Behavior-Rule Based Intrusion Detection Systems for Safety Critical Smart Grid Applications

Robert Mitchell and Ing-Ray Chen, Member, IEEE

Abstract—In this paper, a behavior-rule based intrusion detection system (BRIDS) is proposed for securing head-ends (HEs), distribution access points/data aggregation points (DAPs) and subscriber energy meters (SEMs) of a modern electrical grid in which continuity of operation is of the utmost importance. The impact of attacker behaviors on the effectiveness of a behavior-rule intrusion detection design is investigated. Using HEs, DAPs and SEMs as examples, it is demonstrated that a behavior-rule based intrusion detection technique can effectively trade false positives for a high detection probability to cope with sophisticated and hidden attackers to support ultra safe and secure applications. It is shown that BRIDS outperforms contemporary anomaly-based IDSs via comparative analysis.

Index Terms—Cyber physical systems, data aggregation point, distribution access point, head-end, intrusion detection, safety, security, subscriber energy meter.

I. INTRODUCTION

The most prominent characteristic of a smart grid such as a modern electrical grid or electricity infrastructure is the feedback loop that acts on the physical environment. In other words, the physical environment provides data to the actuators which change the physical environment. Modern electricity infrastructure is often characterized by sophisticated reliability, efficiency, sustainability and utility control units interacting with the physical environment including subscriber appliances. This paper concerns intrusion detection mechanisms for detecting compromised devices embedded in WANs, NANs and HANs for supporting safe and secure applications that subscribers can depend on with confidence.

Intrusion detection system (IDS) techniques for this domain are still in their infancy with very little work reported in the literature. Only [2], [3], [6], [10], [13], [14], [16], [19]–[23] reported related intrusion detection. However, none of these had no numerical data regarding the false negative probability \( p_{fn} \) (i.e., missing a bad node) and the false positive probability \( p_{fp} \) (i.e., misidentifying a good node as a bad node). The other three had minimal numerical data: one or two data points characterizing \( p_{fn}/p_{fp} \) instead of a dataset that could be transformed into a Receiver Operating Characteristic (ROC) plot, i.e., a \( p_{fn} \) versus \( p_{fp} \) curve that describes the relationship between \( p_{fn} \) and \( p_{fp} \) obtained as a result of applying IDS techniques.

Specifically, Zhang et al. [22], [23] studied two detection algorithms called CLONALG and AIRS2Parallel. CLONALG is unsupervised. AIRS2Parallel is semi-supervised. They reported that CLONALG had a detection accuracy between 80.1% and 99.7% and AIRS2Parallel had an accuracy between 82.1% and 98.7%, where the detection accuracy is the likelihood that a node is classified correctly, calculated by \( 1 - p_{fp} - p_{fn} \). He and Blum [10] investigated a series of anomaly-based IDSs including Locally Optimum Unknown Direction (LOUD), Locally Optimum Estimated Direction (LOED), LOUD-Generalized Likelihood Ratio (LOUD-GLR) and LOED-Generalized Likelihood Ratio (LOED-GLR). He and Blum’s LOUD-GLR approach performed the best: The maximum detection rate (i.e., \( 1 - p_{fn} \)) is reportedly 95%. However, no ROC data were given in [10], [22], [23].

Intrusion detection techniques in general can be classified into three types: signature-based, anomaly-based and specification-based techniques. In this paper, specification-based detection is considered rather than signature-based detection to deal with unknown attacker patterns. Specification-based techniques are considered rather than anomaly-based ones (such as those by Zhang et al. [22], [23] and He and Blum [10]) to avoid using resource constrained sensors or actuators in a WAN for profiling anomaly patterns (for example, through learning) and to avoid high false positives (treating good nodes as bad nodes).

To accommodate resource constrained devices, this paper develops the design notion of behavior rules for specifying acceptable behaviors of physical devices in a WAN, NAN or HAN. Rule-based intrusion detection thus far has been applied only in the context of communication networks which have no concern of physical environments and the closed-loop control structure as in a head-end (HE), distribution access point/data aggregation point (DAP) or subscriber energy meter (SEM).

In the literature, specification-based IDS techniques have been proposed for intrusion detection of communication protocol misbehaving patterns [7]–[9], [12]. Da Silva et al. [8] propose an IDS that applies seven types of traffic-based rules to detect intruders: interval, retransmission, integrity, delay, repetition, radio transmission range and jamming. Ioannis et al. [12] propose a multitrust IDS with traffic-based collection that audits the forwarding behavior of suspects to detect blackhole and greyhole attacks launched by captured devices based on the the rate (versus the count) of specification violations. [7], [9] also only considered specification-based state...
machines for intrusion detection of misbehaving patterns in communication networks. The specification-based technique in this paper distinguishes itself from [7]–[9], [12] cited above by addressing the unique requirements of the domain. First, modern electricity infrastructure has control loops that tie the physical environment to the CPS. Second, components are stationary which eliminates IDSs based on instantaneous motion or movement profiles. Third, they are federated systems; bulk power generators, energy markets, transmission providers, distribution providers and subscribers own, host and operate different segments of the CPS. Fourth, their scale is substantial; for example, the count of SEMs could be in the millions. Fifth, these CPSs are heterogeneous. In this work, specification-based behavior rules are derived from control loops which tie the intrusion detection to the critical business rules of the CPS while not relying on motion or track data used in other approaches. Also, the goals of each interest in the CPS are specifically considered. Further, a method to transform behavior rules to a state machine is proposed, so that a device based IDSs [10], [22], [23] via comparative analysis.

The contribution of our work relative to prior work cited is that behavior rules for WAN, NAN and HAN devices controlling actuators and sensors embedded in the physical environment are specifically considered. Further, a method to transform behavior rules to a state machine is proposed, so that a device that is being monitored for its behavior can be checked against the transformed state machine for deviation from its behavior specification. Untreated in the literature [17], in this paper the impact of attacker behaviors on the effectiveness of intrusion detection in the production, transmission, distribution and consumption segments is also investigated. Using HEs, DAPs and SEMs as examples, it is demonstrated that an intrusion detection technique can effectively trade false positives for a high detection probability to cope with more sophisticated and hidden attackers to support ultra safe and secure applications. More-
reckless attacker performs attacks whenever it has a chance. The main objective is to impair the functionality at the earliest possible time. A random attacker, on the other hand, performs attacks only randomly to avoid detection. It is thus insidious and deceptive with the objective to cripple the functionality. The attacker behavior is modeled by a random attack probability $p_a$. When $p_a = 1$ the attacker is a reckless adversary. Imperfect monitoring is modeled by an error parameter, $p_{err}$ representing the probability of a monitor node misidentifying the status of the trustee node due to ambient noise, temporary system faults, and/or wireless communication faults in the environment. In general a node may deduce $p_{err}$ at runtime by sensing the amount of ambient noise, system errors, and/or wireless communication errors around it.

B. Problem Definition

We define the problem to be solved in the context of Fig. 1. Broadly, the problem we are trying to solve is the vulnerability to infrastructure damage, service interruption and revenue loss caused by malicious actors. We aim to provide a solution to this problem by detecting malicious devices that exploit the vulnerability through known or unknown attacks. The solution we are offering is a behavior-rule based design with which misbehavior of a device manifested as a result of attacks exploiting the vulnerability exposed may be detected, regardless of whether the attack is known or unknown. In the context of the electrical power grid in Fig. 1, we aim to solve this problem by detecting malicious devices, including HE, DAP and SEM devices. For example, an opportunistic vandal could completely unfurl the blades of a wind DER during high wind conditions to damage the apparatus. A state sponsored attacker could open the isolation switches at a bulk energy provider to disrupt the service to the utility’s customers. A disgruntled insider at a bulk energy provider could lower the billing rate to cause the enterprise to lose money on all power it sold at the artificially depressed rate. A frugal subscriber could modify the usage reporting module of their subscriber energy meter to reduce their financial obligation to the energy provider. Regardless of the form of attacks, we aim to provide a solution for malicious device detection that is accurate in detection rate (close to 100%) while limiting the false positive probability to a minimum (e.g., less than 10%).

C. Behavior Rules

Our IDS design for the reference model relies on the use of lightweight specification-based behavior rules for each device. They are oriented toward detecting an inside attacker attached to a specific physical device, providing a continuous output between 0 and 1 (while accounting for transient faults and human errors) and allowing a monitor to perform intrusion detection on a neighbor trustee through monitoring. Here a monitor is itself a physical device with capability to do intrusion detection on trustee nodes assigned to it. For example, an SEM may monitor another SEM within radio range. An HE may monitor other HE or DAP trustee devices within radio range. Therefore, an HE might have several sets of behavior rules (and thus several state machines), one for each trustee.

Tables I, II and III list the behavior rules for the HE, DAPs and SEMs. These tables specify the trustee and monitor devices for applying the IDS technique. The networking concepts used in the behavior rules include: Packets received are the inbound protocol data units handled by the communications subsystem or application on a node; they are measured with frequency (Hz) with a domain of 0 to
10 packets per second. A node receives packets for which it is not the intended receiver, but possibly is a waypoint on the path to the destination. The communications subsystem drops these packets or relays them. Packets forwarded counts these packets the communications subsystem passes along using frequency (Hz) over the same domain as packets received. Packet sourcing is when an application generates a protocol data unit and passes it down to the communications subsystem for transmission. A good node populates these packets with legitimate sensor or status data, but a bad node populates these packets with corrupt sensor or status data or replays of previously received packets. \( \epsilon_f \) is a threshold for the difference between packets received and packets forwarded. The networking condition is an abbreviation of packets received and forwarded used to manage the size of the behavior rule state machine. \( \mu_d \) is the nominal power demand. \( \epsilon_d \) is a distance from \( \mu_d \) beyond which a control algorithm should take action to match power supply with demand. \( \mu_r \) is the nominal billing rate. \( \epsilon_r \) is a distance from \( \mu_r \) beyond which a control algorithm should take action to capitalize on a low billing rate or avoid consuming at a high one.

Our behavior-rule specification-based technique approaches the intrusion detection problem from the behavior/evidence domain compared with signature-based techniques that approach the problem from the attacker domain. Hence, the patterns by which an attacker performs attacks and “how” an attacker performs attacks do not need to be known. Rather, a monitor device simply checks the behavior of a trustee device manifested from evidence of compliance/deviation against “good” and “bad” behaviors specified by a set of behavior rules for that device. Our approach thus can address all potential attacks, known or unknown. We claim behavior rule-based detection is able to cope with unknown attacks because all attacks lead to behavior anomaly. This capability is similar to anomaly detection which, unlike signature-based detection, can cope with zero-day attacks. Nevertheless, if the rule set is incomplete, that is, if the specification of anticipated behavior is incomplete, it is possible misbehavior manifested as a result of known or unknown attacks will be missed, and, consequently, the attacker will be undetected.

**D. Transforming Rules to State Machines**

Each behavior rule does not specify just one attack state, but a number of states, some of which are good states in which good behavior (obedience of this behavior rule) is observed, while others are bad states in which bad behavior (violation of this behavior rule) is observed. A behavior rule thus has a number of state variables, each with a range of values, together indicating whether the node is in good or bad behavior status (reflecting all behavior rules).
The following procedure transforms a behavior specification into a state machine: First, the “bad behavior indicator” as a result of a behavior rule being violated is identified. Then, this bad behavior indicator is transformed into a conjunctive normal form predicate and the involved state components in the underlying state machine are identified. Next, for each device (that is, an HE, DAP or SEM), the bad behavior indicators are combined into a Boolean expression in disjunctive normal form. Then, the union of all predicate variables is transformed into the state components of a state machine and their corresponding ranges are established. Finally, the number of states is managed by state collapsing and identifying combinations of values that are not legitimate. How a state machine is derived from the behavior specification in terms of behavior rules for the reference model is exemplified below.

1) Identify Bad Behavior Indicators: Attacks performed by a compromised sensor/actuator will drive the HE, DAP or SEM into certain bad behavior indicators identifiable through analyzing the specification-based behavior rules.

For the HE device, there are nine bad behavior indicators as a result of violating the nine behavior rules for HEs listed in Table I. The first HE bad behavior indicator is that the HE activates a block of appliances but the system demand is above some threshold. The second HE bad behavior indicator is that the HE increases the duty cycle for a block of appliances but the system demand is above some threshold. The third HE bad behavior indicator is that the HE deactivates a block of appliances but the system demand is below some threshold. The fourth HE bad behavior indicator is that the HE decreases the duty cycle for a block of appliances but the system demand is below some threshold. The fifth HE bad behavior indicator is that the HE increases the billing rate but the system demand is above some threshold. The sixth HE bad behavior indicator is that the HE increases the pitch of wind DER generator blades, but the micro grid demand is above some threshold. The seventh HE bad behavior indicator is that the HE decreases the pitch of wind DER generator blades, but the micro grid demand is below some threshold.

For the DAP device, there are eight bad behavior indicators as a result of violating the nine behavior rules for DAPs listed in Table II. The first DAP bad behavior indicator is that the DAP increases the billing rate but the system demand is above some threshold. The second DAP bad behavior indicator is that the DAP deactivates a block of appliances but the system demand is below some threshold. The third DAP bad behavior indicator is that the community DER is not connected, but it is available. The fourth DAP bad behavior indicator is that the DAP increases the pitch of wind DER generator blades, but the micro grid demand is above some threshold. The fifth DAP bad behavior indicator is that the community DER is not connected, but it is available.

For the SEM device, there are nine bad behavior indicators as a result of violating the nine behavior rules for SEMs listed in Table III. The first SEM bad behavior indicator is that the SEM is not generating usage data. The second SEM bad behavior indicator is that time-independent smart appliances are active, but the billing rate is above some threshold. The third SEM bad behavior indicator is that the subscriber is not banking electricity, but the billing rate is below some threshold. The fourth SEM bad behavior indicator is that the subscriber is not banking electricity, but the demand is above some threshold. The fifth SEM bad behavior indicator is that the subscriber is not banking electricity, but the demand is below some threshold.

2) Express Bad Behavior Indicators in Conjunctive Normal Form: Tables IV, V and VI list the bad behavior indicators in Conjunctive Normal Form for HE, DAP and SEM nodes, respectively.

3) Consolidate Predicates in Disjunctive Normal Form: Each type of device (HE, DAP or SEM) has a distinct behavior rule set based on its specific control modules, actuators and sensors. Construct the DNF predicate for each device type by joining the corresponding Tables IV, V or VI expressions with a disjunction. For clarity, the DNF predicate was left unreduced; clauses in the DNF predicate are traced to behavior rules easily. This makes it evident that attacks interact through
common state variables with the same logical clause. While it will yield a more elegant expression and maybe a more efficient implementation, reducing the DNF predicate would obscure the traceability of the logical clauses and interdependence of the behavior rules.

4) Identify State Components and Component Ranges: Continuous components are quantized at integer scale in permissible ranges. For example, system demand is in the range of [0, 1000 GW] and duty cycle is in the range of [0, 100%]. Table VII shows a complete list of the permissible ranges of state components. The resulting HE automation has 623 states, out of which 1008 are identified as safe states and 2448 are unsafe states. Only two values are relevant for networking: whether or not packets forwarded and packets received differ by more than some threshold. Therefore, the domain for this component is collapsed to two values. This treatment yields a modest SEM state machine with 3 x 3 x 2 x 2 x 2 x 2 x 2 x 3 = 3456 states, out of which 1008 are identified as safe states and 1473 are unsafe states. Only three values are relevant for rate: below threshold, normal and above threshold. Also, only two values are relevant for usage reporting: current or missing. Therefore, the domains for these components are collapsed to three and two values, respectively. This treatment yields a modest SEM state machine with 2 x 3 x 2 x 3 x 2 x 2 x 2 x 3 = 3456 states, out of which 396 are identified as safe states and 3060 are unsafe states.

E. Collect Compliance Degree Data

BRIDS relies on the use of monitor nodes, e.g., an SEM or a DAP is a monitor node of another SEM. The monitor node knows the state machine of the trustee node assigned to it. The monitor node periodically measures the amount of time the trustee node stays in safe and unsafe states as the trustee node migrates from one state to another triggered by events causing state transitions. A binary grading policy, i.e., assigning a compliance degree of 1 to a safe state and 0 to an unsafe state, is considered. Let c be the compliance degree of a device. The compliance degree c of a device essentially is equal to the proportion of the time the device is in safe states. Let S be the set of safe states the trustee node traverses over a period of time T. Let t, be the sojourn time that the trustee node stays in a safe state i, as measured by the monitor node. Then the monitor node collects an instance of c by:

$$c = \sum_{i \in S} \frac{t_i}{T}$$

If a node stays only in safe states during T, then by (1), its compliance degree c is one. On the other hand, if a node stays only in unsafe states only during T, then its compliance degree c is zero. The monitor node monitors and collects the trustee node’s compliance degree history c1, c2, . . . , cn for n monitoring periods, where n is sufficiently large, based on which it concludes whether or not the trustee node is compromised.

The state machines generated are leveraged to collect compliance degree data of a good and a bad node. With (1), the compliance degree c is essentially equal to the sum of the probabilities of safe states i.e.,

$$c = \sum_{i \in S} \pi_i$$

where $\pi_i$ is the limiting
probability that the node is in state \( j \) of the state machine and \( S \) is the set of safe states in the state machine. Compliance degree history \( c_1, c_2, \ldots, c_n \) of a node is then collected by means of Monte Carlo simulation. That is, given a good (or a bad) node’s state machine, start from state 0 and then follow the stochastic process of this node as it goes from one state to another. This is continued until at least one state is reentered sufficiently often (say 100 times). Then \( \pi_j \) is calculated using the ratio of the number of transitions leading to state \( j \) to the total number of state transitions. Then one instance of compliance degree is collected. A sufficiently large \( n \) test runs was repeated to collect needed for computing the distribution of the compliance degree of a good or a bad node performing reckless or random attacks.

### F. Compliance Degree Distribution

The measurement of compliance degree of a device frequently is not perfect and can be affected by noise and unreliable wireless communication in the WAN, NAN and HAN segments. The compliance degree is modeled by a random variable \( X \) with \( G(\cdot) = Beta(\alpha, \beta) \) distribution \([18]\) with the value 0 indicating that the output is totally unacceptable (zero compliance) and 1 indicating the output is totally acceptable (perfect compliance), such that \( G(a), 0 \leq a \leq 1 \), is given by

\[
G(a) = \int_0^a \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^\alpha (1-x)^{\beta-1} dx \tag{2}
\]

and the expected value of \( X \) is given by

\[
E_X[X] = \int_0^1 \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1} (1-x)^{\beta-1} dx = \frac{\alpha}{\alpha + \beta} \tag{3}
\]

The \( \alpha \) and \( \beta \) parameters are to be estimated based on the method of maximum likelihood by using the compliance degree history collected \( c_1, c_2, \ldots, c_n \) during the system’s testing phase. The maximum likelihood estimates of \( \alpha \) and \( \beta \) obtained by numerically solving the following equations:

\[
\begin{align*}
\sum_{i=1}^{n} \log c_i &= 0 \\
\sum_{i=1}^{n} \log(1-c_i) &= 0 \\
\frac{\partial \Gamma(\alpha + \beta)}{\partial \alpha} + \frac{\partial \Gamma(\alpha)}{\partial \alpha} &= 0 \\
\frac{\partial \Gamma(\alpha + \beta)}{\partial \beta} + \frac{\partial \Gamma(\beta)}{\partial \beta} &= 0
\end{align*} \tag{4}
\]

where

\[
\partial \Gamma(\alpha + \beta) = \int_{0}^{\infty} \log x x^{\alpha-1} e^{-x} dx.
\]

A less general, though simpler model, is to consider a single parameter Beta distribution with \( \alpha \) equal to 1. In this case, the density is \( \beta(1-x)^{-1} \) for \( 0 \leq x \leq 1 \) and 0 otherwise. The maximum likelihood estimate of \( \beta \) is

\[
\hat{\beta} = \frac{\sum_{i=1}^{n} \log \left( \frac{1}{1-c_i} \right)}{n}
\]

The reason the Beta distribution is chosen is that the domain of the Beta distribution can be viewed as a probability, so it can be used to describe the prior distribution over the probability (of a distribution) which models the node compliance degree. By applying Bayesian inference, the Beta distribution then can be used as the posterior distribution of the probability after observing sufficient instances.

#### G. False Negative and Positive Probabilities

Our intrusion detection technique is characterized by false negative and positive probability, denoted by \( p_{fn} \) and \( p_{fp} \), respectively. A false negative occurs when a bad node is missed as a good device, while a false positive occurs when a good node is misdiagnosed as a bad device. While neither is desirable, a false negative is especially impactful to the system’s continuity of operation. In this paper, a threshold criterion is considered. That is, if a bad node’s compliance degree denoted by \( X_b \) with a probability distribution obtained by (2) is higher than a system minimum compliance threshold \( C_T \), then there is a false negative. Suppose that the compliance degree \( X_b \) of a good node is modeled by a Beta distribution \( G(\cdot) = Beta(\alpha, \beta) \) distribution. Then the host false negative probability \( p_{fn} \) is given by

\[
p_{fn} = Pr\{X_b > C_T\} = 1 - G(C_T). \tag{6}
\]

On the other hand, if a good node’s compliance degree denoted by \( X_g \) is less than \( C_T \), then there is a false positive. Again suppose that the compliance degree \( X_g \) of a good node is modeled by a Beta distribution \( G(\cdot) = Beta(\alpha, \beta) \) distribution. Then the host false positive probability \( p_{fp} \) is given by

\[
p_{fp} = Pr\{X_g < C_T\} = G(C_T). \tag{7}
\]

### III. Numerical Data

Numerical data is reported in this section. A sequence of compliance degree values \( \{c_1, c_2, \ldots, c_n\} \) is first collected for a good or bad device based on Monte Carlo simulation. Equation (5) is then applied to compute the parameter value of \( G(\cdot) = Heta(\alpha, \beta) \) for the probability distribution of the compliance degree for a good device or a bad device performing random attacks. \( p_{fn} \) and \( p_{fp} \) are then calculated by (6) and (7), respectively. The minimum compliance threshold \( C_T \) is then adjusted to control \( p_{fn} \) and \( p_{fp} \) obtainable. With \( \mu_{err} \) a monitor node can misidentify the status the trustee node is in. \( \mu_{err} \) is set to 0.010, 0.015 and 0.020 for HE, DAP and SEM nodes, respectively. This is because 1–2% mis-monitoring due to ambient noise and wireless communication faults in these environments is reasonable. This is based on Lin and Latchman reporting a 0.11–2.04% Power Line Communication packet error rate [15] and Hong et al. reporting a 0.02–4% failure rate [11]. The mis-monitoring error probability of an SEM toward another SEM is higher than that of a DAP toward another DAP, or an HE toward another HE because of limited range and capability of an SEM device.

Tables VIII, IX and X show the \( \beta \) values and the resulting \( p_{fn} \) and \( p_{fp} \) values when \( C_T \) is 0.9 (\( C_T \) is a design parameter.
to be fine-tuned to trade high false positives for low false negatives. Because the expected compliance degree following a distribution is as given by (3), it is seen that \( \beta \) is close to 0 for a good node or a hidden bad node with a low attack probability (e.g., \( p_a = 0.05 \)) since such a node will have the average compliance degree close to 1. On the other hand, \( \beta \) is much larger than 0 for a bad node with a high attack probability (e.g., \( p_a = 1 \)) since such a node will have the average compliance degree much lower than 1.

It is observed that when the random attack probability \( p_a \) is high, the attacker can be easily detected as evidenced by a low false negative probability. Especially when \( p_a = 1 \), a reckless attacker can hardly be missed. On the other hand, as \( p_a \) decreases, the attacker becomes more hidden and insidious, and the false negative probability increases. The false positive probability remains the same regardless of the random attack probability because it is not related to the attacker behavior. By adjusting \( C_T \), the specification-based IDS technique can effectively trade higher false positives for lower false negatives to cope with more sophisticated and hidden random attackers. This is especially desirable for ultra safe and secure applications for which a false negative may have a dire consequence.

Fig. 2 shows a Receiver Operating Characteristic (ROC) graph of intrusion detection rate (i.e., \( 1 - p_f \)) versus false positive probability (\( p_f \)) obtained as a result of adjusting \( C_T \). In Fig. 2 there are several curves for each node type, one for each random attacker case with a different attack probability (e.g., \( p_a = 1 \)). The curves have the same general shape: as the attack probability increases, the detection rate increases (vertically up on a ROC graph) while the false probability increases (toward the right of a ROC graph). It is seen that with the specification-based IDS technique, the detection rate of the node can approach 100% for detecting attackers, that is, an attacker is always detected with probability 1 without false negatives, while bounding the false positive probability to below 0.2% for reckless attackers and below 6% for random attackers.

### IV. Comparative Analysis

The performance of BRIDS is compared with contemporary anomaly-based IDSs for HEs, DAPs and SEMs, including CLONALG and AIRS2Parallel [22], [23], LOUD, LOED, LOUD-GLR and LOED-GLR [10]. Zhang et al. [23] reported that CLONALG had a false positive rate of 0.7% and a false negative rate of 21.02% and AIRS2Parallel had a false positive rate of 1.3% and a false negative rate of 26.32%. Zhang et al. [22] further compared the effectiveness of audit data from three sources: home IDS (HIDS), neighborhood IDS (NIDS) and wide-area IDS (WIDS). These three approaches correspond with the SEM, DAP and HE nodes identified in Fig. 1. Here the authors reported that CLONALG had an accuracy of 99.70% for HEs, 80.10–97.00% for DAPs and 93.90–99.30% for SEMs. They reported that AIRS2Parallel had an accuracy of 91.50% for HEs, 82.10–96.10% for DAPs and 95.10–98.70% for SEMs. The authors provided no information, but presumably the worst detection accuracy is obtained when \( C_f \) is very low. He and Blum [10] investigated LOUD, LOED, LOUD-GLR and LOED-GLR approaches to anomaly-based IDS. They fixed the false positive probability (i.e., \( p_f \)) at 0.1% and showed that the detection rate (i.e., \( 1 - p_f \)) for each approach varies over a wide range based on the parameterization. The LOUD-GLR approach reportedly performs the best with the detection accuracy of 100 – 0.1 = 94.9%.

Tables XI, XII and XIII summarize the comparative performances among contemporary IDSs for HE, DAP and SEM devices, respectively. The performance metric is detection accuracy defined as \( 1 - p_f - p_i \). For cases where \( p_i \) and \( p_f \) are reported [10], [23], the detection accuracy value is shown following the \( 1 - p_f - p_i \) format. For cases where \( p_i \) and \( p_f \) are not reported [22], the detection accuracy value or a range of detection accuracy values is shown only. For comparison, the

### TABLE VIII

\[ \beta \) in Beta(1, \( \beta \),) and Resulting \( p_a \) and \( p_f \) Values Under Various Attack Models for HE (\( C_f = 0.50 \))

<table>
<thead>
<tr>
<th>Attack Type</th>
<th>( \beta )</th>
<th>( p_a )</th>
<th>( p_f )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random (( p_a = 1.00 ))</td>
<td>99.5</td>
<td>0.001%</td>
<td>2.30%</td>
</tr>
<tr>
<td>Random (( p_a = 0.80 ))</td>
<td>4.33</td>
<td>0.0047%</td>
<td>2.30%</td>
</tr>
<tr>
<td>Random (( p_a = 0.40 ))</td>
<td>1.10</td>
<td>7.99%</td>
<td>2.30%</td>
</tr>
<tr>
<td>Random (( p_a = 0.20 ))</td>
<td>0.633</td>
<td>23.3%</td>
<td>2.30%</td>
</tr>
<tr>
<td>Random (( p_a = 0.10 ))</td>
<td>0.449</td>
<td>35.5%</td>
<td>2.30%</td>
</tr>
<tr>
<td>Random (( p_a = 0.05 ))</td>
<td>0.353</td>
<td>44.3%</td>
<td>2.30%</td>
</tr>
</tbody>
</table>

### TABLE IX

\[ \beta \) in Beta(1, \( \beta \),) and Resulting \( p_a \) and \( p_f \) Values Under Various Attack Models for DAP (\( C_f = 0.50 \))

<table>
<thead>
<tr>
<th>Attack Type</th>
<th>( \beta )</th>
<th>( p_a )</th>
<th>( p_f )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random (( p_a = 1.00 ))</td>
<td>49.6</td>
<td>0.001%</td>
<td>4.59%</td>
</tr>
<tr>
<td>Random (( p_a = 0.80 ))</td>
<td>4.19</td>
<td>0.0064%</td>
<td>4.59%</td>
</tr>
<tr>
<td>Random (( p_a = 0.40 ))</td>
<td>1.10</td>
<td>7.89%</td>
<td>4.59%</td>
</tr>
<tr>
<td>Random (( p_a = 0.20 ))</td>
<td>0.644</td>
<td>22.7%</td>
<td>4.59%</td>
</tr>
<tr>
<td>Random (( p_a = 0.10 ))</td>
<td>0.464</td>
<td>34.3%</td>
<td>4.59%</td>
</tr>
<tr>
<td>Random (( p_a = 0.05 ))</td>
<td>0.372</td>
<td>42.5%</td>
<td>4.59%</td>
</tr>
</tbody>
</table>

### TABLE X

\[ \beta \) in Beta(1, \( \beta \),) and Resulting \( p_a \) and \( p_f \) Values Under Various Attack Models for SEM (\( C_f = 0.50 \))

<table>
<thead>
<tr>
<th>Attack Type</th>
<th>( \beta )</th>
<th>( p_a )</th>
<th>( p_f )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random (( p_a = 1.00 ))</td>
<td>32.8</td>
<td>0.001%</td>
<td>6.87%</td>
</tr>
<tr>
<td>Random (( p_a = 0.80 ))</td>
<td>4.06</td>
<td>0.0086%</td>
<td>6.87%</td>
</tr>
<tr>
<td>Random (( p_a = 0.40 ))</td>
<td>1.11</td>
<td>7.78%</td>
<td>6.87%</td>
</tr>
<tr>
<td>Random (( p_a = 0.20 ))</td>
<td>0.656</td>
<td>22.1%</td>
<td>6.87%</td>
</tr>
<tr>
<td>Random (( p_a = 0.10 ))</td>
<td>0.479</td>
<td>33.2%</td>
<td>6.87%</td>
</tr>
<tr>
<td>Random (( p_a = 0.05 ))</td>
<td>0.290</td>
<td>40.7%</td>
<td>6.87%</td>
</tr>
</tbody>
</table>

Fig. 2. HE receiver operating characteristic graph.
adversary is configured with \( p_a = 1 \) (releck attacks). BRIDS performance is shown for \( C_T = 0.50 \) for HE, \( C_T = 0.37 \) for DAP and \( C_T = 0.29 \) for SEM to approximate the CLONALG \( p_{df} \) of 0.7\% [23]. BRIDS performance is shown for \( C_T = 0.73 \) for HE, \( C_T = 0.58 \) for DAP and \( C_T = 0.47 \) for SEM to approximate the AIRS2Parallel \( p_{df} \) of 1.3\% [23]. BRIDS performance is shown for \( C_T = 0.10 \) for HE, \( C_T = 0.06 \) for DAP and \( C_T = 0.05 \) for SEM to approximate the LOUD, LOED, LOUD-GLR and LOED-GLR \( p_{df} \) of 0.1\% [10].

Tables XI, XII and XIII support the claim that by effectively adjusting \( C_T \) to trade false positives for low false negatives, BRIDS outperforms existing anomaly-based IDS approaches, especially for HE and DAP devices.

### V. CONCLUSIONS

For a modern electrical grid, being able to detect attackers while limiting the false positive probability to protect the continuity of operation is of utmost importance. In this paper, a behavior-rule specification-based IDS technique for intrusion detection of physical devices was proposed. The utility by head-ends, distribution access points/data aggregation points and subscriber energy meters was exemplified. This study also demonstrated that the detection probability approaches one (that is, the attacker can always be caught without false negatives) while bounding the false positive probability to below 0.2\% for reck-lers attackers and below 6\% for random attackers (that is, the probability of misidentifying a good node as a bad node can always be bounded to a very low level). Through a comparative analysis, it was demonstrated a behavior-rule specification-based IDS technique outperforms existing anomaly-based IDS approaches for detecting intruders.

Two future research directions extending from this study are (a) investigating and analyzing intrusion response and repair strategies [17]; and (b) implementing behavior rules on applications. Possible intrusion responses include evicting individual compromised nodes, isolating compromised segments (micro grid or larger scope) and adjusting IDS parameters (e.g., \( T_{IDS}, \tau_R \) and \( C_T \)) to increase detection strength. Possible repair strategies are to identify compromised segments and for each one: stop operating, revert all nodes to certified software loads and configurations, rekey/reset passwords and progressively resume operation from the production side of the network towards the subscribers. Possible implementation strategies are to encode the state machine, host IDS software and system IDS software in a high-level language, cross-compile for the targets of interest, deploy and tune the parameterization (e.g., \( T_{IDS}, \tau_R \) and \( C_T \)) based on desired versus actual false negative and positive rates. Another future research direction is to investigate other intrusion detection criteria [1], [4], [5] based on accumulation of deviation from good states in addition to the current binary criterion used in the paper based on a minimum compliance threshold to further improve the detection rate without compromising the false positive probability.

### REFERENCES


**Rob Mitchell** received the B.S. and M.S. degrees in computer science from Virginia Polytechnic Institute and State University, Falls Church, VA, USA, in 1997 and 1998, respectively, where he is currently pursuing the Ph.D. degree in the Department of Computer Science.

His research interests include security, mobile computing, sensor networks, embedded systems, and coding and information theory.

**Ing-Ray Chen (M’98)** received the B.S. degree from the National Taiwan University, Taipei, Taiwan, and the M.S. and Ph.D. degrees in computer science from the University of Houston, Houston, TX, USA.

He is a professor in the Department of Computer Science at Virginia Polytechnic Institute and State University, Falls Church, VA, USA. His research interests include mobile computing, wireless networks, security, intrusion detection, trust management, real-time intelligent systems, and reliability and performance analysis.

Dr. Chen currently serves as an Editor for IEEE COMMUNICATIONS LETTERS, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, Wireless Communications and Mobile Computing, The Computer Journal, Wireless Personal Communications, and Security and Communication Networks. He is a member of ACM.