to Just Born Tech, LLC
Sarasota, FL
Email: kwiatk@sunyit.edu

Charles Kamhoua
Network Security Branch
US Army Research Laboratory
Adelphi, MD
Email: charles.a.kamhoua.civ@mail.mil

Abstract—As Internet of Things (IoT) and Cyber-Physical systems become more ubiquitous and an integral part of our daily lives, it is important that we are able to trust the data aggregate from such systems. However, the interpretation of trustworthiness is contextual and varies according to the risk tolerance attitude of the concerned application and varying levels of uncertainty associated with the evidence upon which trust models act. Hence, the data integrity scoring mechanisms should have provisions to adapt to varying risk attitudes and uncertainties.

In this paper, we propose a Bayesian inference model and a prospect theoretic framework for data integrity scoring that quantify the trustworthiness of data collected from IoT devices by a hub in the presence of an adversary manipulating data and an imperfect anomaly monitoring mechanism. The monitoring mechanism monitors the data being sent from each device and classifies the outcome as not compromised, compromised, and cannot be inferred. These outcomes are conceptualized as a multinomial hypothesis of a Bayesian inference model with three parameters which are then used for calculating a utility value on how reliable the aggregate data is. We use prospect theory inspired approach to quantify this data integrity score and evaluate trustworthiness of the aggregate data from the IoT framework. As decisions are based on how the data is fused, we propose two measuring models- one optimistic and another conservative. The proposed framework is validated using extensive simulation experiments. We show how data integrity scores vary under a variety of system factors like attack intensity and inaccurate detection.

Index Terms—IoT; Prospect theory; Data integrity; Bayesian framework.

I. INTRODUCTION

The proliferation of Internet of Things (IoT) is witnessing an exponential growth, both in terms of market value and number of devices. In situations where multiple devices/networks contend for resources, it is not uncommon that some will deviate from the mutually-agreed upon norms to either i) illegitimately draw additional benefits or ii) mislead a hub from arriving at a fair decision. Thus, there needs to be a mechanism that would establish the trustworthiness of the devices. Usually, trustworthiness is assigned to individual IoT devices based on quality of information they share with others. However, devices may not always be malicious by themselves but their inputs may be compromised (e.g., man in the middle attack, replay etc.) by an adversary before they reach the central decision maker (performing a fused decision). In such a case, focusing on a device’s trustworthiness is not appropriate. Rather, the trustworthiness of the data gathered at the controlling IoT hub is something which is of importance. It may be noted that an IoT hub may have a generic fusion rule and the final dependability is based on both the type of fusion rule used and the integrity of the collected data.

This problem is further exacerbated, when the adversary’s behavior may not always be consistent and is dictated by the context of interaction; for e.g., the adversaries magnitude of attack on a time slot depends on its available power, energy constraints, and the state of other network entities. In other cases, the adversary may behave honestly for a while to gain confidence and then attack later. Hence an adversary’s behavior is often opportunistic, where they rapid switches into different modes render different behavior under different situations known as opportunistic On-Off attacks. In such cases, the regular trust management mechanisms either fail to react quickly or allow quick trust recovery.

In this paper, we propose a Bayesian and prospect theory based framework for data integrity scoring that signifies the trustworthiness of the aggregate data gathered at the IoT hub in presence of an adversary. The adversary can manipulate the data sent to the IoT hub from any IoT device. We assume each IoT device provides a single input on a time slot, all of which are vulnerable to attacks. We assume a generic but imperfect failure monitoring mechanism that produces varying feedback over time. We consider that the outcome of the monitoring mechanism can be classified into three categories: those we know have not been compromised, those we know have been compromised, and those which cannot be inferred either way.

Given these, we compute trustworthiness of the collected data at the IoT hub in making a decision. In this regard, we conceptualize the outcome of monitoring the input over time as a multinomial hypothesis of a Bayesian inference model with three parameters. We build a Bayesian inference-based data integrity scoring model, where we assign a utility value to reflect the reliability of the collected data. Such an integrity scoring model also takes into account the risk a system could afford to tolerate. Therefore, we propose two models of data integrity measuring– first is an optimistic one and the second is

1DISTRIBUTION A. Approved for public release: distribution unlimited.
Case Number: 88ABW-2017-2415.
a conservative one. The optimistic model is for systems where some tolerance for wrong decisions are allowed. However, for a mission critical system where there is almost no room for erroneous decisions, the conservative model could be used. With the probabilities of all the outcomes known, we use prospect theory for computing the utilities. Such an approach helps us model uncertainty in both loss averse and risk averse systems, thus allowing us to differentiate between optimistic and conservative systems.

We conduct extensive simulation experiments and show how data integrity scores vary under a variety of system factors like attack intensity and inaccurate detection. We observe that with more inputs compromised, the data integrity score reduces. Low data integrity scores may also be caused by temporal or initial lack of evidence due to uncertainty.

II. SYSTEM MODEL AND ASSUMPTIONS

We consider a time-slotted system comprising $N$ IoT devices each of which provides only one input (i.e., the vote) on each time slot. The nature of the decision is generic; it could be as simple as a binary voting or it could be some complex decision metric. A centralized IoT hub fuses all votes from each component through a fusion scheme (e.g., majority or plurality voting rule) to arrive at a global decision.

- **Adversarial model:** We assume that all the inputs from each IoT device are exposed to an adversary whose goal is to disrupt the voting process at the central hub. The adversary has some predefined attack resources and can choose to attack different sets of inputs over time and also attack varying number of inputs in each time slot. However, it maintains a long term average of the fraction of the inputs it attacks which we call the probability of attack and denote as $P_a$.

- **Imperfect failure monitoring:** We assume that there is a failure monitoring or anomaly detection mechanism in place that infers whether the input from each device has been compromised or not. Unlike related works [1], [2], we consider that the monitoring mechanism cannot infer an anomaly with certainty. Thus, it classifies the inputs into three categories: i) compromised, ii) not compromised, and iii) undecided. All three are functions of environmental parameters that may be dynamic over time. Also system transients and noisy environments may increase or decrease temporal uncertainty. Hence, the data integrity is computed over time— a larger time window of observation allows a more accurate estimation of the overall data integrity.

- **Uniformly distributed prior inference:** Since there is no bias (or available information) over any of the three possible outcomes of the monitoring process, we assume that the initial probabilities of each is equal. Similarly, we assume that the prior probabilities of an input being compromised or not is also uniformly distributed.

- **Probability of detection:** We define the probability of detection as the percentage of IoTs’ inputs that can be accurately inferred as compromised or not compromised and denote it as $P_{detect}$. Let us further illustrate the meaning of $P_{detect}$ using Fig. 1 that shows an input in reality could be either compromised or not compromised. If compromised, it can be inferred as either as ‘compromised’ with a probability $a_1$ (correct) or ‘undecided’ with probability $a_2$ (uncertain) or ‘not compromised’ with a probability $a_3$ (missed detection). Similarly, if an input was not compromised, it can be inferred as either ‘not compromised’ with a probability $b_1$ (correct) or ‘undecided’ with probability $b_2$ or ‘compromised with a probability $b_3$ (false alarm). Thus, for the two real cases, detection occurs with probabilities $a_1$ and $b_1$. If an input has equal chances of being compromised and not compromised, then $P_{detect} = \frac{a_1 + b_1}{2}$. Else, $a_1$ and $b_1$ will have to be weighted with their corresponding probabilities. For all practical purposes, we consider $P_{detect}$ to be at least 0.5, since it is impractical to have a monitoring mechanism where majority of feedbacks are incorrect. Similarly, $P_{uncertain} = \frac{a_2 + b_2}{2}$ denote the probabilities of ignorance (expressing inherent uncertainty) about IoT inputs. $P_{error} = \frac{a_3 + b_3}{2}$ denotes probability of errors made by the feedback system. These probabilities are used for performance evaluation.

![Fig. 1. Inference possibilities for detection probability](attachment://image)

The above features make the problem of computing the data integrity a probabilistic concept. Hence, we compute the utility value as a incremental process based on observations over time slots. If the adversary uses the same attack strategy, then the utility value will converge sooner. On the other hand, if the adversary changes its attack strategy (i.e., dynamic attack strategy), the utility value will oscillate even for large time windows. Later, we study a special case on how the proposed theoretical model can be modified to accommodate adversaries that do not have a fixed $P_a$ and launch a dynamic (e.g., On-Off) attack strategy.

**On-Off Attack:** The On-Off attack strategy is denoted with an Off:On ratio. In ‘Off’ stage the adversary does not attack. In ‘On’ stage the attacker manipulates a random number of inputs on each time slot. It may be noted that ratios with equal Off to On stages do not depict true inconsistency. Higher ratios like 2:1 and 3:1 are used in this paper. Remember that very high Off:On ratio hardly means the adversary behaves honest most of the time hence may not be realistic.

III. DATA BASED DECISION MAKING

Let the three outputs of the anomaly monitoring mechanism, viz. ‘not compromised’, ‘compromised’, or ‘undecided’ be denoted by $\alpha$, $\beta$ and $\mu$ respectively. Let $n_\alpha$ represent the number of device inputs that have ‘not’ been compromised, $n_\beta$ be the number of compromised ones, and $n_\mu$ be the number for which a decision could not be arrived. Of course, $n_\alpha + n_\beta + n_\mu = N$. Since the values of $n_\alpha$, $n_\beta$ and $n_\mu$ change
over time, we represent these observations at time $t$ as $n_{\alpha}(t)$, $n_{\beta}(t)$ and $n_{\mu}(t)$.

Given that the underlying parameters of the system supplying accurate data are unknown, we use a Bayesian inference approach to incrementally update the corresponding probability estimate for a hypothesis that the data aggregate is correct with a certain probability. The system is only as reliable as the individual inputs are. Therefore, we have to calculate the posterior probabilities associated with encountering each of the three feedbacks. The final data integrity score will be some function of these posterior probabilities which are also known as belief estimate in Bayesian inference.

To begin with, an uniform belief over the three possibilities is assumed as there is no initial information. As time progresses, we update the belief estimate based on the observed values of $\alpha$, $\beta$, and $\mu$ which increases the accuracy of the estimate of the belief associated with each category.

We define $\theta_{\alpha}$, $\theta_{\beta}$, and $\theta_{\mu}$ as the probabilities for an input being ‘not compromised’, ‘compromised’, and ‘undecided’ respectively. Of course, $\theta_{\alpha} + \theta_{\beta} + \theta_{\mu} = 1$, since the outcomes are exhaustive and mutually exclusive. We define $X(\hat{\theta})$ as the hypothesis described by these underlying unknown Bayesian probability parameters where $\hat{\theta} = \{\theta_{\alpha}, \theta_{\beta}, \theta_{\mu}\}$.

Let $D_{\alpha}$, $D_{\beta}$, and $D_{\mu}$ represent the random variables that represent the number of times the outcomes $\alpha$, $\beta$ and $\mu$ occur. The observation data can be represented as random observation vector $D(N) = \{D_{\alpha}, D_{\beta}, D_{\mu}\}$ having a multinomial distribution also known as concentration hyperparameter of the underlying 3-tuple probability parameter described by $\theta_{\alpha}$, $\theta_{\beta}$, and $\theta_{\mu}$.

A. Bayesian Inference

As mentioned earlier, there are $N$ independently monitored components of a system whose parameters for voting behavior are unknown due to changing adversarial attack strategies and the imperfect monitoring mechanism. Given this, we calculate the Bayesian belief associated with ‘not compromised’. Similarly, we will model Bayesian posterior belief for the other two cases as well viz. compromised and undecided.

We use the observation counts from the sequential observations over time to calculate the posterior Bayesian estimate of each of the parameters. Our objective is to estimate and update the probability parameters in $X(\hat{\theta})$, viz. $\theta_{\alpha}$, $\theta_{\beta}$, and $\theta_{\mu}$ based on observation evidence $D(N)$ and prior information on the hypothesis parameter, $\hat{\theta}$, itself.

Since there is no information about $\hat{\theta}$ initially, we consider the prior parameters of $\hat{\theta}$ to be uniformly distributed. Subsequent observations decide how these parameters are updated. Our first step is to calculate the Bayesian estimate of $\hat{\theta}$.

First, we show the case of estimating belief that a ‘not compromised’ occurs ($\theta_{\alpha}$). Since in Bayesian inference, the assumption is that prior and posterior probability have the same distribution, we can formally define the probability parameters as:

\[
P(X(\hat{\theta}) = \alpha | \hat{\theta}) = \theta_{\alpha} = P(X(\hat{\theta}) = \beta | \hat{\theta}) = \theta_{\beta} = P(X(\hat{\theta}) = \mu | \hat{\theta}) = \theta_{\mu}
\]

This assumption is due to the well known fact that a Dirichlet distribution acts as a conjugate prior to multinomial distributions [3]. Hence prior and posterior preserve the same form.

The observations data $D(N)$ can be treated as a multinomial distribution with probability parameter $\theta_{\alpha}$, $\theta_{\beta}$, and $\theta_{\mu}$, where the probability mass function is given by:

\[
P(D_{\alpha} = n_{\alpha}, D_{\beta} = n_{\beta}, D_{\mu} = n_{\mu} | \hat{\theta}) = \frac{N!}{n_{\alpha}! n_{\beta}! n_{\mu}!} \theta_{\alpha}^{n_{\alpha}} \theta_{\beta}^{n_{\beta}} \theta_{\mu}^{n_{\mu}}
\]

(2)

Given this we can use Bayes theorem to calculate the posterior belief estimate on the event of a positive interaction $X(\hat{\theta}) = \alpha$, given observation data $D(N)$ as:

\[
P(X(\hat{\theta}) = \alpha | D(N)) = \frac{P(X(\hat{\theta}) = \alpha, D(N))}{P(D(N))}
\]

(3)

The denominator of the above equation is the marginal probability that can be conditioned or marginalized on all possible outcomes for $\hat{\theta}$ and since probabilities are continuous

\[
P(D(N)) = \int P(D(N)|\hat{\theta})f(\hat{\theta})d(\hat{\theta})
\]

(4)

Since there is no prior information on $\hat{\theta}$ (before any observations) in Eqn. (4), we can assume it to be uniformly distributed such that $f(\hat{\theta}) = 1$ and we can put Eqn. (2) in Eqn. (4), and get

\[
P(D(N)) = \frac{N!}{(N+2)!}
\]

(5)

Assuming conditional independence between the $X(\hat{\theta})$, $D(N)$ and $\hat{\theta}$, we calculate the numerator of Eqn. (3), $P(X(\hat{\theta}) = \alpha, D(N))$, as:

\[
P(X(\hat{\theta}) = \alpha, D(N)) = \frac{N!(n_{\alpha} + 1)}{(N+3)!}
\]

(6)

Thus, Eqn. (3), can be solved by dividing Eqn. (7) by Eqn. (6), which gives

\[
P(X(\hat{\theta}) = \alpha | D(N)) = \frac{n_{\alpha} + 1}{N+3}
\]

(8)

Similarly, $P(X(\hat{\theta}) = \beta | D(N)) = \frac{n_{\beta} + 1}{N+3}$ and $P(X(\hat{\theta}) = \mu | D(N)) = \frac{n_{\mu} + 1}{N+3}$. These equations are the expressions for posterior belief of ‘not compromised’, ‘compromised’, and ‘undecided’. To simplify the notations of belief estimates of the three categories, we rewrite them as $R_{\alpha}$, $R_{\beta}$, $R_{\mu}$ respectively. Of course, it can be verified that $R_{\alpha} + R_{\beta} + R_{\mu} = 1.$
IV. DATA INTEGRITY UNDER UNIFORM ATTACKS

In this section, we propose two system models under different conditions. Then, we propound a data integrity measurement approach based on prospect theory.

A. Optimistic System Model

As was discussed, there are three possible outcomes for each of the components of IoT framework in each time slot: compromised, not compromised, and undecided. Each of these outcomes will incur some costs. As it is evident, the compromised components will effectuate the highest cost, denoted by $c_c$. Not compromised devices will also cause some cost denoted by $c_n$. The third cost, denoted by $c_u$, is associated with the devices that remained undecided. Based on the system requirement, we will take different measures for undecided components. The general relation between these costs is:

$$c_n < c_u \leq c_c$$ (9)

For optimistic systems, we consider half of the undecided components as compromised because we assumed that adversary has uniformly chosen the IoT inputs to attack i.e., there is no reason for preferential attack on a certain IoT device’s input. In this case $c_u$ is defined as follows:

$$c_u = \frac{c_c + c_n}{2}$$ (10)

Of course, when the proportion of undecided is high, we may not be as confident on the integrity measurement than when we have fewer undecided.

B. Conservative System Model

Unlike the optimistic approach, where the undecided ones are split in an equal ratio, the conservative model treats the undecided ones as if they are more likely to be compromised. In this case, we consider two weights for the compromised and not compromised costs namely $w_1$ and $w_2$ in a way that $w_1 + w_2 = 1$ and $0.5 < w_2 \leq 1$. Hence,

$$c_u = w_1c_n + w_2c_c$$ (11)

This conservative way of computing the undecided cost is more appropriate for mission-critical systems where the decisions can mostly be made based on the ‘not compromised’ inputs. Depending on how conservative a system is, we define the weight $w_2$. By increasing $w_2$, the chance of assuming a compromised device as a not compromised one reduces. If the system is highly mission-critical and there is no room for risk, we consider $w_2 = 1$. In this case, all undecided devices are considered as compromised even if there could be some that were not compromised.

C. Data Integrity Measurement Using Prospect Theory

Using the probability values, posterior belief of the three possible outcomes, obtained in section IV, we want to calculate a utility value based on prospect theory. This utility value is obtained as follows:

$$Utility = \sum_{i=1}^{3} V(\delta_i)W(R_i)$$

$$= V(\delta_0)W(R_0) + V(\delta_\beta)W(R_\beta) + V(\delta_\mu)W(R_\mu)$$ (12)

where $V$ denotes value function and $W$ denotes weighting function. $\delta_0$, $\delta_\beta$, and $\delta_\mu$ are three deviation values related to the three independent outcomes of extracted data from each device in an IoT network. These deviations show the difference between profit function, denoted by $\pi$, and reference point denoted by $\pi_p$. Due to the independency of the outcomes, three different profit functions are defined for each and are denoted by $\pi_\alpha$, $\pi_\beta$, and $\pi_\mu$. The deviation values and profit functions are defined as follows:

$$\begin{align*}
\delta_\alpha &= \pi_\alpha - \pi_p; & \pi_\alpha &= nc - n_\alpha c_n \\
\delta_\beta &= \pi_\beta - \pi_p; & \pi_\beta &= nc - n_\beta c_c \\
\delta_\mu &= \pi_\mu - \pi_p; & \pi_\mu &= nc - n_\mu c_u
\end{align*}$$ (13)

We need to define another variable called the ‘reference point’ to calculate deviation values. $\pi_p$ is defined by assuming all the IoT devices as not compromised:

$$\pi_p = n(c - c_n)$$ (14)

According to Eqns. (13) and (14), the state which is considered for all the $n$ nodes at the reference point will determine whether the real outcomes are gains or losses. By considering all the $n$ nodes as not compromised at the reference point, any node which is not compromised will always be considered as a gain and other outcomes will be counted as a gain or loss based on the cost values. However, if we had considered the nodes as compromised or undecided at the reference point, the value of $\delta$ in each state would increase such that even compromised devices yield a gain. The reason for the gain is that the not compromised devices have the lowest cost.

As for the value function, it is an asymmetrical S-shaped function as shown in Fig. 2. It is asymmetric because of its loss aversion nature which causes the same absolute values to have more impact for the loss than the impact on gain. Its value is dependent on the deviation of the profit values from the reference point, defined in Eqn. (13). Value function is obtained as follows:

$$V(\delta) = \begin{cases} 
\delta\gamma & \text{if } \delta \geq 0; \\
-\lambda(-\delta)^\gamma & \text{if } \delta < 0
\end{cases}$$ (15)

![Fig. 2. Examples of value function with different $\lambda$ and $\gamma$](image)

Positive part of the value function represents the gain and its negative section denotes the loss in reliability of the integrated data from the IoT devices. $\lambda$ and $\gamma$ are two parameters used for controlling loss aversion and risk aversion where $\lambda > 1$, and $0 < \gamma < 1$ [4]. By increasing $\lambda$, the IoT system will become more loss averse and consequently will become more asymmetric with the loss part becoming more convex. By
decreasing parameter $\gamma$, the IoT system will become more risk averse. The effect of these parameters in value function is shown in Fig. 2. Choosing the right values for these two parameters depends on the IoT system. As the system becomes more conservative, it becomes more loss averse.

According to Eqn. (12), we need to define another function called weighting function. Based on prospect theory, in real life decision making process, people overreact to lower probabilities and under-react to higher probabilities [5]. Here, we have the same situation. For example, if one device is compromised, it will have a significant impact in reliability of the aggregated data. However, if we have 30 devices and 20 of them are compromised, having one more compromised device will not bring about a significant difference. Furthermore, the effect of probability weights is not the same for loss and gain [6]. As it is obvious, it is desirable to have the minimum possible loss rather than achieving a huge gain since even a little loss will result in losing confidence in the aggregated data. Therefore, two similar weighting functions but with two different parameters, denoted by $\omega$ and $\rho$, are defined for gain denoted by $W^+(p)$ and loss denoted by $W^-(p)$ as follows:

$$
\begin{align*}
W^+(p) &= \frac{p^\omega}{[p^\rho + (1-p)^\rho]^\frac{1}{\rho}} & 0.5 \leq \rho < 1 \\
W^-(p) &= \frac{1-p^\omega}{[p^\rho + (1-p)^\rho]^\frac{1}{\rho}} & 0.5 \leq \omega < 1
\end{align*}
$$

\tag{16}

$\rho$ and $\omega$ are defined in a way to emphasize loss which means choosing lower values for $\omega$ in comparison with $\rho$. Their values also depend on the system that we are dealing with. In conservative systems, the effect of weighting function for loss will be higher than optimistic systems. Therefore, conservative systems have lower values of $\omega$ in order to cause strict penalty for loss. On the other hand, the value of $\rho$ for conservative systems is higher than the optimistic ones since achieving gain in conservative systems is not as highly valued as in optimistic systems. In other words, achieving gain in a conservative system will not achieve a positive utility as high as its optimistic counterpart. The effects of these two parameters are demonstrated in Fig. 3.

![Fig. 3. Effect of $\rho$ and $\omega$ on weighting function. $P$ corresponds to the posterior belief of the 3 possible outcomes](image1)

Using these definitions, we obtain a utility value for an IoT system in each time slot. This utility value indicates the aggregated data from all the nodes in this IoT framework is reliable to what extent. Each IoT network will have its own parameters. Based on these parameters, a threshold is defined for the utility value of each IoT system. If the calculated utility of this system is higher than this threshold, the aggregated data from this system has acceptable integrity.

V. Asymmetric Trust Update for On-Off Attacks

In On-Off attacks, the adversaries have preferences over time periods. In such a case, both CWMA or EWMA would not reflect true behavior of the node. The equal weighted CWMA will lag in reflecting such attacks, while EWMA will enable the system to quickly recover or redeem its reputation when it switches back to honest behavior. Under On-Off attacks, the data integrity scoring framework should not allow the system to recover its integrity score even though the adversary starts behaving well after a short burst of attack. In Fig. 4, we show an example where the adversary employs a 2:1 Off-On ratio where it divides the time domain of 300 slots in four stages: ‘Stage 1’ ranging from $t = 0 − 100$, ‘Stage 2’ ranging from $t = 101 − 150$, ‘Stage 3’ ranging from $t = 151 − 250$, ‘Stage 4’ ranging from $t = 251 − 300$. Finally ‘Stage 5’ ranging from $t = 301 − 500$ is a no attack phase to analyze the after effects of On-Off attacks. In Stage 1, the adversary does not attack in a bid to gain a high trust of the system initially. In Stage 2, it attacks for the next 50 time slots with a random magnitude on each of the time slots. In Stage 3, it does not attack on any of the 100 slots. In Stage 4, it again attacks on 50 slots with a random attack magnitude. In Stage 5, it again behaves cooperatively. Suppose an algorithm checks whether the system is compromised or not every 50 slots and uses a threshold of zero below which the collective system’s data is deemed ‘unusable’ for decision making. On 100-th, 200-th, 250-th, 350-th, 400-th, 450-th and 500-th slot, the system’s data is deemed usable by both trust/score update schemes although adversary is employing a stealthy On-Off attack. We see that CWMA reacts too slowly and fails to reflect malicious nature even at the end of the Stage 2. On the other hand, EWMA detects attacks quickly but also allows such nodes to quickly recover their reputation on 151-th and 301-th slot in the ensuing Off period. This motivates the need for a better data integrity scoring update scheme.

![Fig. 4. Shortcomings of CWMA and EWMA](image2)

A. Weighted Integrity Score

Let us denote the utility of data integrity as obtained from Eqn. 12 as $u_{di}$. Since the interval of $u_{di}$ is large, the depiction of trust values and bounded classification decisions as dependable or not get difficult. Hence, we map these scores to a bounded lower dimensional plane via a scaling trick, which ensures all negative $u_{di}$ values are mapped between
[-1.0] and positive \( u_{\text{di}} \) values are mapped between [0,+1]. Also, the weights monotonically increase with increasing data integrity and vice-versa. Therefore, we report the normalized weight \( w_{\text{di}} \) by giving a value between [-1,1] using Eqn. (17).

The normalized weighted integrity score is given by:

\[
\begin{align*}
    w_{\text{di}} &= \begin{cases} 
    1 - e^{-|u_{\text{di}}|} & \text{if } u_{\text{di}} > 0; \\
    -(1 - e^{-|u_{\text{di}}|}) & \text{if } u_{\text{di}} < 0; \\
    0 & \text{if } u_{\text{di}} = 0
    \end{cases}
\end{align*}
\]

(17)

The above equation which uses trust to give weights helps to clearly distinguish between two classes of nodes. At any time \( t \), the weighted integrity score is denoted as \( w_{\text{di}}(t) \).

### B. Asymmetric Weighted Moving Average scheme

We propose an Asymmetric Weighted Moving Average (AWMA) technique that is based on the socially inspired concept that bad actions are far more remembered than good actions. This forms the basis of the asymmetric weighted moving average scheme, where slots with instantaneous integrity trust \( w_{\text{di}}(t) \) lower than a threshold \( \Gamma_{\text{on-off}} \) are given more weight than time slots where \( w_{\text{di}}(t) \) has higher values. The value of \( \Gamma_{\text{on-off}} \) is dictated by a system specific risk attitude and defines what can be termed as sufficiently good behavior. In the update of trust values, there are two important things; the cumulative average and current trust value. We introduce four weighting factors \( \chi_a, \chi_{b_{\text{max}}}, \chi_{c_{\text{min}}} \) and \( \chi_d \) such that \( 0 < \chi_a < 1; 0 << \chi_{b_{\text{max}}}; 0 < \chi_{c_{\text{min}}}, 0 < \chi_d < 1 \). Note the fact that \( \chi_{c_{\text{min}}} \) is much less than \( \chi_{b_{\text{max}}} \) introduces an asymmetry. Now there may be four possible scenarios at time \( t \) with regard to On-Off attacks.

**Case (a):** \( w_{\text{di}}^{\text{avg}}(t-1) > \Gamma_{\text{off}} \) and \( w_{\text{di}}(t) > \Gamma_{\text{on-off}} \)

**Case (b):** \( w_{\text{di}}^{\text{avg}}(t-1) > \Gamma_{\text{off}} \) and \( w_{\text{di}}(t) \leq \Gamma_{\text{on-off}} \)

**Case (c):** \( w_{\text{di}}^{\text{avg}}(t-1) \leq \Gamma_{\text{off}} \) and \( w_{\text{di}}(t) > \Gamma_{\text{on-off}} \)

**Case (d):** \( w_{\text{di}}^{\text{avg}}(t-1) \leq \Gamma_{\text{off}} \) and \( w_{\text{di}}(t) \leq \Gamma_{\text{on-off}} \)

In Case (a), a cumulative average higher than \( \Gamma_{\text{on-off}} \) suggests a system is maintaining a sufficiently good behavior. If the current trust value is also higher than \( \Gamma_{\text{on-off}} \) then it suggests continuity of the good behavior. Hence continuing good behavior is rewarded with a high weighting factor \( \chi_a \) to \( w_{\text{di}}(t) \) and low weightage given to \( w_{\text{di}}^{\text{avg}}(t-1) \) using \( 1 - \chi_a \). We name \( \chi_a \) as a rewarding factor such that \( 1 > \chi_a > 0 \). It helps a historically reliable system to improve, or at-least maintain its reputation, if it also behaved in a cooperative manner in time slot \( t \). Hence for Case (a), cumulative trust is updated as:

\[
    w_{\text{di}}^{\text{avg}}(t) = (1 - \chi_a) \times w_{\text{di}}^{\text{avg}}(t-1) + \chi_a \times w_{\text{di}}(t)
\]

In Case (b), a cumulative average higher than \( \Gamma_{\text{on-off}} \) and \( w_{\text{di}}(t) \leq \Gamma_{\text{on-off}} \) suggests a system maintaining a sufficiently good behavior up to time \( t-1 \) and then initiated some anomalous behavior. Hence all the good behavior until now needs to be forgotten and a very high weight be given to the current slot’s anomalous behavior. This will cause the system’s cumulative trust value to quickly decrease. Once this happens, Case (c) would ensure that the cumulative trust is not able to redeem itself quickly. Hence \( w_{\text{di}}(t) \) is weighted with a high value \( \chi_{b_{\text{max}}} \) such that \( 1 > \chi_{b_{\text{max}}} >> 0 \) and \( w_{\text{di}}^{\text{avg}}(t-1) \) is weighted using \( 1 - \chi_{b_{\text{max}}} \). We name \( \chi_{b_{\text{max}}} \) as a punishment factor. The higher the value of punishment factor the quicker the drop in the reputation and hence the more severe the system’s reaction will be to new evidence of malicious behavior. In such cases, the cumulative trust is updated as:

\[
    w_{\text{di}}^{\text{avg}}(t) = (1 - \chi_{b_{\text{max}}}) \times w_{\text{di}}^{\text{avg}}(t-1) + \chi_{b_{\text{max}}} \times w_{\text{di}}(t)
\]

In Case (c), a cumulative average lower than \( \Gamma_{\text{on-off}} \) but a current trust value \( w_{\text{di}}(t) \) higher than \( \Gamma_{\text{on-off}} \) signifies a system where current inputs are cooperative but has a history of anomalous behavior which may be as recent as \( t-1 \). Hence even though \( w_{\text{di}}(t) \) may be high we assign it a very low weight \( \chi_{c_{\text{min}}} \) such that \( 0 < \chi_{c_{\text{min}}} << 1 \) and assign \( 1 - \chi_{c_{\text{min}}} \) to \( w_{\text{di}}^{\text{avg}}(t-1) \). We name \( \chi_{c_{\text{min}}} \) as the redemption factor that controls how fast or slow a system with malicious history can redeem its trustworthiness if it shows good behavior for a sufficiently long time. Redemption factors also make it possible for systems which experienced noise redeem their trust values. A low redemption factor ensures that the trust value is not increased quickly even though a system starts to behave honestly after a period of malicious behavior. In this case cumulative trust is updated as:

\[
    w_{\text{di}}^{\text{avg}}(t) = (1 - \chi_{c_{\text{min}}}) \times w_{\text{di}}^{\text{avg}}(t-1) + \chi_{c_{\text{min}}} \times w_{\text{di}}(t)
\]

In Case (d), both cumulative average and current trust value of node \( j \) are below \( \Gamma_{\text{on-off}} \) indicating continuing anomalous behavior. In such a case, we provide \( \chi_d \) known as retrogression factor as weight to the current value and \( 1 - \chi_d \) weight to cumulative average such that trust is updated as:

\[
    w_{\text{di}}^{\text{avg}}(t) = (1 - \chi_d) \times w_{\text{di}}^{\text{avg}}(t-1) + \chi_d \times w_{\text{di}}(t)
\]

The above scheme, termed as asymmetric weighted moving average, is effective in defending against On-Off attacks which is not possible using equally weighted or exponential weighted moving averages. In the simulation section, we also show that this can also be effective to distinguish malicious IoT devices and devices experiencing intermittent noise.

### VI. Simulation Model and Results

We simulate a generic system with 100 IoT devices. Inputs from all devices are monitored by an imperfect monitoring mechanism that produces three possible outcomes. The probability of detection and attack are varied to capture their effects on data integrity measurements.

Under non-opportunistic attacks, an adversary attacks and compromises different sets of inputs over time. The number of inputs compromised vary over each time slot; although the long-term average of the number of inputs compromised, denoted by \( P_a \), remains the same. Under opportunistic On-Off attacks, we study how a system can establish appropriate trustworthiness under opportunistic time dependent attacks.
We study the data integrity utility values for different values of $P_a$ and $P_{detect}$; and plot instantaneous and moving average of data integrity scores. For calculating the utility values during all the simulations, the parameters are considered as follows: $\lambda = 2$, $\gamma = 0.5$, $\omega = 0.63$, and $\rho = 0.69$.

### A. Optimistic and Conservative Utility Values Under Same Attack Conditions: Instantaneous and Average

In Fig. 5, we plot the instantaneous and steady state utility values for both optimistic and conservative models when the adversary launches attacks with $P_a = 0.1$ and the system is able to detect aggregated data with $P_{detect} = 0.9$.

The optimistic system is defined with these costs: $c_{e} = 0.1$, $c_{n} = 0.01$, and $c_{u} = 0.055$. The costs for conservative system are defined as follows: $c_{e} = 0.1$, $c_{n} = 0.01$, and $c_{u} = 0.09$.

We observe that the instantaneous utility values fluctuate over time owing to the particular realizations of $P_a$ in a time slot and imperfect monitoring based on $P_{detect}$. As expected, with sufficient observations, the moving average of optimistic utility values converges to a positive steady state value around 0.5. Furthermore, the moving average of conservative model utility values converges to a steady state value lower than the corresponding values related to optimistic system. This value is almost zero. Therefore, under the same attack and detection conditions, the collected data from an optimistic system is usually reliable but the aggregate data from a conservative system is mostly unreliable.

![Fig. 5. Instantaneous and Moving Avg. utility of optimistic and conservative systems under same $P_a = 0.1$ and $P_{detect} = 0.9$](image)

### B. Utility Value and Attack Magnitude

In Fig. 6, we plot steady state utility values in an optimistic system for different attack magnitudes, from $P_a = 0.1$ to $P_a = 1$, under the same monitoring mechanism, $P_{detect} = 0.9$. In Fig. 7, we have illustrated the average utility values for different attack magnitudes, $P_a$, under two different monitoring mechanism for both an optimistic system and a conservative system. $P_{detect}$ is 0.9 in Fig. 7(a), and it is 0.5 in Fig. 7(b).

According to Fig. 6 and Fig. 7, aggregate data in an optimistic system with a low $P_a$ is reliable most of the time. However, if the attack magnitude increases, the collected data becomes mostly unreliable. In contrast, the aggregate data in a conservative system is mostly unreliable even with low magnitude of attack. However, with an increase in magnitude of attack in a conservative system, it becomes more unreliable.

Fig. 7 also reveals that with a reduction in $P_{detect}$, the effect of attack magnitude on utility values follows the same trend with just a small decrease in utility values. The effect of attack magnitude on utility value is almost linear. Furthermore, as $P_{detect}$ decreases, the difference between average utility values of an optimistic system and a conservative system increases as $c_{e}$ is same for both systems. The only difference is in $c_{u}$ which is dependent on undecided devices.

![Fig. 6. Avg. utility values of optimistic system over time under different $P_a$](image)

![Fig. 7. Average utility values of optimistic and conservative systems over time vs attack magnitude for different values of $P_{detect}$](image)

### C. Utility Value and Imperfect Monitoring

Imperfect monitoring can also have negative effect on utility values and subsequently on reliability of collected data from an IoT system as much as attack magnitude. Fig. 8 shows the effect of imperfect monitoring for an optimistic system over time while the attack magnitude remains the same, $P_a = 0.1$. We observe that by an increase in $P_{detect}$, the aggregate data become more reliable with a linear trend.

![Fig. 8. Avg. utility values of optimistic system over time with increasing $P_{detect}$](image)

### D. Defending against On-Off attacks: A special case

We consider an adversary launching On-Off attacks in the five stages over 500 slot times as was discussed in Section V. The first 300 slots are an active attack period and 300-500 is used to study how trust recovers. For most results we consider a 2:1 ratio. Later we compare results with 2:1 and 3:1 On-Off
ratio. We plot the results of On-Off attacks as calculated by the IoT hub using equations from the asymmetric weighted moving average discussed in Section V. We compare the results with other popular trust update schemes and justify the suitability of asymmetric averaging to On-Off attacks.

1) Choice of weighing factors and threshold: The weighing factors \( \chi_a, \chi_{b_{\text{max}}}, \chi_{c_{\text{min}}}, \) and \( \chi_d \) are chosen as 0.99, 0.999, 0.001 and 0.001. We can verify that this satisfies the conditions: \( 0 < \chi_{c_{\text{min}}} \ll \chi_{b_{\text{max}}} < 1, \) \( 0 < \chi_a < 1, \) and \( 0 < \chi_d < 1. \) The skewed values of the weighing factors \( \chi_{c_{\text{min}}} \) and \( \chi_{b_{\text{max}}} \), justify the asymmetry provided by giving negative behaviors a very high weightage and positive behavior a very low weightage on the first occurrence of a negative behavior. The choice of \( \chi_a \) and \( \chi_d \) can be used to control the rate of trust redemption. If a system requires slower trust redemption then lower values of \( \chi_a \) and \( \chi_d \) are necessary. We put these weighing factors in the four case based equations discussed in Section V. Since there is no fixed magnitude of attack we keep the mid point between the trust value range \((-1, +1)\) as \( \Gamma_{\text{on-off}} = 0. \) However, \( \Gamma_{\text{on-off}} \) can be adjusted according to the requirements of the system. More conservative systems will have \( \Gamma_{\text{on-off}} > 0. \) Different values of \( \chi_{\text{min}} \) and \( \chi_{\text{max}} \) can be chosen to ensure more fairness to nodes in a network inherently susceptible to more bit flips due to noise.

2) Comparison with Equal Weighted CWMA: In Fig. 9, we show how AWMA performs as opposed to the CWMA. We observe that at Stage 1 with no attacks, both schemes preserve a high trust value, but when attacks start from the 101st time slot for the next 50 slots, AWMA ensures that the cumulative dependability score is decreased more rapidly and preserves a low value. On the other hand, CWMA is slow to react due to the adversary having behaved well in the first 100 slots. This happens because once the current value in a slot is less than 0, the proposed AWMA model forgets the previous high reputation through a very low value \( 1 - \chi_{b_{\text{max}}} = 0.001 \) and expresses extremely high weight \( \chi_{b_{\text{max}}} = 0.999 \) to the current values from the 101th time slot, thus causing the cumulative trust at stage 2 to decrease rapidly. Even at the end of Stage 5, when attacks have ceased for the last 200 slots, we see that the dependability score given by the asymmetric average is low enough to reflect the adversary’s malicious attacks, while the equal weighted moving average fails to capture this outcome because the off-on attack ratio is 2:1, i.e., more slots with no attacks. This happens because previous cumulative trust of less than 0 (selected \( \Gamma_{\text{on-off}} \)) at the end of Stage 2 is given a very high weight compared to current honest behavior. It prevents the trust values to improve even during honest behavior.

3) Comparison between higher and lower On-Off ratios: A 3:1 On-Off attack ratio is less aggressive than 2:1. Hence, after 500 slots, we should expect 3:1 to have higher dependability score. It may be noted a ratio as high as 1:1 is not a characteristic of On-Off attack and too low attack ratio hardly effects the system. In Fig. 10, we observe the differences in the dependability scores under 2:1 and 3:1 attack ratios.

VII. CONCLUSIONS

In this paper, we proposed a Bayesian framework to maintain data integrity in an IoT system under opportunistic data manipulation by an adversary. By considering an imperfect monitoring mechanism, we quantified the trustworthiness of the data being collected by an IoT hub using utility values obtained by prospect theory based on a multinomial hypothesis. An asymmetric weighted moving average scheme is also proposed that can counter stealthy On-Off attacks. The proposed framework has been validated using extensive simulations and the results bring out the efficacy of the framework.

REFERENCES