SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks (2014)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.957

RESEARCH ARTICLE

An integrated detection system against false data injection attacks in the Smart Grid

Wei Yu[1]*, David Griffith[2], Linqiang Ge[1], Sulabh Bhattarai[1] and Nada Golmie[2]

[1] Department of Computer and Information Sciences, Towson University, Towson, MD, U.S.A.
[2] National Institute of Standards and Technology (NIST), Gaithersburg, MD, U.S.A.

ABSTRACT

The Smart Grid is a new type of power grid that will use advanced communication network technologies to support more efficient energy transmission and distribution. The grid infrastructure was designed for reliability; but security, especially against cyber threats, is also a critical need. In particular, an adversary can inject false data to disrupt system operation. In this paper, we develop a false data detection system that integrates two techniques that are tailored to the different attack types that we consider. We adopt *anomaly-based detection* to detect strong attacks that feature the injection of large amounts of spurious measurement data in a very short time. We integrate the anomaly detection mechanism with a *watermarking-based detection scheme* that prevents more stealthy attacks that involve subtle manipulation of the measurement data. We conduct a theoretical analysis to derive the closed-form formulae for the performance metrics that allow us to investigate the effectiveness of our proposed detection techniques. Our experimental data show that our integrated detection system can accurately detect both strong and stealthy attacks. Copyright © 2014 John Wiley & Sons, Ltd.

KEYWORDS
Smart Grid; false data injection attacks; countermeasures

***Correspondence**
Wei Yu, Department of Computer and Information Sciences, Towson University, Towson, MD, U.S.A.
E-mail: wyu@towson.edu

1. INTRODUCTION

In the Smart Grid, advanced communication network technologies will be used to make the next generation power grid efficient, reliable, and secure. To enable the effective operation of the Smart Grid, the supervisory control and data acquisition (SCADA) system is used to enable wide area situational awareness of power grid status, and the advanced metering infrastructure (AMI) system is used to provide the two-way communication between customers and utility providers that can promote energy usage efficiency.

The operation and control of the Smart Grid highly depends on a complex network of computers, software, and communication technologies. An adversary who is able to compromise the system can target elements of this critical infrastructure and potentially pose great damage to the grid, including extended power outages and destruction of electrical equipment. For example, cyber attacks can be launched through the public network from virtually anywhere; coordinated attacks can originate from different places and impact multiple targets in the Smart Grid. These attacks can take many forms. Examples include spreading malware, exploiting vulnerabilities in protocols, gaining access to the power grid's control system grid through network or information interfaces, eavesdropping on sensitive communications or stealing information, injecting false pricing or meter-reading measurements, and many others [1].

Because the measurement component supported by equipment such as smart meters and sensors plays an important role in the Smart Grid, it can also be a target for cyber attacks. For example, Mo *et al.* stated that an adversary can have a wide variety of motivations for launching attacks on the power grid that range from economic reasons and pranks all the way up to terrorism [2]. Once adversaries gain access to the Smart Grid, they can conduct a wide range of attacks: gaining unauthorized access to meter and sensor data, reporting incorrect meter and sensor data, making billing information unavailable, disrupting system operation, and others. Because those measuring devices may be connected through network interfaces, an adversary can possibly launch attacks against those devices [3,4]. For example, the adversary can modify data

and compromise the measuring component through injecting malicious codes into the memory of the measuring component [5]. The adversary then injects false energy demand and supply messages into the power grid through compromised measuring components. *Stuxnet* is a particularly notorious example of this type of malware, which actually exploits vulnerabilities in SCADA systems to cause physical damage to controlled components [6]. Note that, if a node is denoted as a compromised node, we mean that the measuring component of the node is compromised by the adversary.

Of particular concern is the use of false measurement data from compromised meters or sensors to disrupt Smart Grid operations. False data injection attacks can have significant negative consequences. For example, an adversary may attempt to send low usage data to obtain lower electric bills that would have severe impacts if performed on a large scale [7], or an adversary can send large amounts of spoofed data to the utility, leading to an incorrect estimation of the state of the grid [8] or incorrect energy distribution decision [9], with potentially catastrophic results. For example, one of the main causes of the massive blackout on 14 August 2003, which affected 50 million people in the Northeast USA, and Ontario and Québec provinces in Canada, was the state estimation programs for key areas that failed to provide the system operators with accurate information [10]. In addition, Liu *et al.* [8] showed that an adversary, armed with the knowledge of a network's configuration, can inject false data into the state estimation algorithms without being detected. Yuan *et al.* [11] developed a new false data injection attack, namely, the load redistribution attack, and quantitatively analyzed its damage on power system operations. Lin *et al.* [9] investigated the vulnerabilities of distributed energy distribution processes and investigated a number of possible attacks against distributed energy resources. As we can see, defending against data injection attacks is a critical issue in Smart Grid research and development.

To address this critical issue, our paper makes the following novel contributions.

First, we propose a detection system that integrates anomaly-based detection with data watermark-based detection. We consider the following two general types of attacks: (i) strong attacks and (ii) stealthy attacks. The goal of strong attacks is to inflict maximum damage in the shortest possible time. Such attacks can be detected by the statistical anomaly-based detection, which will serve as our first line of defense. A stealthy adversary, in contrast, is more interested in compromising and manipulating the Smart Grid over a long period while avoiding detection. Such attacks are difficult to uncover with anomaly-based detection alone. To defend against such stealthy attacks, we propose the watermarking-based detection scheme as our second line of defense. Note that our watermarking-based detection embeds a light-weight pseudorandom noise (PN) code and requires very low computational resources.

Second, we use real-world measurement data obtained from the Internet and from Stanford University [12] to derive an empirical distribution of the metering data and analyze the effectiveness of our proposed defense schemes. Using the real-world meter-reading data set, we conducted experiments and confirmed that metering data can be well-approximated with a Gaussian distribution. Using this fact, we are able to define metrics to compute the accuracy of our detection scheme and present analytical results for both anomaly-based and watermarking-based detection. Our statistical anomaly-based detection scheme uses the assumption that the behavior of an adversary differs from that of a legitimate grid customer or operator in ways that can be quantified [13]. We develop a receiver operating characteristic (ROC) that shows how to adjust the sensitivity of the detector based on the trade-off between maximizing the detection rate and minimizing the false positive rate. We also analyze the impact of different parameters on detection accuracy in different scenarios in the watermarking-based detection scheme. On the basis of the analytical results, we discuss how to determine the severity of false data injection attacks, that is, the fraction of data that is falsely injected by the adversary.

To deal with stealthy attacks, our watermark-based detection scheme embeds secured watermarks or signals (i.e., PN codes with a chip rate higher than the protected data rate) in the real-time measuring data stream prior to transmission over the communication network (e.g., Internet Protocol [IP] networks). However, the communication network may not be secured and thus may be prone to cyber attacks. To detect false data injection attacks over the potentially unsecured communication network, we embed secure watermarks into the real-time measuring data and transmit the resulting watermarked data stream to the utility provider. The measurement device can use a separate, dedicated, low rate, and secure channel that can be provided as part of the existing power grid infrastructure (e.g., closed serial networks and power line communication) to transmit and synchronize watermarks. We assume that the storage and transmission of watermarks and the synchronization of watermarks between measurement devices and utility provider are secure. The utility providers and other network devices (e.g., aggregators) use both the watermark and the watermarked data in a correlation-based test that can detect false data injection attacks in the transmission path.

Third, we conduct experiments using the real-world meter-reading data set as an example to demonstrate the effectiveness of our proposed integrated detection schemes. The experimental data show that the anomaly-based detection scheme can accurately detect strong and rapid attacks and, when properly tuned, has a low false positive rate and a high detection rate. Our watermarking-based detection scheme can effectively deal with stealthy attacks and estimate the attack strength. For example, as the strength of an attack increases, our correlation test result decreases in a linearly proportional fashion, which means that we can effectively detect any modification to the watermarked data stream.

A preliminary version of this paper appeared in the Proceedings of the IEEE International Conference on Communication, June 2012. The conference version was only five pages while this journal extended version is over 15 pages. In comparison with the conference version, we have significantly extended the paper and included substantial new materials such as new anomaly detection algorithm, metering data analysis, theoretical modeling and analysis, evaluation results, background, discussion, and others.

The remainder of our paper is organized as follows: we introduce the system models in Section 2. In Section 3, we present our integrated detection approach in detail. We conduct modeling and analysis of the effectiveness of our proposed techniques in Section 4, and we present the evaluation results of our proposed techniques in Section 5. We discuss our techniques to deal with existing false data injection attacks in Section 6. We conduct the literature review in Section 7. Finally, we conclude our paper in Section 8.

2. SYSTEM MODELS

In the Smart Grid, there are two major systems: AMI and SCADA. With smart meters as measurement components, the AMI system is used to enable two-way communication between customers and utility providers to promote efficient energy usage. A smart meter collects usage data and transmits these data to the utility provider for monitoring and billing purposes. Through smart meters, appliances and other devices at the customer-premises equipments can be remotely controlled to allow the operator to efficiently balance generation and load. However, the wide use of smart meters also creates the potential for an adversary to compromise those devices. Once compromised, demand requests from those devices can be forged (e.g., requesting a large amount of energy), which can mislead the electric utility with incorrect information about regional energy usage. If an adversary controls a large number of smart meters and manipulates measurement data from them, the demand response algorithms in the Smart Grid can be seriously disrupted.

The SCADA system is a critical element of the Smart Grid that is used to enable wide area situational awareness of power grid status. In particular, the utility provider uses the SCADA system to monitor and control the physical plant in the power grid with the aid of various sensing devices. In a SCADA system, distributed remote terminal units (RTUs) or programmable logical controllers are deployed in the field, along with sensors that monitor the status of the power grid (e.g., power generation, transmission, and distribution). The RTUs or programmable logical controllers transmit the measurement data to a regional SCADA controller that transmits the information to operating stations. The sensor data can be transmitted through a variety of communications channels such as wireless, fiber optics, or copper wire. Historically, a SCADA system was self-contained and used dedicated wide area networks (WANs) running proprietary protocols for communications between RTUs and the central SCADA computer, and operating stations and computers were on the same local area network. However, modern SCADA systems consist of distributed functions connected over a WAN running IPs. This can enhance reliability but creates opportunities for adversaries.

Generally speaking, both AMI and SCADA consist of three key components: the meter (or sensor)[†] at the customer's premises or in the field, the data aggregator, and the utility. All components in the Smart Grid are connected through the computer networks that increasingly use standard protocols, such as IP, and which may not be fully secured. The measurement data from smart meters is typically generated at regular intervals (e.g., hourly); data can also be pulled "on-demand" from the meter at any time by the customer (e.g., via a web portal owned by the utility provider) or by the provider itself (e.g., for auditing purposes). Other elements of the Smart Grid generate data more frequently. For example, the sensor data from devices such as phasor measurement units can be generated over one hundred times per second. We also assume that, in addition to the less secure channels over commercial WANs, other channels that are more reliable and secure but typically offer lower data rates (e.g., closed serial networks, hardened secure communication lines, power line communication, and others) are available to connect meters and sensors to the utility through the aggregator. In this paper, we will consider using such channels for transmitting watermarks.

An adversary may launch cyber attacks by compromising individual meters or sensors, or by hacking into the communication networks between meters and AMI/SCADA systems, or by breaking into the AMI and/or SCADA systems through the control center office local area network or through the customer's home area network [3,4,8,14]. As an example, an adversary can modify data and compromise the meter and sensor through injecting malicious codes into the memory of measuring components [5]. By compromising the communication network, an adversary can disrupt packet flows (e.g., rerouting traffic to the adversary's network) so that they can be modified at leisure. An adversary can impersonate the control center and send unauthorized commands to meters and sensors as well. Note that using encryption and decryption techniques cannot resolve the problem that we address in this paper because of the high cost and the need to support legacy sensors and meters in the power grid.

We assume that an adversary can compromise multiple components, including meters, sensors, aggregators, and data transmission channels (e.g., network interfaces). After a component or data communication channel is compromised, the information stored in it or transmitted by it becomes visible to the adversary. An adversary can

[†] Note that the terms *meter* and *sensor* are used interchangeably in this paper, except when specifically stated otherwise.

manipulate data and send false data through compromised measurement components, which causes the utility provider to obtain incorrect measurement readings from the power grid that can result in loss of revenue. In addition, the false data injection consumes network and computation resources and renders AMI and SCADA systems ineffective. We assume that while elements of the metering network are connected to the public Internet, the utility itself is connected to a secured network and cannot be directly compromised.

The power grid infrastructure is commonly homogeneous, especially on the consumer side; thus, all meters are equally vulnerable to compromise. This is an interesting issue to pursue in the future. One possible approach is to deploy heterogeneous meters with different operating systems; this would reduce the risk of the entire set of meters being compromised simultaneously. Note that protecting the ensemble of meters from being compromised is not the focus of this paper; rather, our focus is to develop means of detecting false data injected through compromised measurement components. We also note that watermarking can be applied on a meter-by-meter basis so that corruption of the metering data is detectable even if a large number of meters are affected. In fact, this approach would allow the utility to identify which meters have been compromised.

3. OUR APPROACH

In this section, we present our approach to detect false data injection attacks in the Smart Grid. Recall that to disrupt the Smart Grid, an adversary needs to compromise the measurement components in either the meter in AMI, the sensor in SCADA, the aggregator, or the utility. The adversary can make large changes to the measurement data in a short period or make small, subtle changes over a long time interval. The former case can be classified as a strong and rapid attack that can cause significant damage to the grid but which can be detected by the anomaly-based detection scheme that we will introduce first. The latter case can be classified as a slow and stealthy attack that can still cause significant damage to the grid over the long run. To deal with slow and stealthy attacks, we propose a novel watermarking-based detection scheme that we will describe in detail, including the methods of attack detection under different scenarios. Except where otherwise stated, we use meter-reading data as an example to demonstrate and evaluate our developed detection schemes. However, our developed detection schemes are generic and can be applied to detect false data from sensors in SCADA systems as well as meters in AMI networks.

We conduct experiments using real-world meter-reading data sets from meters in an AMI network as an example to demonstrate the effectiveness of our proposed integrated detection schemes. We plan to evaluate the performance using sensor data set from SCADA if such data set becomes publicly available. Unfortunately, to the best of our knowledge, there is no such publicly available SCADA sensor data set; thus, we leave the SCADA verification for future work. We note, however, that SCADA systems have tight timing constraints, the sensors that make up these systems have limited resources available, and there are many legacy devices in the power grid that cannot be easily replaced. Because our watermark-based detection scheme is computationally lightweight and can be implemented easily on legacy equipment, we expect that the detection techniques discussed in this paper will be able to detect false data injection attacks in SCADA systems at relatively low cost. All notations can be found in Table I.

3.1. Anomaly-based detection

An intrusion detection system can be classified as either signature-based or anomaly-based. In signature-based detection, a large repository of known attack signatures will be maintained in a database and will be used to detect known attacks [15]. Nevertheless, the disadvantage of this approach is that it cannot detect new attacks. In anomaly-based detection, the security administrator defines the baseline, or normal, measures of the system behavior. The detection system monitors various system segments to compare their states to the defined normal baselines and then identify anomalies. One common technique is to estimate the normal behavior of the system to be protected and generate a detection alert whenever the deviation between the observed behavior and normal behavior exceeds a predefined threshold [13,16].

Our proposed anomaly-based defense is a statistical threshold-based detection scheme that works in the following way. At the utility side, we calculate m and v, which are, respectively, the mean and standard deviation of the sum of a given set of meter-reading data. We will use m and v as the metrics of normal behavior of the system. Next, we establish a pair of thresholds T_l and T_h for identifying readings from the data source as anomalous; we set $T_l = m - kv$ and $T_h = m + kv$, where k determines the maximum acceptable degree of deviation from the original measurement data. We define $\mathbf{X} = \{X_i\}_{i=1}^{n}$ to be a set of n data points from the source; X_i is the i^{th} element of $\mathbf{X}$. We can consider X_i anomalous if $X_i \notin [T_l, T_h]$. We show the full statistical anomaly-based procedure for detecting the presence of anomalous data in the set $\mathbf{X}$ in Algorithm 1.

To set up an anomaly-based detection scheme, the utility provider needs to obtain the statistical parameters of the meter-reading data by collecting legitimate data from its customers and determining the proper threshold to use. In real-world practice, electricity usage varies during the day and depends on many factors such as weather. Note that in Section 4.1, we conduct an analysis using the real-world meter-reading data set and show that the probability distribution of the meter data can be approximated with a Gaussian distribution.

Once an adversary knows our detection methods, they may pump spurious data that follow a Gaussian distribution. However, from the adversary's perspective, the

Algorithm 1 Anomaly-based detection.

Input: **Received Meter-Reading Data**
    Set of n Meter Readings: $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$
Parameter: Threshold parameter k
Output: X_i ($i = 1, 2, \ldots, n$) is normal data or not

$m = \mathsf{E}[\mathbf{X}]$
$v = \sqrt{\mathsf{Var}[\mathbf{X}]}$
$S_X = \sum_{i=1}^{n} X_i$
if $|S_X - m| > kv$ **then**
    S_X contains abnormal data
else
    S_X contains only normal data
end if

manipulated data must be significantly different from the legitimate measurements in order to cause damage (e.g., the mean value of manipulated measurements will be different from that of the true measurements even though the adversary may pump their data following a Gaussian distribution). For example, the adversary may compromise the meter and manipulate meter-readings such that they follow a Gaussian distribution over a long time interval. This approach is a slow and stealthy attack, rather than a strong and rapid one. To defend against such stealthy attacks, we propose the watermarking-based detection scheme as our second line of defense, which we discuss next.

3.2. Watermarking-based detection

Generally speaking, watermarking is a means of embedding data in a data stream so that the watermark uniquely identifies the originator of the data stream. In network security applications, a watermarker varies the transmission rate of packets in a traffic flow during predetermined time intervals actively. Usually the watermarker uses two rates, which encode the zeros and ones that compose the embedded watermark. For example, by examining the packet arrival rate in these intervals, an accomplice located elsewhere in the network can determine the path that the flow follows through the anonymous network [17,18]. This technique has been proposed for network flow monitoring and was originally proposed as a forensic technique to identify malicious anonymous communications or as a possible attack that can be used to eavesdrop on anonymous communications [17,18]. In a departure from the existing work on this technique, we consider using watermarking as a means to detect slow and stealthy false data injection attacks by validating the integrity of the meter-reading data stream. We assume that the storage and transmission of watermarks, and the synchronization of watermarks between measurement devices and utility provider, are secured.

Unlike standard encryption and decryption schemes, a watermark is lightweight PN code whose modulation process requires minimal computational resources. Hence, the watermarking-based detection method can be easily implemented in legacy devices and cost-effective meters (or sensors) in the power grid. Also, note that using encryption and decryption techniques cannot resolve the problem that we address in this paper because of the higher computational cost and the need to support legacy sensors and meters that have already been deployed. Generally speaking, the more complex and secure the encryption, the greater the computational overhead. However, endpoint devices such as smart meters or sensors are developed using cost-effective hardware that may not be capable of supporting the most robust encryption algorithms.

Table I. Notation.

X:	Meter-reading data stream
m, v, S_x:	Mean value, standard derivation, and sum of meter-reading data stream X
k:	Parameter to determine the threshold for anomaly detection
T_h, T_l:	Thresholds for anomaly detection
T_w:	Threshold for watermark detection
C_t:	PN code at time t generated at the transmitter
C_r:	The local generated PN-code at the receiver
L:	Length of PN code
B:	Transmitted baseband signal
A:	Watermark amplitude
W_t:	Watermark value at time t
T_t:	Watermarked meter-reading data transmitted at the transmitter
λ_t:	Watermarked data received at the receiver
λ'_t:	Watermarked data received at the receiver after filtering the direct current component
$\mathbf{S}$:	Similarity degree for watermarking detection
H_0:	Hypothesis that data follow a Gaussian distribution
H_1:	Hypothesis that data do not follow a Gaussian distribution
α':	Significance level for Kolmogorov-Simirnov test
P_D:	Detection rate as the probability of correctly determining the meter-reading data has been falsely injected
P_F:	False positive rate defined as the probability an attack is mistakenly detected when no attack occurs
C_M:	Cost of a missed attack detection
C_F:	Cost of a false alarm
$\rho = C_F/C_M$:	Ratio of the false alarm and missed detection costs
σ_x:	Standard deviation of the meter-reading data
$\Delta(\cdot)$:	Correlation operator for computing the similarity degree
ω:	Converted meter-reading data after subtracting the mean value of original meter-reading data
p:	Attack severity degree as the probability of the meter-reading data being falsely injected

PN, pseudorandom noise.

In contrast, our watermarking-based detection scheme embeds a light-weight PN code and requires very few computational resources. Finally, encryption and decryption might not be supported by legacy meters or sensors. In contrast, our watermarking-based detection methods can be easily and efficiently used by those devices.

Our watermarking-based detection scheme is based on direct sequence spread spectrum (DSSS) modulation. In DSSS communications systems, a bit stream $b(t)$ is modulated by a higher-rate data stream $c(t)$ whose elements, known as chips, are of shorter duration than the bits that they are modulating. Chip values are generated by a periodically repeating PN code, and the chip stream's spectrum is wider than that of the bit stream because of the shorter chip period. Because the spectrum of the resulting signal is the convolution of the spectra of the bit stream and the chip stream, it will be spread over a wide bandwidth. The receiver modulates the received sequence with its own copy of the chip sequence $c(t)$, resulting in the retrieval of the original bit stream $b(t)$.

In our approach, the watermark sequence plays the role of the bit stream, which we spread with a PN code prior to adding the resulting sequence to the metering data that we want to validate. An important point is that the rate of the baseband metering data sequence (i.e., the rate at which the meter generates data points) is the same as the rate of the PN code. Thus, the watermark sequence is constant over multiple metering data points. In addition, we insert the watermark at a random location in the metering data stream; the adversary therefore does not know where the watermark is. In this paper, the watermark consists of a sequence of ones equal in length to one period of the PN code, but more complex watermarks can easily be devised, using this approach.

For DSSS watermarking, there are two important modules within the framework: (i) mark generation at the transmitter (steps 1 and 2 listed in the succeeding text) and (ii) mark recognition at the receiver (steps 3 and 4 listed in the succeeding text). The system architecture of watermark encoding and decoding consists of four major components: *signal modulator*, *data modulator* (both at the transmitter's side), *data demodulator*, and *signal demodulator* (both at the receiver's side), as shown in Figure 1.

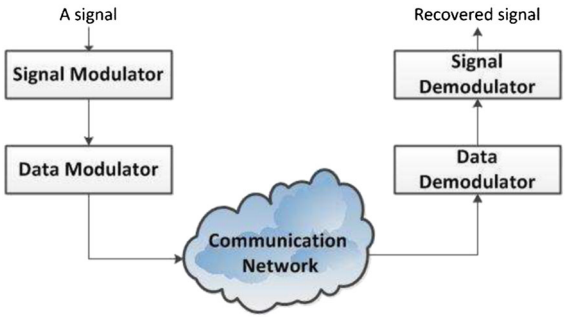


Figure 1. Workflow for watermark encoding and decoding.

Step 1: Signal modulator. The first step in the watermarking process is to apply the PN spreading code, C, to the watermark, x, which is known to both the transmitter and receiver. In this paper, the watermark is constant, and we let $x = +1$. Note that because the watermark is inserted into the data stream at a random time, it is difficult for an adversary to evade detection, even if the watermark is known. More generally, the watermark can be a series of bits, each of which belongs to the set $\{-1, +1\}$. The length of the PN code is L, and its chip duration is t_c seconds per chip. The modulator multiplies x by C to produce the transmitted baseband signal B, such that at any time t, we have

$$B_t = x\,C_t = C_t \tag{1}$$

The modulator also applies an amplification factor of A to the baseband signal so that the resulting spread watermark waveform at time t is $AB_t = AC_t$.

Step 2: Data modulator. The second step embeds the watermarks generated during the preceding step into the meter-reading data stream, X. There are numerous ways to embed watermarks into a data stream; here, we use the following procedure. When a chip in C is -1, we reduce the meter-reading data that occupy that chip's period by A. Likewise, when a chip in C is $+1$, we increase the meter-reading data by A. Note that the meter-reading data stream should be long enough for the detection system to embed watermarks. Hence, the watermarked signal T_t that is the instantaneous manipulated meter-reading data stream can be written as

$$T_t = AC_t + X_t \tag{2}$$

An adversary may attempt to manipulate the meter-reading data stream as it is transmitted through the network. As we stated in Sections 1 and 2, such false data injection attacks can have significant negative consequences for Smart Grid operations. If the meter-reading data stream is manipulated by an adversary, the embedded watermarks will be disrupted, allowing detection of the attack. Note that we are not using the watermarks to modify the transmitting node's packet generation frequency, as was proposed in the original works on this technique [17,18]; we are simply adding the spread watermarks to the original meter measurement data stream. In addition, the watermark is independent of the type of data being transmitted by the meter.

Step 3: Data demodulator. At the utility side, we can formulate the received meter-reading data signal λ_t as

$$\lambda_t = AC_t + X_t \tag{3}$$

The utility captures the meter-reading data stream and divides it into segments, producing a discrete-time signal where t is the time index. Each segment lasts for a chip duration of t_c seconds. Note that we show only the simple non-aggregated case here (i.e., the data come from a single source); we will discuss the details of the data aggregation case in Section 3.3. We assume that X_t varies slowly over a watermark bit interval, and we apply a high-pass filter to

the received signal λ_t to remove the baseband (i.e., roughly constant value) component, which is X_t. Because we are using a high-pass filter, we want to use a PN code that will be minimally distorted by the application of the filter. A PN code with long runs of +1 or –1 terms will produce decaying signals at the filter output. We chose a PN code that is an oscillating series of –1 and +1 terms; this is the most high-frequency digital signal possible and is passed intact by even low-order filters. The filtered received signal λ'_t can be approximately represented as follows:

$$\lambda'_t \approx AC_t + X'_t \tag{4}$$

where $X'_t \approx X_t - m$ and where m is the mean value of the meter-reading data stream X.

Step 4: Signal demodulator. The receiver uses a locally generated copy of the PN code, C', that is identical to the PN code at the transmitter. The receiver uses C' to despread the filtered received signal λ'_t to obtain the received baseband signal. The despreading operation is a second modulation stage, in which we compute the covariance of the filtered signal with C', which gives us

$$\lambda_b = \frac{A}{L}\boldsymbol{C}\cdot\boldsymbol{C}' + \frac{1}{L}\boldsymbol{X}'\cdot\boldsymbol{C}' \tag{5}$$

where $(\cdot)$ is the dot product operator, $\boldsymbol{C}$ is a length-L vector containing the PN code values, and $\boldsymbol{X}'$ is a length-L vector containing the values of X'. If the PN codes are identical and aligned so that $\forall t$, $C_t = C'_t$, we have $(\boldsymbol{C}\cdot\boldsymbol{C}')/L = 1$. Because X' and C' are highly non-correlated and C' is zero-mean, then if the data stream has not been modified, $\lambda_b = A + \varepsilon$, where $\varepsilon = (\boldsymbol{X}'\cdot\boldsymbol{C}')/L$, and $\mathsf{E}[|\varepsilon|] \ll A$. To classify the received signal as +1 or –1, we can use a decision rule. We set a threshold value denoted as T_w for recognizing watermarks. If the strength of the received, despread baseband signal λ_b falls above the threshold value, we decide that the spread watermark is present, and we conclude that the data are intact; otherwise, we decide that the watermark is not intact, and we conclude that the data were manipulated by an adversary.

In order to despread the received watermarked data stream, the locally generated PN code at the receiver needs to be synchronized with the PN code at the transmitter in both frequency and time. The synchronization in frequency can be handled by letting the transmitter and receiver share the same values of parameters (e.g., chip duration and others). To synchronize the two PN codes in time, it is reasonable to assume that both the transmitter and receiver use the standard network time protocols [19] or a precision clock synchronization protocol such as IEEE 1588 [20]. To deal with the misalignment introduced by data stream transmission delays, we adopt the correlation-based scheme discussed earlier and in Section 3.3 to derive the correlation function of the filtered received signal. To this end, we can use a sliding window to move back and forth. In this way, a segment of the measurement stream for the best match can be found. As such, we can determine whether the expected PN code exists in the measurement stream.

3.3. Attack cases

On the basis of the watermarking-based detection scheme discussed in Section 3.2, we now examine how to apply it to detect slow, stealthy false data injection attacks in different cases. As we mentioned in Section 2, both AMI and SCADA systems are composed of three components: the meter (or sensor) at the customer's premises or transmission fields, the aggregator between the meter (or sensor) and the utility provider, and the utility provider itself. All components in the Smart Grid are connected using communication networks such as IP networks. We assume that the utility provider is connected to a separate secured network with low probability of being compromised. Hence, we assume than an adversary can compromise only the meter, sensor, and aggregator. We reiterate that using encryption and decryption techniques cannot resolve the problem that we address in this paper because of the high computational cost and the need to support the large number of legacy sensors and meters already deployed in the power grid. In the following, we discuss several different cases based on which network element is compromised.

Case 1: Compromised aggregator. The aggregator receives watermarked meter-reading data from meters and aggregates multiple meter-reading data steams into a single data flow. If the aggregator is compromised, an adversary can selectively replace the data. We model this effect by defining a parameter $p \in [0, 1]$ to model the severity of the attack; p is the probability that a given data point has been modified. Without loss of generality, the aggregated data from K meters are $\sum_{i=0}^{K}(AC_i + X_i)$, where i is the index of the meters. By using the watermarking scheme, the utility correlates the data received from the aggregator and the correlation result measured by $\mathbf{S}$, which is derived from Equation (5) and is given by

$$\mathbf{S} = \frac{1}{L}(Ax\boldsymbol{C}_i + \boldsymbol{X}_i)\cdot\boldsymbol{C}_i = Ax + \frac{1}{L}\sum_{t=1}^{L} X_{i,t}C_{i,t} \tag{6}$$

where x is the value of the watermark, $X_{i,t}$ is the output from the i^{th} meter at time t, and $C_{i,t}$ is the value of the i^{th} PN spreading code at time t. From the correlation result, the utility can determine whether the aggregated data have been modified.

Case 2: Compromised communication channel. In this case, an adversary compromises the network connecting the aggregator to the utility provider and alters or manipulates the data from the meter. As a result, the watermarks present embedded within the meter-reading data will be altered. Thus, when the utility correlates the watermarked meter-reading data stream with the watermark, it can determine whether the meter-reading data have been modified.

Algorithm 2 Watermark embedding and detection algorithm

Input: Meter Reading: X_t at time t; PN Code: C_t
Parameters: *Amplitude: A; Length of Watermark: L;*
Number of Meters: K; Attack Strength: $p \in [0, 1]$
Output: Similarity Degree: S
if Non-Aggregation **then**
  $W_t \leftarrow A \cdot C_t$
  $T_t \leftarrow X_t + W_t$
  if no attack happened **then**
    $\lambda_t \leftarrow X_t + AC_t$
    $\mathbf{S} \leftarrow \frac{1}{L}\sum_{t=1}^{L} \lambda_t \cdot C_t$
  else if attack happened **then**
    p = rand(0, 1) is the attack severity
    $\lambda'_t \leftarrow X'_t + A \cdot C'_t, t = 1, \ldots, L$
    $\mathbf{S} \leftarrow \frac{1}{L}\sum_{t=1}^{L} \lambda'_t \cdot C_t$
  end if
else if Aggregation **then**
  $T_x \leftarrow \sum_{i=1}^{K} X_{i,t} + A\sum_{i=1}^{K} C_{i,t}$
  if no attack happened **then**
    $\lambda_t \leftarrow \sum_{i=1}^{K} X_{i,t} + A\sum_{i=1}^{K} C_{i,t}$
    $\mathbf{S} \leftarrow \frac{1}{L}\sum_{t=1}^{L} \lambda_{i,t} \cdot C_{i,t}$
  else if attack happened **then** ▷ transmitted signal T_x is replaced by percentage of p
    $\lambda'_t \leftarrow \sum_{i=1}^{K} X'_{i,t} + A\sum_{i=1}^{K} C'_{i,t}$
    $\mathbf{S} \leftarrow \frac{1}{L}\sum_{t=1}^{L} \lambda'_t \cdot C_{i,t}$
  end if
end if

Case 3: Compromised meters or sensors The meter and sensor represent weak points in the Smart Grid. They can be placed in a physically non-secured location or may lack sufficient processing power to run strong security software. Thus, the security of watermarks is critical in our approach. To this end, we can leverage existing techniques such as pre-shared keys [21] to ensure the security of watermarks. For example, the meter or sensor contains an integrated chipset that is pre-programmed when manufactured. The chipset functions as the watermarking facility; the input is the original meter-reading from meter or sensor, and the output is the watermarked data. The chipset contains a confidential pre-shared key. The watermarking function requires two factors, the meter-reading data (the input) and the pre-shared key stored securely in the chipset. Even if the meter or sensor is compromised, the content stored in the chipset cannot be accessed, and the pre-shared key will be secure. In addition, without the pre-shared key, even if the watermarking algorithm is known, an adversary cannot run the watermarking procedure in other physical devices in an attempt to impersonate the meter. Thus, the security of the watermarking process can be ensured. As we stated in Section 2, we assume that there is a low bandwidth secured communication channel between the meters/sensors and the utility provider. A channel of this type can be provided by the existing power grid communication infrastructure (e.g., power line communication).

Note that the algorithm for watermark-based detection is shown in Algorithm 2, which covers both the aggregated data and the non-aggregation data cases; the parameter p is the strength of the attack.

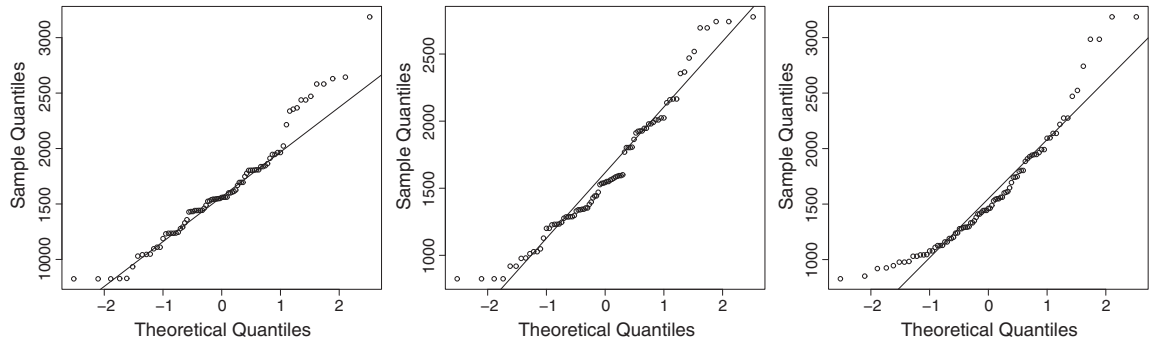
4. MODELING AND ANALYSIS

In this section, we analyze the accuracy of our proposed detection techniques. We first analyze the real-world meter-reading data and show the data can be approximated with a Gaussian distribution. The consumption pattern can be heterogeneous depending on the consumer's habits, and variations in consumer habits may have impact on the detection accuracy. However, examination of the experimental data shows that the Gaussian approximation is valid for individual users. On the basis of the results of this analysis, we define performance metrics for our detection schemes and use them to develop results for both anomaly-based and watermarking-based detection.

4.1. Real-world meter-reading data analysis

Our initial set of real-world metering data is a month's worth of data from a residence in Wodonga, Australia that the owners have made publicly available at the following website: http://www.chookchooks.com/power.html. We analyzed the recorded data and used the hourly power usage as the meter-reading data stream for our experiments. We aggregated the meter-reading data in three time windows: morning (8:00–12:00), afternoon (14:00–18:00), and evening (20:00–24:00). We consider two methods to validate the distribution of meter-reading data. The first method is to conduct a Kolmogorov–Smirnov test on the meter-reading data in each time window. Generally speaking, the Kolmogorov–Smirnov test quantifies the distance between the sample's empirical distribution function and the cumulative distribution function of the reference distribution or between the empirical distribution functions of two samples [22]. In our experiments, we conducted Kolmogorov–Simirnov test with a significance level ($\alpha' = 0.05$) for the meter-reading data in each time window, where α' is the probability that we mistakenly reject the Gaussian distribution hypothesis when it is actually true.

For the Kolmogorov–Simirnov test, we have the following two hypotheses: H_0 (the data have a Gaussian distribution) and H_1 (the data do not have a Gaussian distribution). We compute the *p-value* based on the test statistics and compare it to the threshold α. We can think of α as the probability, given that the null hypothesis H_0 is true, that the test statistic indicates that we should reject H_0. Hence, a greater p-value is a stronger evidence for accepting hypothesis H_0. This indicates that the threshold value of the significance level represents the case, where the null hypothesis H_0 will be accepted for all values of α less than the p-value. In our test, the significance levels that we obtained from the meter-reading data in the morning,



(a) Q-Q plot of morning data. (b) Q-Q plot of afternoon data. (c) Q-Q plot of evening data.

Figure 2. Quantile-Quantile plots of Australian metering data versus Gaussian distribution.

afternoon, and evening windows are 0.3687, 0.0925, and 0.1871, respectively. Because all are greater than $p = 0.05$, we conclude that we can approximate the meter-reading data in all three windows using a Gaussian distribution.

Our second method is to use a Quantile-Quantile (Q-Q) plot to confirm that the distribution of meter data can be approximated by a Gaussian distribution. A Q-Q plot is a graphical method for comparing two probability densities by plotting their quantiles (i.e., cumulative distributions) against each other [23,24]. We select a set of intervals to associate with each quantile. For each quantile, we plot the empirical value (i.e., the proportion of the data that falls within the interval) versus the theoretical value (i.e., the probability that a random variable with the target distribution falls within the interval). Figures 2a, 2b, and 2c show the Q-Q plots of data from the morning, afternoon, and evening windows, respectively. As we can see, the curves are well approximated by straight lines, and this also indicates that we can reasonably model the meter-reading data using the Gaussian distribution.

We also obtained a real-world meter-reading data set from Stanford University that consists of data from various houses over a 200-day period, as described by Houdea *et al.* [12]. We conducted the experiments to validate the statistical distribution of real-world meter-reading data. The experimental results show that the statistical distribution of meter-reading data can be approximated with a Gaussian distribution [25].

4.2. Detection accuracy

To measure the detection accuracy in terms of how reliably a false data injection attack can be detected, we consider two metrics. The first metric is the *detection rate*, P_D, which is defined as the probability of correctly identifying a set of meter-reading data that have been falsely injected. From the defender's perspective, a higher P_D value implies greater detection accuracy. The second metric is the *false positive rate* P_F, which is defined as the probability that an attack is mistakenly detected when none has occurred. From the defender's perspective, a lower value of P_F corresponds to a lower chance of false alarms.

In the following, we first present the analytical results for anomaly-based detection and then show the analytical results for watermarking-based detection. There are some important parameters in our integrated detection system, including the mark data amplitude A, the threshold T_h, the watermark detection threshold T_w, and the PN code length L. Again, we would like to note that in Section 4.1, we conduct analysis on the real-world meter-reading data, showing that the distribution of meter-reading data can be approximated with a Gaussian distribution.

4.3. Anomaly-based detection

We develop a receiver characteristic that shows how to adjust the sensitivity of the anomaly-based detector based on the tradeoff between maximizing the detection rate and minimizing the false positive rate. In our analysis, we rely on the *Central Limit Theorem*, which says that the density of the sum of n independent random variables tends to a normal density as $n \to \infty$. We assume that the readings from the n meters are mutually independent so that we can approximate the distribution of their sum $S_x = \sum_{i=1}^{n} X_i$ with a Gaussian distribution with mean m and standard deviation v. Our real-world meter-reading data analysis confirms this hypothesis as well.

We can let m and v be functions of time in order to account for variations in usage patterns during the day without affecting the following analysis. We assume that an adversary generates false data so that during the attack, the sum of the n meter-readings is also Gaussian but with mean $m + rv$ and standard deviation qv, where the factors $q \in \mathbb{R}^+$ and $r \in \mathbb{R}$ indicate the strength of the attack. Because we do not know q and r *a priori*, we check the value of S_X and compare it to the known mean m. We declare an anomaly if $S_X \notin [m - kv, m + kv]$ (i.e. $|S_X - m| > kv$), where k is a sensitivity parameter.

The detection rate P_D is the probability of successfully detecting an attack, which is the probability that $S_X \sim \mathcal{N}(m + rv, qv)$ and $|S_X - m| > kv$:

$$P_D(k|q,r) = 1 - \frac{1}{qv\sqrt{2\pi}}\int_{m-kv}^{m+kv} \exp\left(\frac{-(u-m-rv)^2}{2q^2v^2}\right) \mathrm{d}u$$
$$= 1 - \Phi\left(\frac{k-r}{q}\right) + \Phi\left(\frac{-k-r}{q}\right) \quad (7)$$

where

$$\Phi(z) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{z} e^{-t^2/2}\mathrm{d}t$$

is the Gaussian cumulative distribution function. Likewise, we can compute the false positive rate P_F, which is the probability of declaring an attack when none occurs (i.e., $S_X \sim \mathcal{N}(m,v)$ and $|S_X - m| > kv$) as

$$P_F(k) = 1 - \int_{m-kv}^{m+kv} \frac{\exp(-(u-m)^2/(2v^2))}{v\sqrt{2\pi}}\,\mathrm{d}u$$
$$= 1 - \Phi(k) + \Phi(-k) \quad (8)$$

We plot $P_F(k)$ versus k in Figure 3. If avoiding false positives is our only performance criterion, then setting $k > 2$ is a sound strategy. However, we gain a low false positive at the price of a low detection rate, so we need to find a value of k that balances the two competing goals. In Figure 4, we plot $P_D(k|q,r)$ versus q and r for three values of k. Decreasing k clearly leads to a high detection rate over a greater portion of the (q,r) space but at the cost of a higher false positive rate. In addition, there are portions of the parameter space, particularly where $|r|$ is large and q is close to unity, where we have a high detection rate regardless of the value of k that we use.

Note that P_F does not depend on q and r, while P_D does. If we can estimate the joint distribution of q and r, $f_{QR}(q,r)$, we can obtain the unconditional detection probability, which is

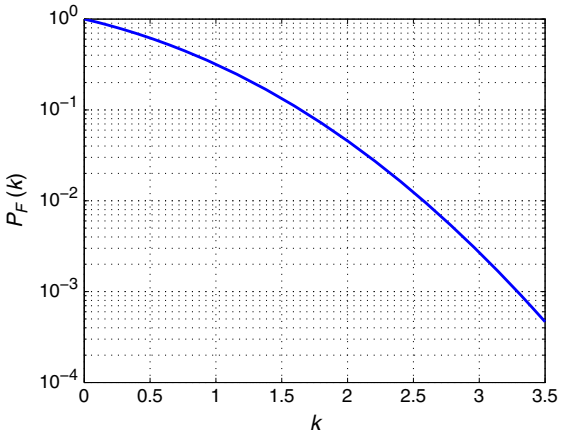


Figure 3. Plot of $P_F(k)$ versus k.

$$P_D(k) = \int_0^\infty\int_0^\infty P_D(k|q,r) f_{QR}(q,r)\,\mathrm{d}q\,\mathrm{d}r \quad (9)$$

In practice, it may not be possible to characterize the adversary's behavior so well that we can obtain $f_{QR}(q,r)$. Still, we can use a receiver plot of $P_D(k|q,r)$ versus $P_F(k)$ to obtain optimal values of k, given that we expect q and r to fall within certain ranges of values. In order to find the operating point of interest (i.e., the value of k), we need to use two cost parameters. These are C_M, the cost of a missed attack detection and C_F, the cost of a false alarm. The maximum risk associated with a detection rule is minimized if we choose an operating point so that P_D and P_F satisfy the following Equation [26]:

$$C_M(1 - P_D) = C_F P_F \quad (10)$$

For our case, given q and r, we must choose k so that

$$P_D(k|q,r) = 1 - \rho P_F(k) \quad (11)$$

where $\rho = C_F/C_M$ is the ratio of the false-alarm and missed-detection costs.

We consider several representative values of ρ. We assume that the cost of a missed attack is greater than that of a false positive, so $\rho < 1$. In Figure 5, we compute the value of k that satisfies Equation (11) versus q and r, where $0 < q \leq 10$ and $-5 \leq r \leq 5$, for two different values of ρ. Several observations result immediately from the figure. First, the sensitivity of k to r decreases with increasing q. Similarly, the sensitivity of k to the value of q is greatest when r is large. For a given pair of attack values (q,r), the required value of k decreases as ρ decreases; this follows because a greater emphasis on preventing misses results in using a lower threshold.

4.4. Watermarking-based detection

In the following, we first analyze the impact of different parameters on the detection accuracy in different scenarios involving watermarking-based detection. On the basis of the analytical results, we discuss how to determine the severity of false data injection attacks, that is, that value of the parameter p, which is the fraction of data that is falsely injected by an adversary.

4.4.1. Non-aggregation case.

Let $\Delta(\boldsymbol{v}_1, \boldsymbol{v}_2) = \sum_{i=1}^{L} v_{1,i}v_{2,i}/L$ be the correlation (i.e., the similarity degree) between the pair of length-L vectors $\boldsymbol{v}_1$, $\boldsymbol{v}_2$. For our analysis, L is the length of the PN code that we use to generate watermarks. Let vector $\boldsymbol{c}_i = \langle c_{i,1}, c_{i,2}, \ldots, c_{i,n}\rangle \in \{-1,+1\}^n$ be the PN code and $\boldsymbol{\omega}_i = \langle \omega_{i,1}, \omega_{i,2}, \ldots, \omega_{i,n}\rangle$ be the meter-reading values after subtracting m, the mean value of the original set of meter-readings. Assume random variables $\omega_{i,1}, \ldots, \omega_{i,n}$ are independent and identically distributed and are drawn from a Gaussian random distribution with

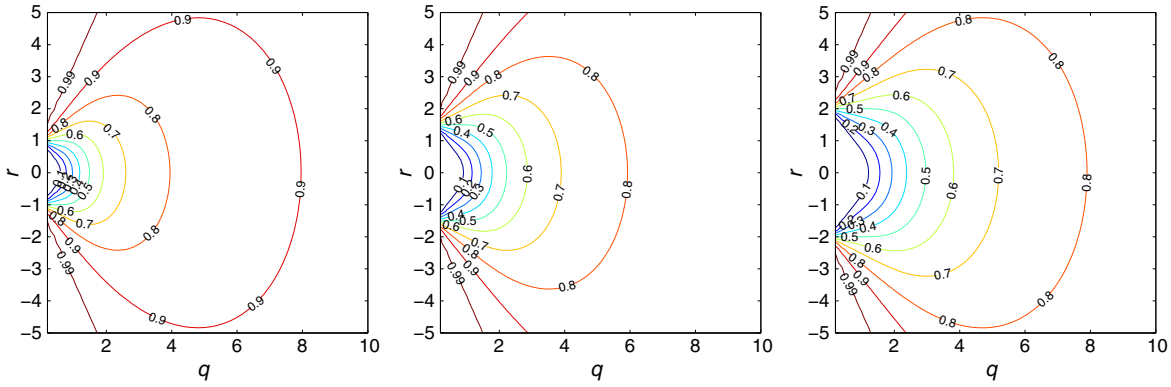


Figure 4. Contour plots showing P_D versus q and r.

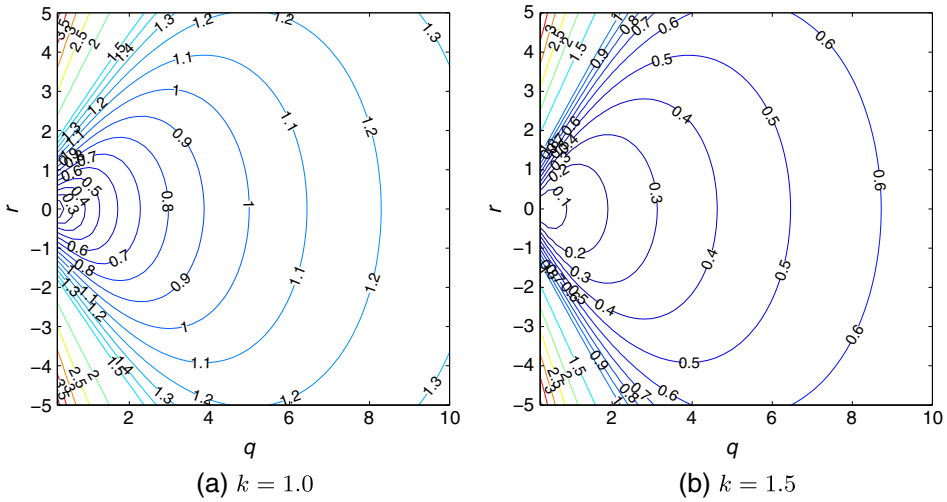


Figure 5. Contour plots showing optimal k-value versus q and r.

zero mean and standard deviation σ_x. Recall that in Section 4.1, we showed that the real-world meter-reading data that we collected can be approximated by a Gaussian distribution.

We assume that the utility provider or other network devices (e.g., aggregators) will conduct the correlation testing using the algorithms described in Section 3.2. Recall that T_w is the watermark detection threshold and A is the mark data amplitude. Also, $\boldsymbol{\lambda}_i$ is the length-n vector of meter-reading data received at the utility. The detection probability P_D is the probability that a watermark can be correctly recognized, and the false positive rate, P_F, is the probability that the defender mistakenly determines that an attack has occurred.

The PN code has low correlation with the (shifted) meter data (i.e., $\mathsf{E}[\Delta(\boldsymbol{\omega}_i, \boldsymbol{c}_i)] = \sum_{j=1}^{L} \mathsf{E}[\omega_{i,j} c_{i,j}]/L \approx 0$); therefore, $\mathsf{E}[\Delta(\boldsymbol{\lambda}_i, \boldsymbol{c}_i)] \approx 0$. If the standard deviation of the meter-reading data is σ_x, the variance of the similarity degree is

$$\begin{aligned}\mathsf{Var}[\Delta(\boldsymbol{\omega}_i, \boldsymbol{c}_i)] &= \mathsf{E}\left[(\Delta(\boldsymbol{\omega}_i, \boldsymbol{c}_i) - 0)^2\right] \\ &= \frac{1}{L^2} \sum_{j=1}^{L} \sum_{k=1}^{L} \mathsf{E}[\omega_{i,j}\, \omega_{i,k}] \underbrace{\mathsf{E}[c_{i,j}\, c_{i,k}]}_{=\delta_{jk}} \\ &= \frac{1}{L^2} \sum_{j=1}^{L} \mathsf{E}[\omega_{i,j}^2] = \frac{\sigma_x^2}{L}\end{aligned} \tag{12}$$

where δ_{jk} is the Kronecker delta. Therefore, $\Delta(\boldsymbol{\lambda}_i, \boldsymbol{c}_i) \sim \mathcal{N}(0, \sigma_x^2/L)$, approximately. Thus, we can compute P_D and P_F, respectively, as

$$\begin{aligned}P_D &= 1 - \mathsf{Pr}\{\Delta(\boldsymbol{\lambda}_i, \boldsymbol{c}_i) \le T_w \,|\, \boldsymbol{\lambda}_i = \boldsymbol{\omega}_i + A\boldsymbol{c}_i\} \\ &= 1 - \mathsf{Pr}\{\Delta(\boldsymbol{\omega}_i, \boldsymbol{c}_i) \le T_w - A\} \\ &= 1 - \tfrac{1}{2}\,\mathrm{erfc}\left(\tfrac{A - T_w}{\sigma_x}\sqrt{\tfrac{L}{2}}\right)\end{aligned} \tag{13}$$

and

$$P_F = \mathrm{Pr}\{\Delta(\boldsymbol{\lambda}_i, \boldsymbol{c}_i) \geq T_w \mid \boldsymbol{\lambda}_i = \boldsymbol{\omega}_i\} = \tfrac{1}{2}\,\mathrm{erfc}\left(\frac{T_w}{\sigma_x}\sqrt{\frac{L}{2}}\right) \tag{14}$$

where

$$\mathrm{erfc}(z) = \frac{2}{\sqrt{\pi}}\int_z^\infty e^{-t^2}\,\mathrm{d}t$$

is the complementary error function. Note that Equation (14) shows that the largest value that P_F can take is 1/2.

We can examine the performance of a given watermark detection scheme by plotting P_D versus P_F to produce a ROC, as shown in Figure 6, where we present a set of ROC curves in each of the two subfigures. To produce each ROC, we vary only the threshold value, T_w. In Figure 6a, we set $L = 8$ and plot ROCs for three values of A: $\sigma_x/10$, $\sigma_x/2$, and σ_x. In Figure 6b, we use the same three values of A but use $L = 32$. Comparing Figure 6a to 6b, we observe that increasing the PN code length, L, results in a ROC that is closer to the ideal performance curve. Also, increasing the watermark amplitude, A, improves the watermark detector's performance. We observe that Figure 6b shows that using a large value of A (roughly equal to the standard deviation of the metering data during the watermark period) results in a nearly ideal ROC for the detector, even though the PN code length is rather short. This would seem to indicate that the utility should use very large-amplitude watermarks; however, as we explain later, that approach would make it easier for an adversary to identify when a watermark is being inserted into the data stream.

Given values for A, σ_x, and L, we can determine the value of T_w that is optimal in the sense that it corresponds to the point on the ROC that is closest to (0, 1), which is the performance point for an ideal detector. Let A and T_w be multiples of σ_x: $A = \alpha\sigma_x$ and $T_w = \beta\sigma_x$. The squared distance from a point on the ROC to the point (0, 1) is $D = P_F^2 + (P_D - 1)^2$, which we can write as

$$D = \tfrac{1}{4}\,\mathrm{erfc}^2\left(\beta\sqrt{\frac{L}{2}}\right) + \tfrac{1}{4}\,\mathrm{erfc}^2\left((\alpha-\beta)\sqrt{\frac{L}{2}}\right) \tag{15}$$

The derivative of D in Equation (15) with respect to β is

$$\frac{\mathrm{d}D}{\mathrm{d}\beta} = \sqrt{\frac{L}{2\pi}}\left[e^{-(\alpha-\beta)^2L/2}\,\mathrm{erfc}\left((\alpha-\beta)\sqrt{\frac{L}{2}}\right) - e^{-\beta^2L/2}\,\mathrm{erfc}\left(\beta\sqrt{\frac{L}{2}}\right)\right] \tag{16}$$

If we set $\beta = \alpha/2$ in Equation (16), we have $\mathrm{d}D/\mathrm{d}\beta = 0$. To check that this value of β minimizes D, we compute $\mathrm{d}^2D/\mathrm{d}\beta^2$ and evaluate it when $\beta = \alpha/2$, giving

$$\left.\frac{\mathrm{d}^2D}{\mathrm{d}\beta^2}\right|_{\beta=\alpha/2} = \frac{2L}{\pi}e^{-\alpha^2L/4} + \alpha\sqrt{\frac{L^3}{2\pi}}\,e^{-\alpha^2L/8}\,\mathrm{erfc}\left(\frac{\alpha}{2}\sqrt{\frac{L}{2}}\right) \tag{17}$$

Because $\alpha > 0$, $\mathrm{d}^2D/\mathrm{d}\beta^2 > 0$ in Equation (17); therefore, $T_w = A/2$ is the detection threshold that produces the best performance.

We can see the effect of varying L and A in Figure 7, in which we plot P_F and P_D versus L for two values of A, with $T_w = A/2$ for both values. Figure 7 shows that not only does increasing L improve the detector's performance but also that the detector's sensitivity to the watermark amplitude, A, depends strongly on the value of L. For short PN codes, a large increase in A is required to produce a significant improvement in performance, whereas very large improvements result from modest increases in the value of A when we use longer codes. In addition, the figure shows the relationship between P_F and P_D for the special case, where the detector threshold is optimized ($T_w = A/2$); recalling Equations (13) and (14), we obtain $P_D(L) = 1 - P_F(L)$.

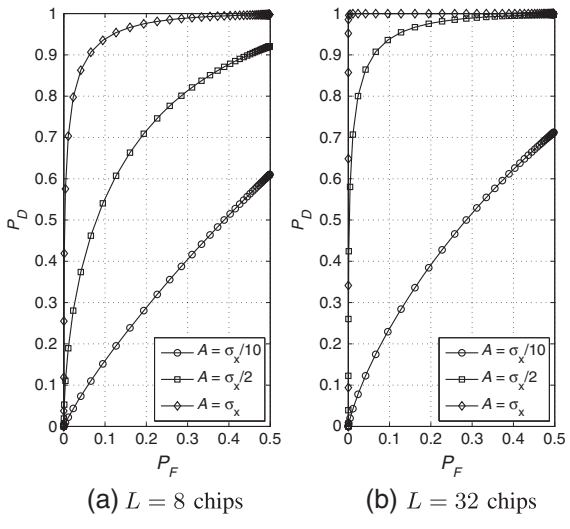


Figure 6. Receiver operating characteristics for various values of A, using both small and large pseudorandom noise code lengths.

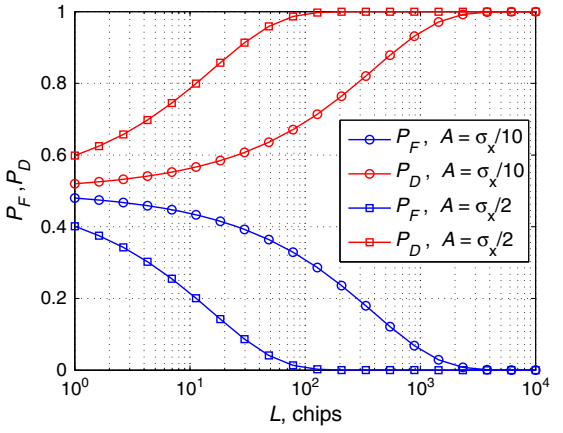


Figure 7. Plot of P_F and P_D versus L with $T_w = A/2$, where $A = \sigma_x/10$ and $A = \sigma_x/2$.

In other words, the probability of a false alarm for the optimal detector is equal to the probability of a missed detection.

4.4.2. Aggregation case.

Assume the aggregator combines readings from K meters through summation and forwards the results to the utility. We have $\{\boldsymbol{\omega}_i\}_{i=1}^K$, the set of K meter-reading vectors $\{\boldsymbol{X}_i\}_{i=1}^K$ that have had their respected mean values subtracted ($\boldsymbol{\omega}_i = \boldsymbol{X}_i - m_i$) and $\{\boldsymbol{c}_i\}_{i=1}^K$, the corresponding set of K PN codes. Without loss of generality, we assume the data in each of the K meter-reading streams are normally distributed with mean zero and variance σ_x and that the data in any steam is independent of the data in the other $K-1$ streams. Also, we assume that all K PN codes are mutually orthogonal (i.e., $\Delta(\boldsymbol{c}_i, \boldsymbol{c}_j) = \delta_{ij}$). After applying watermarks to each of the streams, the utility receives the aggregated data vector

$$\boldsymbol{\lambda} = \sum_{i=1}^{K} \boldsymbol{\omega}_i + A \sum_{i=1}^{K} \boldsymbol{c}_i \tag{18}$$

Consider the j^{th} data stream, $1 \leq j \leq K$. To detect the presence of a watermark bit, the utility first applies the j^{th} PN code $\boldsymbol{c}_j$ by computing the similarity measure $\Delta(\boldsymbol{\lambda}, \boldsymbol{c}_j)$, which we can expand as follows:

$$\begin{aligned} \Delta(\boldsymbol{\lambda}, \boldsymbol{c}_j) &= \textstyle\sum_{i=1}^{K} \Delta\left(\boldsymbol{\omega}_i, \boldsymbol{c}_j\right) + A \sum_{i=1}^{K} \Delta\left(\boldsymbol{c}_i, \boldsymbol{c}_j\right) \\ &= A + \textstyle\sum_{i=1}^{K} \Delta\left(\boldsymbol{\omega}_i, \boldsymbol{c}_j\right) \end{aligned} \tag{19}$$

From the development for the non-aggregated case, $\Delta\left(\boldsymbol{\omega}_i, \boldsymbol{c}_j\right) \sim \mathcal{N}(0, \sigma_x^2/L)$, $1 \leq i \leq K$. Furthermore, because it is the sum of K independent and identically distributed normal random variables, $\sum_{i=1}^{K} \Delta\left(\boldsymbol{\omega}_i, \boldsymbol{c}_j\right) \sim \mathcal{N}(0, K\sigma_x^2/L)$. Analogous to the non-aggregated case, if we use a decision threshold T_w, the detection probability is

$$\begin{aligned} P_D &= 1 - \mathsf{Pr}\left\{\textstyle\sum_{i=1}^{K} \Delta(\boldsymbol{\omega}_i, \boldsymbol{c}_j) \leq T_w - A\right\} \\ &= 1 - \tfrac{1}{2}\,\mathrm{erfc}\left(\tfrac{A - T_w}{K\sigma_x}\sqrt{\tfrac{L}{2}}\right) \end{aligned} \tag{20}$$

and the false positive rate P_F is

$$P_F = \mathsf{Pr}\left\{\textstyle\sum_{i=1}^{K} \Delta(\boldsymbol{\omega}_i, \boldsymbol{c}_j) \geq T_w\right\} = \tfrac{1}{2}\,\mathrm{erfc}\left(\tfrac{T_w}{K\sigma_x}\sqrt{\tfrac{L}{2}}\right) \tag{21}$$

From the development of Equations (20) and (21), it follows that the optimal threshold is still $T_w = A/2$. Note that the increase in the noise power by a factor of K means that the watermark amplitude A must be increased by K as well in order for the detector to perform as well as a detector operating on a single data stream.

4.5. Estimating the severity of data injection attacks

In addition to detecting false data injection attacks, our approach can estimate the severity of these attacks, that is, the proportion of data being manipulated or injected by an adversary. We define the attack severity as $p \in [0, 1]$, which is the probability that the meter-reading data have been modified. Assume that we have a non-aggregated stream of meter-reading data where the vector of values with the mean subtracted is $\boldsymbol{\omega}_i = \langle \omega_{i,1}, \omega_{i,2}, \ldots, \omega_{i,L} \rangle$ and the watermarks are $\boldsymbol{c}_i = \langle c_{i,1}, c_{i,2}, \ldots, c_{i,L} \rangle$, respectively. Let p be the fraction of data points in the received, watermarked data stream that an adversary has manipulated. The received, manipulated vector is $\boldsymbol{\lambda}'_i = \boldsymbol{\omega}'_i + A\boldsymbol{c}'_i$. Without loss of generality, we sort the manipulated meter data and PN vectors so that the modified elements occupy the first pL positions in each vector:

$$\boldsymbol{\omega}'_i = \langle \omega'_{i,1}, \omega'_{i,2}, \ldots, \omega'_{i,pL}, \omega_{i,pL+1}, \ldots, \omega_{i,L} \rangle \tag{22}$$

$$\boldsymbol{c}'_i = \langle c'_{i,1}, c'_{i,2}, \ldots, c'_{i,pL}, c_{i,pL+1}, \ldots, c_{i,L} \rangle \tag{23}$$

We assume that the manipulated PN code chips are independent of the unaffected PN chips so that $\mathsf{E}[c'_{i,n} c_{i,n}] = \mathsf{E}[c'_{i,n}]\mathsf{E}[c_{i,n}] = 0$ for $n = 1, 2, \ldots, pL$. The similarity measure computed by the receiver is

$$\begin{aligned} \Delta(\boldsymbol{\lambda}_i, \boldsymbol{c}_i) &= \Delta(\boldsymbol{\omega}'_i, \boldsymbol{c}_i) + A\Delta(\boldsymbol{c}'_i, \boldsymbol{c}_i) \\ &= \frac{1}{L} \textstyle\sum_{n=1}^{L} \omega'_{i,n}\, c_{i,n} \\ &\quad + \frac{A}{L}\left[\textstyle\sum_{n=1}^{pL} c'_{i,n}\, c_{i,n} + \sum_{n=pL+1}^{L} c_{i,n}^2\right] \end{aligned} \tag{24}$$

Because $\boldsymbol{\omega}'_i$ and $\boldsymbol{c}_i$ are non-correlated, the mean value of $\Delta(\boldsymbol{\omega}'_i, \boldsymbol{c}_i)$ is close to zero; from Equation (24), $\mathsf{E}\{\Delta(\boldsymbol{c}'_i, \boldsymbol{c}_i)\} = 1 - p$. Hence, we have

$$\hat{p} = 1 - \frac{\Delta(\boldsymbol{\lambda}_i, \boldsymbol{c}_i)}{A} \tag{25}$$

as our estimate of the attack severity.

5. PERFORMANCE EVALUATION

In this section, we conduct the performance evaluation of our proposed integrated detection techniques using real-world meter-reading data. Note that our detection techniques are generic and can be applied to detect false data from both meters in AMI networks and sensors in SCADA networks. Unfortunately, to the best of our knowledge, there is no sensor data set in SCADA available to the public, and so, we leave it in our future work. Given the tight timing constraints in SCADA system, the limited computational resources in sensors, and the constraints imposed by the large population of legacy sensors in the power grid,

we expect that the detection techniques we have developed, such as watermark-based detection, can effectively deal with false data injection attacks in SCADA at low cost. In the following, we introduce the evaluation methodology and then present the evaluation results.

5.1. Methodology

We simulated our approach using Matlab 7.9.0 [MathWorks (Corporate Headquarters) 3 Apple Hill Drive Natick, MA 01760-2098, UNITED STATES][‡]. We use the meter-reading data that we collected from http://www.chookchooks.com, to simulate the possible scenarios in the Smart Grid. The webpage provides local power usage data over a 1-month period for a home in Wodonga, Australia. We tested the anomaly-based detection using both the basic data stream and a modified version that incorporates the effects of a simulated attack, and we calculated the relevant statistics and the detection rate. For the watermarking-based detection, we calculated the similarity degree **S** in each scenario using Equation (6).

In our evaluation, we consider three parameters: the amplitude of the watermark A, the length of the watermark L, and the severity of attack p, where $p \in [0, 1]$. Note that $p = 0$ means that there is no attack and the received signal is same as the transmitted signal; $p = 1$ means the attack is strong enough to affect the entire transmitted signal. We used the m-sequence generation program in Matlab to generate the sequences of watermark bits. We evaluated our approach in two cases: *no aggregation* and *aggregation*. In each case, we analyzed the behavior of the similarity degree **S** with respect to the various parameters. For each case, we kept two of the three parameters fixed and analyzed the dependence of the **S** on the remaining parameter. To evaluate the performance of our detection techniques, we use P_D and P_F as metrics, which we defined in Section 4.2.

5.2. Evaluation results

Figure 8 shows the value of P_F for different thresholds; a lower threshold value results in a high value of P_F, although the rate of decrease of P_F drops off considerably for $T_h > 2.5$, indicating diminishing returns for higher values of the threshold. Figure 9 shows the relationship between the detection rate P_D and the attack strength for different values of the detection threshold in anomaly-based detection. In the figure, $k = 0.5, 1, 1.5$ are parameters to control thresholds T_h and T_l discussed in Section 3.1. We consider the attack to manipulate the meter readings by

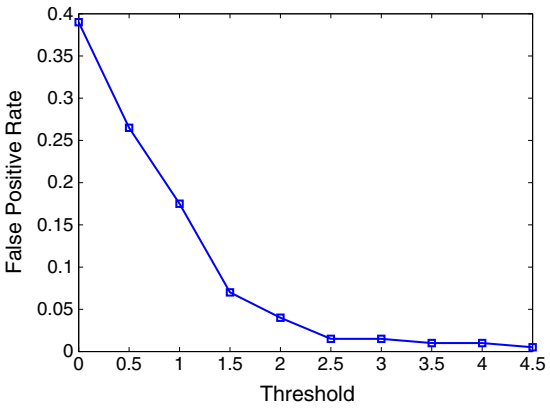


Figure 8. False positive rate versus threshold T_h.

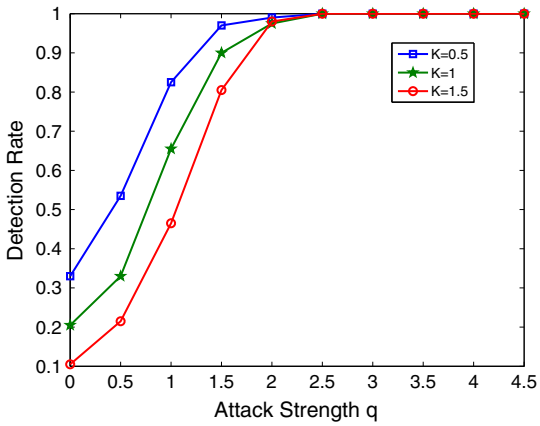


Figure 9. Detection rate versus attack strength.

adding $A_q = q^*\sigma$ to the normal meter reading, where q is defined as the attack strength that the data are manipulated, and σ is the standard deviation of the meter-reading data. From this figure, we can see that our detector's ability to identify attacks decreases with decreasing attack strength, although we are able to achieve better than 80% detection probability when the attack strength is greater than 1.5. In addition, a higher detection rate can be achieved by reducing threshold value. We conclude that if we establish a sufficiently low threshold, we can effectively detect attacks but at the price of a high false positive rate. Note in the figure that setting the threshold to 0.5 gives a detection rate of 30% when the attack strength is zero, which is a clear indication of a high rate of false positives. Hence, as we showed in Section 4.3, an optimal threshold value should be selected to balance the attack detection rate against and false positive rate; the exact tradeoff will depend on the utility operator's criteria.

Figure 10 illustrates the similarity degree **S** as a function of the watermark length L for several values of the watermark amplitude A when there is no attack. The figures show that increasing the amplitude increases the similarity degree, although **S** reaches an asymptote equal to A once L reaches a value between 200 and 300 chips. As we would

[‡] Certain commercial equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

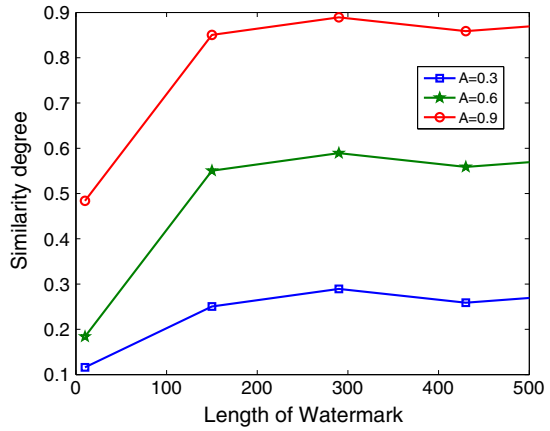


Figure 10. Similarity degree versus length of watermark.

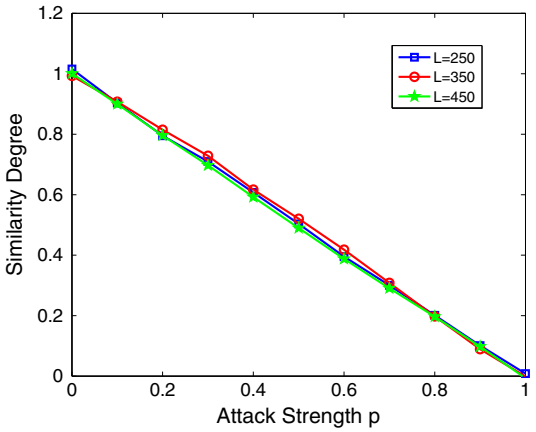


Figure 12. Similarity degree versus severity of attack.

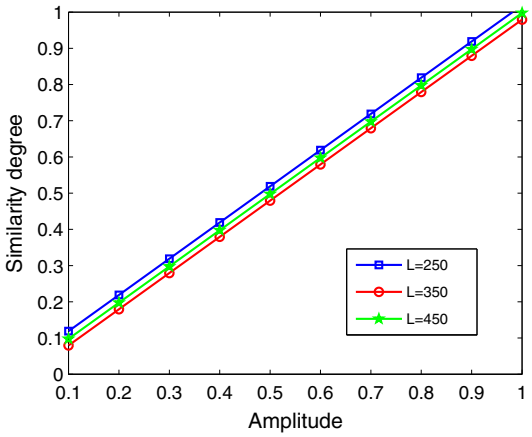


Figure 11. Similarity degree versus watermark amplitude.

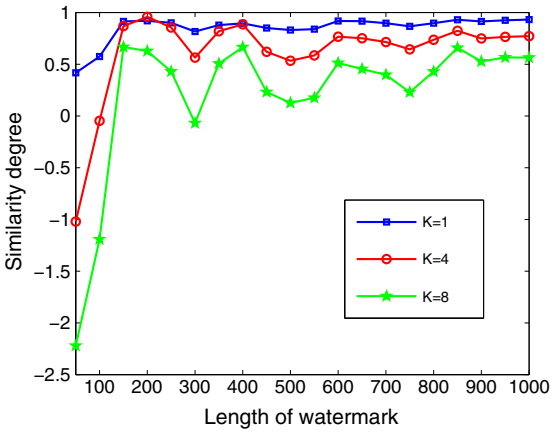


Figure 13. Similarity degree versus watermark length for aggregation case.

expect, the similarity degree shows the same pattern for each value of the amplitude that we considered.

Figure 11 illustrates the relationship between the similarity degree and the amplitude of the watermark in terms of different PN code lengths, for the case where there is no attack. The similarity degree increases linearly with respect to the watermark amplitude. Hence, this further validates that, when there is no attack, the similarity degree should approximate the amplitude of the watermark, independent of the length of the watermark.

Similarly, Figure 12 shows the relationship between the similarity degree and the attack strength in terms of different watermark lengths for the non-aggregation scenario. In this case, we set the watermark amplitude to unity. Note that attack strength, $p \in [0, 1]$, is defined as the ratio that the data are manipulated. For example, when $p = 0.4$, there is 40% chance that transmitted meter readings will be manipulated. As we can see, when the attack strength is zero (i.e., when there is no attack), the similarity degree is one, which indicates that the received signal was not modified. When the attack is strong enough to replace the transmitted signal completely (i.e., when strength of the attack is one), the similarity degree becomes zero (i.e., the received signal was completely changed). We observe similar behavior for $L = 350$ chips and $L = 450$ chips; a shorter code actually yields higher values for $\mathbf{S}$.

Figure 13 illustrates the relationship between the similarity degree and the watermark length in terms of K, the number of meter-reading streams in an aggregation scenario. In the experiments, we choose watermark amplitude as 0.3 as default. As we can see from this figure, as K increases, there is a corresponding increase in the variance of $\mathbf{S}$, because $\sum_{i=1}^{K} \Delta\left(\boldsymbol{\omega}_i, \boldsymbol{c}_j\right) \sim \mathcal{N}(0, K\sigma_x^2/L)$. This increase in noise results in greater difficulty in detecting the watermark. The figure also shows that increasing L compensates for the increased noise, although even with large L, there is a reduction in $\mathbf{S}$ as K increases, indicating that increasing A may also be necessary if the utility is aggregating metering data. These results show that the similarity degree is a function of watermark length and that for a given length of watermark, we obtain worse performance with data aggregation than with non-aggregation.

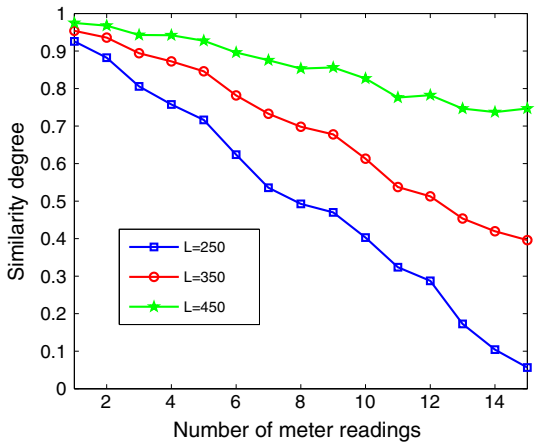


Figure 14. Similarity degree versus number of aggregated meters.

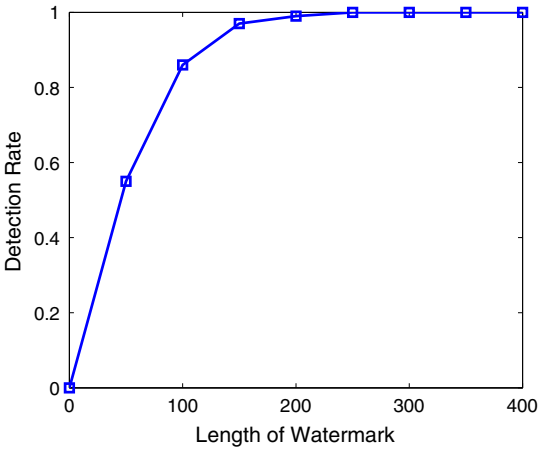


Figure 16. Detection rate versus watermark length (using time hopping spread spectrum).

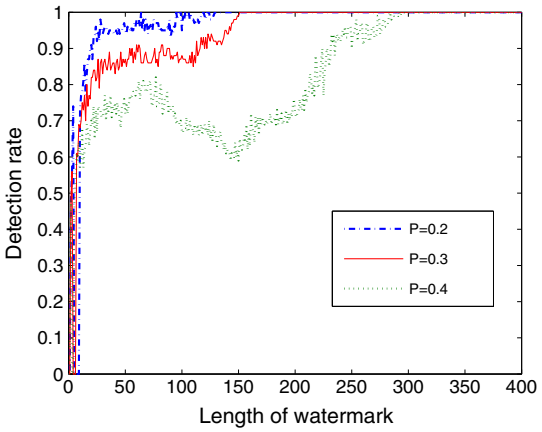


Figure 15. Detection rate versus watermark length.

Figure 14 illustrates the relationship between the similarity degree and the number of aggregated meter-reading data streams, K, for various PN code lengths. As Figure 13 shows, increasing K results in a decrease in $\mathbf{S}$, which can be quite severe. degree will decrease. However, the rate at which $\mathbf{S}$ decreases is a function of L, and even modest increases in the PN code length can significantly improve the detector performance.

Figure 15 illustrates the relationship between P_D, the detection rate, and the number of aggregated meter-reading data streams, K, for various values of p, the attack strength. The figure shows that when $p < 0.2$, a short watermark ($L < 50$ chips) gives good performance ($P_D > 0.9$), with P_D close to unity for $L > 150$ chips. Naturally, stronger attacks require longer PN codes to provide good detection rates; if $p = 0.4$, we need $L > 300$ chips for P_D to be close to unity.

To reduce the probability of interception and recognition of the watermark process, we also could leverage the time hopping spread spectrum (THSS) technique [27] and simulated the time hopping technique. We first generated a PN code and then used an additional THSS technique to vary the inter-chip intervals in the PN code. At the receiver side, we used the same THSS code to recover the PN code chips and then conducted the similarity test. Figure 16 illustrates the relationship between the detection rate P_D and L, the length of the watermark. For the scenario in Figure 16, we set the attack strength to $p = 0.3$. As we can see from the figure, the performance of the THSS scheme is similar to that of standard watermarking-based detection. The difference is that the data modulator process will take a longer time to spread signal bits if we use THSS. Correspondingly, the receiver needs to observe the transmitted meter-reading data over a longer time window to detect an attack.

6. DISCUSSION

In this section, we discuss some issues related to false data injection attacks against various parts of the Smart Grid.

(1) Energy transmission. Lin *et al.* [9] investigated the vulnerabilities of distributed energy routing and transmission and showed that it can be disrupted by false data injection attacks. However, how to defend against those attacks still remains as an open issue. Our integrated detection approach can be applied to detect falsified energy data and reduce the impact of injection attacks on the energy distribution system. For example, when an adversary launches a rapid attack by manipulating the energy demand significantly in order to disrupt the energy routing process, our proposed anomaly-based detection scheme can detect the attack and prevent the disruptive routing updates that would be caused by such an attack. When an adversary launches a stealthy attack by performing subtle manipulations of the energy demand data over a long period, our proposed watermarking-based detection scheme would be able to detect this type of attack.

(2) Attacks against process control systems. Alvaro *et al.* [28] introduced an anomaly-based attack detection scheme for general control systems, of which Smart Grid systems such as SCADA are a subset. In their approach, one creates a model of the system of interest that allows one to see what output signals the system will produce in response to various input or feedback signals. An attack on the system can be detected by comparing the expected output from the model with the received (possibly compromised) output data from the actual system. This detection scheme works well for large-scale and short-duration attacks. However, when an adversary launches a stealthy attack lasting for a relatively long time, the detection system becomes less efficient. By integrating anomaly-based detection with our watermarking-based detection scheme, we can defend against rapid attacks as well as slow, stealthy attacks against data-driven systems, including process control systems.

(3) Power system state estimation. Liu *et al.* [8] considered state estimators in electrial networks and examined the effect of several types of false data injection attacks that can bypass existing schemes for detecting bad data. Nevertheless, the detection of such stealthy attacks remains an open issue. Our proposed attack detection scheme can estimate the strength of attacks, in addition to detecting various attack types.

7. RELATED WORK

The Smart Grid leverages advanced communication technologies to make the power grid efficient and reliable. Nevertheless, it has been shown that new technologies (i.e., smart meters and sensors) can increase the vulnerability of the power grid to cyber attacks [29,30]. For example, the false data injection attack is one of the threats that affect the operation of the Smart Grid. An adversary can in theory cripple the power grid by attacking the energy management system [31].

A number of false data attack and detection schemes have been proposed [2,8,32–41]. For example, Liu *et al.* [8] showed that an adversary who knows the knowledge of the network could inject false data, bypass the bad data detection, and ultimately manipulate the network's state estimation. Mo and Sinopoli [2] investigated the effects of false data injection attacks and developed criteria to improve system resilience. Yi *et al.* [40] studied an adaptive cumulative sum algorithm to detect attacks at the network control center. In this paper, we have developed an integrated detection system that combines both anomaly-based and watermark-based detection to deal with both strong and stealthy false data injection attacks in the Smart Grid. Through both theoretical analysis and evaluation, we also investigate the effectiveness of our proposed detection schemes.

Watermarking has been widely used to determine whether a set of data are genuine [42–45]. For example, Feng and Potkonjak [42] developed the watermarking techniques using cryptographically encoded authorship signatures. Kamel and Juma [44] developed a lightweight watermarking technique for networks of wireless sensors that have limited power. In contrast to these efforts, our work uses watermarking as a defense strategy to detect stealthy attacks in the Smart Grid.

8. CONCLUSION

In this paper, we investigated two types of strategies to defend against false data injection attacks. One is to apply a statistical anomaly-based detection technique to defeat intense and short-duration attacks. The other is to apply a watermarking-based detection technique to defeat slow and stealthy attacks. Our experimental data show that our integrated defense system can accurately detect both strong and stealthy attacks. Our data show that our anomaly-based detection technique can detect attacks accurately when the attacker changes up to 25% of the meter-reading data and the watermarking-based detection technique can detect stealthy attacks and accurately estimate the attack strength. An interesting direction for future research lies in identifying the origination point of the attack and which network elements are affected.

ACKNOWLEDGEMENTS

This work was supported in part by the US National Science Foundation under grant CNS 1117175. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

1. Mo Y, Kim T, Brancik K, Dickinson D, Lee H, Perrig A, Sinopoli B. Cyber-physical security of a Smart Grid infrastructure. *Proceedings of the IEEE* January 2012; **100**(1): 1–15.
2. Mo Y, Sinopoli B. False data injection attacks in control systems, *Preprints of the 1st workshop on Secure Control Systems*, CPSWEEK, Stockholm, Sweden, 2010; 56–62.
3. McLaughlin S, Podkuiko D, McDaniel P. Energy theft in the advanced metering infrastructure. *Critical Information Infrastructures Security of Lecture Notes in Computer Science* 2010; **6027**: 176–187.
4. Cleveland F. Cyber security issues for advanced metering infrastructure (AMI), *Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, Pittsburgh, PA, USA, 2008; 1–5.
5. Song K, Seo D, Park H, Lee H, Perrig A. Omap: one-way memory attestation protocol for smart meters, *Proceedings of 2011 Ninth IEEE International Symposium on Parallel and Distributed Processing with*

Applications Workshops (ISPAW), Busan, South Korea, 2011; 111–118.
6. Karnouskos S. *Stuxnet Worm Impact on Industrial Cyber-Physical System Security*, 2011.
7. Mashima D, Cardenas A A. Evaluating electricity theft detectors in Smart Grid networks, *Proceedings of the 15th International Conference on Research in Attacks, Intrusions, and Defenses (RAID)*, Amsterdam, The Netherlands, 2012; 210–229.
8. Liu Y, Ning P, Reiter M. False data injection attacks against state estimation in electric power grids, *Proceedings of the 16th ACM Conference on Computer and Communications Security,* ACM, Chicago, IL, USA, 2009; 21–32.
9. Lin J, Yu W, Yang X, Xu G, Zhao W. On false data injection attacks against distributed energy routing in Smart Grid, *Proceedings of ACM/IEEE Third International Conference on Cyber-Physical Systems (ICCPS)*, Beijing, China, 2012; 1–10.
10. Wikipedia. *Northeast blackout of 2003*. (Available from: http://en.wikipedia.org/wiki/Northeast Blackout of 2003]/) [Accessed on 19 September 2011].
11. Yuan Y, Li Z, Ren K. Modeling load redistribution attacks in power systems. *IEEE Transactions on Smart Grid* 2011; **2**(2): 382–390.
12. Houdea S, Todd A, Sudarshan A, Flora JA, Armel KC. Real-time feedback and electricity consumption: a field experiment assessing the potential for savings and persistence. *The Energy Journal. International Association for Energy Economics, HighBeam Research* 2013: 87–103, (Available from: http://www.stanford.edu/group/peec/cgi-bin/docs/behavior/research/FieldExperimentPowermeter_vrevised_May2012_authors_vf_pdf.pdf).
13. Stallings W. *Network Security Essentials: Applications and Standards*, Vol. 2. Prentice Hall: Englewood Cliffs, NJ, USA, 2007.
14. Abur A, Exposito A. *Power System State Estimation: Theory and Implementation*, Vol. 24. CRC, 2004.
15. *Intrusion detection system*, 2013. (Available from: http:// webopedia.com/TERM/I/intrusion_detection_system. html).
16. Garcia-Teodoro P, Diaz-Verdejo J, Macia-Fernandez G, Vazquez E. Anomaly-based network intrusion detection: techniques, systems and challenges. *Computers & Security* 2009; **28** (1–2): 18–28.
17. Yu W, Fu X, Graham S, Xuan D, Zhao W. DSSS-based flow marking technique for invisible traceback, *Proceedings of IEEE Symposium on Security and Privacy*, Berkeley, CA, USA, 2007; 18–32.
18. Wang X, Chen S, Jajodia S. Network flow watermarking attack on low-latency anonymous communication systems, *IEEE Symp. on Security and Privacy*, Berkeley, CA, USA, May 2007; 116–130.
19. Mills D, Martin (Ed), J, Burbank J, Kasch W. Network time protocol version 4:protocol and algorithms specification*, RFC 5905*, June 2010. (Available from: http://www.rfc-editor.org/rfc/rfc5905.txt).
20. IEEE standard for a precision clock synchronization protocol for networked measurement and control systems, *IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002)*, 2008; c1–269.
21. Jin T, Noubir G, Thapa B. Zero pre-shared secret key establishment in the presence of jammers, *Proceedings of the Tenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, New Orleans, LA, USA, 2009; 219–228.
22. Duda R, Hart P, Stork D. *Pattern Classification*, ser. Pattern Classification and Scene Analysis: Pattern Classification. John Wiley & Sons, Inc: Hoboken, NJ, USA, 2000.
23. Caton PW, Nayuni NK, Murch O, Corder R. Endotoxin induced hyperlactatemia and hypoglycemia is linked to decreased mitochondrial phosphoenolpyruvate carboxykinase. *Life Sciences* 2009; **84**: 738–744.
24. Ben MG, Yohai VJ. Quantile-quantile plot for deviance residuals in the generalized linear model. *Journal of Computational and Graphical Statistics* 2004; **13**: 36–47.
25. Yu W, An D, Griffith D, Yang Q, Xu G. On statistical modeling and forecasting of energy usage in Smart Grid. *Technical Report 2013-09-001*, Department of Computer and Information Science, Towson University, 2013.
26. Van Trees H. *Detection, Estimation, and Modulation Theory, Part I*. John Wiley and Sons, Inc: Hoboken, NJ, USA, 1968.
27. Smillie G. *Analogue and Digital Communication Techniques*. Arnold: London, 1999.
28. Cárdenas AA, Amin S, Lin Z-S, Huang Y-L, Huang C-Y, Sastry S. Attacks against process control systems: risk assessment, detection, and response, *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, New York, NY, USA, 2011; 355–366.
29. Cisco. *Smart Grid Security Solutions Brief*, 2009. (Available from: http://cisco.com/web/strategy/docs/94energy/CiscoSmartGridSecurity_solutions_brief_c22-95556936.pdf).
30. Wang W, Lu Z. Cyber security in the Smart Grid: survey and challenges. *Computer Networks* 2013; **57** (5): 1344–1371.
31. Kosut O, Jia L, Thomas R, Tong L. On malicious data attacks on power system state estimation, *Pro-*

ceedings of 45th IEEE Universities Power Engineering Conference (UPEC), Cardiff, United Kingdom, 2010; 1–6.
32. Xie L, Mo YL, Sinopoli B. False data injection attacks in electricity markets, *Proceedings of 1st IEEE International Conference on Smart Grid Communications*, Gaithersburg, MD, USA, October 2010; 226–231.
33. Jia L, Thomas RJ, Tong Eds., L. Malicious data attack on real-time electricity market, *Proceedings of 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011.
34. Akella R, McMillin BM. Information flow analysis of energy management in a Smart Grid. *Lecture Notes in Computer Science* 2010; **6351**: 263–276.
35. Sandberg H, Teixeira A, Johansson KH. On security indices for state estimators in power networks, *Preprints of the First Workshop on Secure Control Systems, CPSWEEK 2010*, Stockholm, Sweden, 2010; 63–68.
36. Kosut O, Jia L, Thomas RJ, Tong L. Limiting false data attacks on power system state estimation, *Proceedings of 2010 Conference on Information Sciences and Systems*, Princeton, NJ, USA, March 2010; 1–6.
37. Kim T, Poor H. Strategic protection against data injection attacks on power grids. *IEEE Transactions on Smart Grid* 2011; **2**(2): 326–333.
38. Bobba R, Rogers K, Wang Q, Khurana H, Nahrstedt K, Overbye T. Detecting false data injection attacks on dc state estimation, *Preprints of the First Workshop on Secure Control Systems, CPSWEEK*, Stockholm, Sweden, 2010; 18–26.
39. Giani A, Bitar E, Garcia M, McQueen M, Khargonekar P, Poolla K. Smart grid data integrity attacks: characterizations and countermeasures, *IEEE SmartGridComm*, Brussels, Belgium, October 2011; 232–237.
40. Yi H, Li H, Campbell K, Zhu H. Defending false data injection attack on Smart Grid network using adaptive cusum test, *2011 45th Annual Conference on Information Sciences and Systems (CISS)*, Baltimore, MD, USA, 2011; 1–6.
41. Pasqualetti F, Carli R, Bullo F. A distributed method for state estimation and false data detection in power networks, *IEEE SmartGridComm*, Brussels, Belgium, October 2011; 469–474.
42. Feng J, Potkonjak M. Real-time watermarking techniques for sensor networks, *Proceedings of SPIE Security and Watermarking of Multimedia Contents*, Santa Clara, California, USA, 2003; 391–402.
43. Cao J, Fowler J, Younan N. An image-adaptive watermark based on a redundant wavelet transform, *Proceedings of 2001 IEEE International Conference on Image Processing,* Vol. 2, Thessaloniki, Greece, 2001; 277–280.
44. Kamel I, Juma H. Simplified watermarking scheme for sensor networks. *International Journal of Internet Protocol Technology* 2010; **5**(1): 101–111.
45. Kirovski D, Malvar HS. Spread-spectrum watermarking of audio signals. *IEEE Transactions on Signal Processing* 2004; **51**(4): 1020–1033.

SUPPORTING INFORMATION

Additional supporting information may be found in the online version of this article at the publisher's web site.