An ICS could be subject to cyber and/or physical attacks, which can be launched either remotely or locally. Attackers may tamper sensor reading or inject spoofing sensor data, and manipulate the actuators which, will cause anomaly of operations and eventually lead to physical damages to the system. Traditional intrusion detection methods based on network traffic cannot detect many low layer attacks originated in the physical domain, as there would be no abnormal network traffic [26].

Sensor data is transmitted to a Programmable Logic Controller (PLC) to take an appropriate action based on the sensor measurement. If an adversary can spoof sensor data in the digital or physical domain, it can derive a system to an unsafe state. The focus here is not on the confidentiality of the data as in legacy computer security but the integrity and trustworthiness of the data [15, 17]. Detection methods based on the physics of the process against attacks on sensor reading have been proposed in recent studies [2, 23, 25–27, 29, 31]. An attacker who tries to defy rules of physics would also expose itself. An understanding of the physics of the process can help to secure an ICS.

1.1 Proposed Technique

A novel technique is proposed to identify a physical process and detect data integrity attacks in an Industrial Control System (ICS). The proposed technique uses the small deviations in the process due to the deviations of the process (herein called process skews). The process skew is a noise that appears in sensor measurements due to the process fluctuations. Uniqueness in the skews is due to the specified operational constraints of the physical process. To create a process skew based fingerprints, it is challenging to extract process skew information from the sensor measurements. The proposed idea is inspired by the idea of clock skew in computers [16]. The concept of clock skew is that due to manufacturing inaccuracies, the clock of a computer will present a skew from its designed frequency. Similarly, for a process due to inaccuracies in the process, it would have a skew from what it is designed for. An example is that of a water pipe taking water to fill a tank. Pipes and tanks of two different sizes would take/store a different amount of water. Even if the pipes are of the same size, two different amounts of pumping force will result in a different amount of water flowing or being stored. The flow of water in a pipe and water storage in a tank are examples of the physical process. At the design stage, these processes are designed to meet certain operational requirements. However, when these processes are running they show small offsets from the designed parameters due to the physical inaccuracies in...
2 MOTIVATION AND OVERVIEW

In this section, we will present details related to Secure Water Treatment Testbed (SWaT), which is used as a case study in this work. An overview of the proposed technique is also presented.

2.1 Industrial Control Systems

Industrial Control Systems (ICS) is a broad domain of connected industrial systems. A particular example of a water treatment industrial process is considered in this study. In particular, Secure Water Treatment Testbed (SWaT) at Singapore University of Technology and Design is being used as a motivating example in this paper. SWaT is a fully functional testbed and is open for researchers to use. A brief introduction is provided in the following, but an interested reader is referred to the testbed paper [19]. The SWaT testbed produces the purified water, and it is a scaled-down version of a real water treatment process. In Figure 1 it can be seen that the testbed is distributed and there are different stages, where each stage is labeled as $P_n$, where $n$ is the nth stage. There are six stages in the SWaT testbed $P_1$ through $P_6$. Each stage is equipped with a set of sensors and actuators. Sensors include water quantity measures such as level, flow, and pressure and water quality measures such as pH, ORP and conductivity. Actuators are different motorized valves and electric pumps. Stage 1 is the raw water stage to hold the raw water for the treatment and stage 2 is the chemical dosing stage to treat the water depending on the measurements from the water quality sensors. Stage 3 is the ultra-filtration stage. Stage 4 is composed of de-chlorinator and stage 5 is equipped with reverse osmosis filters. Stage 6 holds the treated water for distribution. Data from the sensors and actuators are communicated to the PLCs using a level 0 network and PLC communicates to each other over a level 1 network, as shown in Figure 1.

2.2 Overview of the Proposed Technique

A major challenge is to extract the process skew from sensor data. An overview of the proposed technique is shown in Figure 2. The first step is to extract the measurements for a specific state of the process. It means that based on the actuator data, it is possible to determine the physical state of the process. For example, if the inlet pump is ON then the water is being filled in a tank. By knowing the state of the pump, it is possible to know the state of the physical process. However, such state information from the sensors and actuators might be spoofed by an attacker. Next, based on the state of each process a model along with the design parameters of the physical process is used to estimate the physical state of the process, for example, the water level in a tank. The difference between these estimates and real sensor measurements establishes an offset value, an amount by which the process is offset from what it should be, as per the design. These process offsets, when accumulated over time, reveals the process skew but still contain fluctuations due to the sensor noise. A linear regression model is used to obtain the best fit for each process skew. Process skew is obtained by calculating the rate of change of linear regression on offsets with respect to time. A theoretical proof based on the calculated entropy of the process skew is used to establish the uniqueness of process skews. A CUSUM detector is used to detect attacks based on the process skew. Details on the design of each block in Figure 2 are presented in Section 4.

3 THREAT MODEL

In an ICS, state of the physical process is known via sensors. System is kept in the normal operating bounds by the controllers based...
Process Skew

3.1 Attacker Model
Assumptions on Attacker: It is assumed that the attacker has access to the sensor’s measurements. A powerful attacker can arbitrarily change sensor measurements to the desired sensor value. We do not consider replay attack in this article because process skew profile for process would be preserved during a replay attack.

3.2 Attack Scenarios
Data Injection Attacks: For data injection attacks, it is considered that an attacker injects or modifies the real sensor measurement. In general, for a complex ICS, there can be many possible attack scenarios. We consider a generic attack to show the performance of the proposed technique. We evaluate the proposed technique for a range of network attack scenarios from benchmark attacks on SWaT testbed [14]. These attacks cover a wide range of attacks on both sensors and actuators. Since the proposed technique extracts the process skew for the physical properties, thus chemical sensors are excluded from this study, leaving us with a total of 25 attacks as detailed in Table 5 in Appendix. In general, an attack vector can be defined as,

$$\bar{y}_k = y_k + \delta_k,$$

where $y_k$ are the real sensor measurement, $\bar{y}_k$ is sensor measurement with a possible attack and $\delta_k$ is the data injected by an attacker at time step $k$. The detail about each $\delta_k$ (attack vector) is described in Table 5 in the Appendix where it can be seen that it ranges from an abrupt injection of data to more slow/stealthy change in sensor measurements.

3.3 Attack Execution
All the attacks which are taken from reference work [14], are executed by compromising the Supervisory Control and Data Acquisition (SCADA) system.
1. If flow present it is 1 else 0.

Table 1: The tank 1 in stage 1 of the SWaT testbed has one inlet valve labeled as MV-101 and one outlet pump labeled as P-101. Notice that there is a secondary backup pump also at the outlet labeled as P-102. Based on inflow and outflow there can be four possible states for the level in tank 1 based on input and output flow process.

<table>
<thead>
<tr>
<th>State (Inlet Flow</th>
<th>Outlet Flow)</th>
<th>Design Parameters</th>
<th>Process Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1 (0</td>
<td>1)</td>
<td>Outlet flow = 2.47 m$^3$/hr</td>
<td>In this scenario, the water is being flown out of the tank 1 i.e., emptying process.</td>
</tr>
<tr>
<td>S2 (0</td>
<td>0)</td>
<td>Inlet flow = 2.54 m$^3$/hr and Outlet flow = 2.47 m$^3$/hr</td>
<td>In this scenario, the water level stays constant, i.e., static process.</td>
</tr>
<tr>
<td>S3 (1</td>
<td>0)</td>
<td>Inlet flow = 2.54 m$^3$/hr</td>
<td>In this scenario, the water is being flown into the tank 1. i.e., filling process.</td>
</tr>
</tbody>
</table>

Figure 4: Level sensor in the SWaT testbed in stage 1 labelled as LIT-101 under the normal operation. This figure shows multiple runs of a physical process, e.g., water filling, water flowing out, both or none of the previous process. Each of these processes is labelled as S1 to S4, and the details are given in Table 1.

4 DESIGN OF THE PROPOSED TECHNIQUE

In this section, all the components of the proposed technique are discussed in detail.

4.1 Extracting the Process States

We begin by considering an example from the SWaT testbed. In Figure 4, level sensor (LIT-101) measurements for a duration of normal process are shown. It can be seen that based on the inflow and outflow, there can be four possible process states, i.e., S1: outflow is present but no inflow, S2: neither inflow nor inflow, S3: inflow is present but no outflow, S4: both input and output flow processes are present. Table 1 shows a detailed description of the four possible process states in the water tank. Design parameters in Table 1 shows the design for the inflow and outflow process and which process is present in a particular state.

The water treatment plant is run for seven days continuously and the data for the normal operations of the plant is collected. In Figure 5 four possible process states for the water level in tank 1 are shown. This data presents the particular states extracted from the seven days of the normal operation. There are hundreds of occurrences for each process state. Different colors in the plot represent different runs of the normal operation. The effects of the noise are evident from the variability of the process slope.

Each process is expected to behave according to the design parameters as shown in Table 1. However, as we can visually see in Figure 5 there are deviations due to the process noise. In Figure 5, the first state S1 shows different runs of the water emptying process from the tank. We can see the variations in each process run due to the sensor noise. This is also evident from the static (S2) and water filling (S3, S4) processes. Having seen the deviations and noise in these physical processes, the next step is to figure out the variation due to the process offset from the design. To quantify the amount of skew, we need to learn the process dynamics for all these states under the designed set points.

4.2 Design based System Model

In Figure 7 the sensor measurements for the water filling process and estimated sensor value based on the design are shown. The accumulated offset is also labelled to make the visual sense of the idea. A physical system diagram for stage 1 is shown in Figure 6. Tank 1 in stage 1 of the SWaT testbed is being used as a running example to demonstrate the idea. In Figure 6, it is shown that the water level in the tank is measured using a level sensor and the inflow and outflow of the water is being controlled by the motorized valve (MV-101) at the input and pump (P-101) at the output respectively. The idea is to model this inflow and outflow by considering the physical principles and the design of the physical process. Process skew information is extracted by figuring out the process dynamics drift from the design due to the process noise. For a tank, we know that the rate of change of water inside the tank is equal to the difference between water flowing into the tank and water flowing out from the tank with respect to time. We can represent this using mass-balance equation [24] such as,

$$\frac{dV}{dt} = Q_{in} - Q_{out}$$

$$\frac{dh}{dt} = \frac{Q_{in} - Q_{out}}{A}$$ since $V = A \times h$, (2)
Figure 5: These sub-figures show four possible states of a physical process in a water tank, as described in Table 1. Level sensor in the SWaT testbed in stage 1 labelled as LIT-101 under the normal operation.

Figure 6: Modeling the process for the level sensor in Tank1.

Figure 7: The idea of process skew. Water level sensor measurement and its estimates using the model are shown. The difference between both is defined as the offset.

where \( V \) represents the volume of the tank, \( A \) is the cross-sectional area of the tank, and \( h \) is the height of the water inside the tank. (2) provides a linear equation, we can see the term \( Q_{in} - Q_{out} \) represents the water flow which depends upon the PLC control actions implemented via MV-101 and P-101. From Figure 6, it can be seen that using the height and diameter of the tank from design documents, it is possible to figure out the volume and the cross-sectional area of the tank. Let us consider that state of the physical process as the height of the water inside the tank. Then the solution of this equation gives us the following result.

\[
x_{k+1} = x_k + u_k,
\]

where \( u_k \) is the PLC control action. Here \( x_k \) represents water level in the tank at time \( k \). The control action \( u_k \) can be an either open/close (for the motorized valve) or on/off (for the pump). Similarly, we can describe the sensor state and we can get the set of system equations.

\[
\begin{align*}
    x_{k+1} &= Ax_k + Bu_k, \\
    y_k &= Cx_k.
\end{align*}
\]

(3)

Where \( y_k \) is the sensor measurement driven by the control action \( u_k \). Matrices \( A, B \) and \( C \) are the state-space matrices of appropriate dimensions. From (3), it can be seen that if we have a system state value at time \( k \), then given the PLC control \( u_k \) we can predict the next state at time \( k+1 \). Table 1 shows a list of design parameters for each type of control action. For example, S4 has the MV-101 control as to open the valve and P-101 as turned on, given the information of this control from PLC, we know from the design of the physical process that how much the water level in the tank should increase. However, as we will see, due to the process noise, there would be deviations in the process states from what it was designed for.

4.3 Extracting the Process Offsets

Using the process design and the system of equations in (3), we could extract the process skews, i.e., how much the real process dynamics are offset from the designed physical process. In figure 8 we can see the offsets in the level of the water in tank1.

**Definition 4.1.** Process Offset: Deviation of the process dynamics due to the process inaccuracies, from the design at each time step.

The process offsets are calculated at each time step for the time while the process is active. All the process offsets are accumulated over time and then process skew is extracted.

**Definition 4.2.** Process Skew: Slope of the accumulated process offsets for a process activity time frame.

In Figure 8, we can see the accumulative offsets for the different process states. S1 represents the case of water outflow from the tank.
A negative slope indicates that the real process is actually slower than the designed parameters. S2 is the case when the process is static and there is no inflow or outflow. Hence, the process is missing, so no process skew exists. For the case of S3, only the inflow is present and the positive slope shows that the real process is actually faster than the designed one. S4 is the case when both the inflow and outflow are present. In this case, it can be seen that the real process is actually slower than the design. Now all these physical state scenarios happen in the same physical process that is, the water tank in stage 1 of the SWaT tested. Although it’s the same process, it is observed that based on the process skew all the physical states of the process could be distinguished from each other. This establishes a process fingerprint. However, one important observation to make in Figure 8 is that the offsets are noisy due to the sensor noise. The challenge here is to remove the sensor noise effect without disturbing the process offsets. In the following, a mathematical expression is derived for the process skew. Consider the linear time-invariant model of the system with sensor and process noise as

\[
\begin{aligned}
    x_{k+1}^s &= Ax_k^s + Bu_k + v_k, \\
    y_k^s &= Cx_k^s + \eta_k.
\end{aligned}
\]

(4)

where \( y_k^s \) is the sensor measurement with the measurement noise \( \eta_k \) and \( x_{k+1}^s \) is the system state.

**Proposition 4.1.** At each time step, the difference between sensor measurements given by (4) and sensor measurement estimate (by design) given by (3) is calculated to obtain the process offset as, \( y_{k+1}^s - y_{k+1}^* = CA[O_k] - CV_k - \eta_{k+1} \), where \( O_k = x_k - x_k^s \) is the offset.

**Proof:** The difference between (4) and (3) is given as,

\[
\begin{aligned}
    y_{k+1} - y_{k+1}^* &= CAx_k + CBu_k - CAx_k^s - CBu_k - CV_k - \eta_{k+1}, \\
    y_{k+1} - y_{k+1}^* &= CA(x_k - x_k^s) - CV_k - \eta_{k+1}.
\end{aligned}
\]

(5)

(6)

(7)

As the offset is defined as the difference the real system state and the estimated state of the system \( (x_k - x_k^s) \), it produces,

\[
y_{k+1} - y_{k+1}^* = CA[O_k] - CV_k - \eta_{k+1}
\]

(8)

From (8) that the offset \( (O_k) \) can be extracted at each time step. From (8), it is observed that the process offset contains the noise from the sensor; therefore, it is important to fit a straight line to data to get the process skew. Since the process skew is the slope of the accumulated process offsets, we need a straight line to represent each of the above physical states. Towards that end, we resort to the linear regression model for each process offset as evidently, the offsets are linear in time.

**4.4 Process Skew**

To establish the linearity between the time and the progression of the process, correlation coefficients are used. Correlation calculates the level of the linear relationship between variables. If we have a high correlation between two variables, then it means that the values for those increase or decrease in a linear relationship. However, uncorrelated variables might still be dependent on each other it is just that the relationship might be nonlinear. For \( N \) scalar values of two variables, the Pearson correlation coefficient is defined as,

\[
\rho(X, Y) = \frac{\sum_{i=1}^{N}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{N}(X_i - \bar{X})^2} \sqrt{\sum_{i=1}^{N}(Y_i - \bar{Y})^2}}
\]

(9)

where \( \bar{X} \) is the mean of the variable \( X \) and \( \bar{Y} \) is the mean of the variable \( Y \). We have found that the process data is linearly correlated with the time as the process is linearly increasing or decreasing in time. Linear regression approach is adopted to get the data models describing the relationship between the variable in a mathematical form. Least squares fit is used to obtain the model. For a set of \( n \) observed values of \( X \) and \( Y \) given by \( X = (x_1, x_2, ..., x_n) \) and \( Y = (y_1, y_2, ..., y_n) \) respectively. These values for a system of linear equations which can be represented in matrix form as,

\[
\begin{bmatrix}
y_1 \\
y_2 \\
\vdots \\
y_n
\end{bmatrix} =
\begin{bmatrix}
1 & x_1 \\
1 & x_2 \\
\vdots & \vdots \\
1 & x_n
\end{bmatrix}
\begin{bmatrix}
\beta_0 \\
\beta_1 \\
\vdots \\
\beta_i
\end{bmatrix}
\]

(9)
which can be simplified to,
\[ Y = \beta_0 + \beta_1 X + \epsilon, \]
where \( \beta_0 \) is the y-intercept, \( \beta_1 \) is the slope/regression coefficient and \( \epsilon \) is the model error.
Figure 9 shows a linear model fitting through the process skew data. This linear model is used to find the slope that defines the process skew. Figure 9 shows a visual idea regarding the accuracy of the linear model. To quantify the goodness of a system model, mean square error (MSE) is used as a metric. In particular, one minus the root mean square error (RMSE) defines the estimation accuracy or best fit of a model,
\[ \text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}} \tag{11} \]
MSE is the difference between sensor measurement and sensor measurement estimate squared and essentially gives the distance between measured and estimated value or in other words, how far the estimated value from the measured value is. The model accuracies for the three stages of SWaT and corresponding process states used in this study (from SWaT testbed) are shown in Table 2. It can be seen that the obtained system model is very accurate, with almost zero mean error for all the runs of a process. Table 2 shows the mean of models created for all the runs of the process. The process offsets are accumulated for the run of a process,
\[ O_{\text{accum}} = \sum_{k=1}^{n} (O_k) \tag{12} \]
and the corresponding process skew is given as,
\[ \text{Process skew} = \frac{d(O_{\text{accum}})}{dt}. \tag{13} \]

4.5 Skew Uniqueness

In Figure 10, process skew distribution for all the eight physical processes in the three stages of SWaT testbed is shown. It can be observed that all the processes can be uniquely distinguished based on the process skew profile. Figure 10 shows a visual analysis for process skew uniqueness; however, we will see a mathematical proof for the skew uniqueness. It is imperative to study that fingerprints are information-theoretically unique in order to negate the possibility of impersonation attacks. An attacker can use skews of her processes to design compromises. Let \( w(t) \) be the signal corresponding to a process skew. In order to present an information-theoretic analysis on the top, we study justification of two important criteria:

(1) mutual information between skews as recorded for the same process, i.e., in successive operations should be high, \( \approx 1 \), and
(2) conditional entropy of skews with other process skews should be very low, \( \ll 1 \).

In order to investigate these relations mutual information, \( I(\cdot) \), for process \( i \) is defined as
\[ I(w_{ij}, w_{ik}) = H(w_{ij}) - H(w_{ij}|w_{ik}) \]
where \( i \in 1 : S, j \in 1 : N, H(w_{ij}) \) is the entropy of \( j \)th attempt by a process \( i \) and \( H(w_{ij}|w_{ik}) \) is conditional entropy of \( i \)th process for \( j \)th attempt, given the features of \( k \)th attempt. For high recall, mutual information for each of the process skew should be close
The process skews for different runs of a particular process; for example, a water filling process is accumulated. The process skew is labelled as $S_{i, j, k}$ here for easy reference, where $k$ is the time step. The standard CUSUM [20] procedure is explained using the following equations.

**CUSUM:**

\[
\begin{align*}
S_{0,i} &= \bar{T}_i, \\
S_{0,i} &= \bar{T}_i, \\
\hat{k}^+ &= 0, \\
\hat{k}^- &= 0,
\end{align*}
\]

\[
\begin{cases}
S^+_{k,i} = \max(\bar{T}_i, S^+_{k-1,i} + r_{k,i} - \hat{k}_i - \kappa_i), & \text{if } S^+_{k-1,i} \leq \tau^+_i, \\
S^+_{k,i} = \tilde{T}_i & \text{and } \hat{k}^+_i = \hat{k}^+_i + 1, & \text{if } S^+_{k-1,i} > \tau^+_i.
\end{cases}
\]

\[
\begin{cases}
S^-_{k,i} = \min(\bar{T}_i, S^-_{k-1,i} + r_{k,i} - \hat{k}_i + \kappa_i), & \text{if } S^-_{k-1,i} \geq \tau^-_i, \\
S^-_{k,i} = \tilde{T}_i & \text{and } \hat{k}^-_i = \hat{k}^-_i + 1, & \text{if } S^-_{k-1,i} < \tau^-_i.
\end{cases}
\]

**Design parameters:** Bias $\kappa_i > 0$; threshold $\tau_i > 0$.

**Output:** $\text{Alarm}(s) = \hat{k}^+_i + \hat{k}^-_i$.

From (14)-(15), it can be observed that $S^+_{k,i}$ and $S^-_{k,i}$ accumulate the distance measure $r_{k,i}$ over time to measure how far are the values of the residual from the target mean ($\bar{T}_i$). To tune the CUSUM detector there is also a slack variable $\kappa$ chosen to be $\frac{1}{2} + \sigma_i$ in this study, where $\Gamma$ is a multiplier to the standard deviation ($\sigma$) and usually taken between 3 and 5 [20]. An alarm is raised when this accumulation becomes greater or less than a chosen threshold $\tau_i$. The sequence $S_{k,i}$ is reset to the target mean value each time it becomes negative or larger than $\tau_i$. If $r_{k,i}$ is tightly bounded and $\kappa_i$ is not sufficiently large, the CUSUM sequence $S_{k,i}$ grows unbounded until the threshold $\tau_i$ is reached, no matter how large $\tau_i$ is set. In order to prevent such drifts, the slack variable $\kappa_i$ must be selected properly based on the statistical properties of the distance measure. Once $\kappa$ is chosen, the threshold $\tau_i$ must be selected to achieve a required false alarm rate $\mathcal{A}_i$. $\mathcal{A}_i$ is defined as the false alarm rate for the CUSUM procedure defined as the expected proportion of observations which are false alarms [1, 30].

## 5 Evaluation

The proposed technique is evaluated in a real water treatment testbed. The following metrics are used for performance evaluation. We define $TP_i$ as true positive for class $c_i$ when it is rightly classified based on the ground truth. False-negative $FN_i$ is defined as the wrongly rejected, and False positive $FP_i$ as wrongly accepted. True negative $TN_i$ is the rightly rejected class. The True Positive Rate (TPR) and False Positive Rate (FPR) are defined as follows:

\[
\begin{align*}
\text{True Positive Rate (TPR)} &= \frac{TP}{TP + FN} = 1 - FNR, \\
\text{False Positive Rate (FPR)} &= \frac{FP}{FP + TN} = 1 - TNR.
\end{align*}
\]

Ideally, FPR should be as small as possible and TPR as high as possible. Both TPR and FPR being ratios range between 0 and 1.

### 5.1 Normal Operation

For the normal operation data from the SWaT testbed is collected for a period of seven days. During the normal operation, the plant was run continuously under the normal conditions and as it was designed to operate. The operating conditions from the design was already presented in Table 1. For all the possible process states, data is extracted. Process offsets are extracted for each process in Stage1, Stage3 and Stage4 of the SWaT testbed. Stage2 and Stage5 is constituted of chemical sensors and reverse osmosis process respectively; therefore, those two stages are not considered in this study. This work is focused on studying the physical properties of the process. Studying the chemical properties of the process is out of the scope of this work. During seven days, water filling or the emptying process happened hundred of times. Process offsets are calculated for each of these process runs. Process offsets are noisy due to the noise from the sensors. A linear regression model is fit to handle the noise in the signal. After the linear model is fitted, we obtain a straight line for accumulated process offsets over process time frame. The rate of change of these process offsets is defined as the process skew. Figure 8 shows process offsets for different process states of the Stage1 of the SWaT testbed. Figure 9 shows an example of linear model fitting for the process offsets. The obtained linear model can be used to calculate process skews. Normal process skews are used with a CUSUM detector to establish
Table 3: Design and performance of CUSUM detector on the normal data. \( \mu \) and \( \sigma \) are mean and standard deviation of the process skews.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( \kappa )</td>
<td>0.0013</td>
<td>0.0017</td>
<td>7.27e-04</td>
<td>0.0217</td>
<td>7.90e-04</td>
<td>0.0014</td>
<td>0.0027</td>
<td>0.0012</td>
<td></td>
</tr>
<tr>
<td>( \tau )</td>
<td>0.0102</td>
<td>0.0219</td>
<td>0.0073</td>
<td>0.1083</td>
<td>0.0126</td>
<td>0.0199</td>
<td>0.0277</td>
<td>0.0148</td>
<td></td>
</tr>
<tr>
<td>( \mu )</td>
<td>0.0070</td>
<td>-0.0288</td>
<td>-0.0113</td>
<td>0.0082</td>
<td>-0.0161</td>
<td>0.0146</td>
<td>-0.0277</td>
<td>-0.0327</td>
<td></td>
</tr>
<tr>
<td>( \sigma )</td>
<td>0.0026</td>
<td>0.0034</td>
<td>0.0015</td>
<td>0.0433</td>
<td>0.0016</td>
<td>0.0028</td>
<td>0.0054</td>
<td>0.0025</td>
<td></td>
</tr>
<tr>
<td>TNR</td>
<td>96.04%</td>
<td>97.46%</td>
<td>97.10%</td>
<td>96.99%</td>
<td>96.97%</td>
<td>96.38%</td>
<td>97.39%</td>
<td>95.31%</td>
<td></td>
</tr>
<tr>
<td>FPR</td>
<td>3.96%</td>
<td>2.54%</td>
<td>2.90%</td>
<td>3.01%</td>
<td>3.03%</td>
<td>3.62%</td>
<td>2.61%</td>
<td>4.69%</td>
<td></td>
</tr>
</tbody>
</table>

RQ1: Can process skews be used to fingerprint each process state? Table 3 shows a high true negative rate meaning it is possible to identify each process state with a high accuracy based on the process skew fingerprint. A physical process goes through different process states during the operation of the process plant. For example, for the process of a fluid tank, either fluid is flowing out, flowing in, both or in a static state. Since different process states have different skews, it is possible to uniquely identify each process state based on its process skew fingerprint.

RQ2: Does a process skew depend on the initial conditions of a process dynamics? This means that, does it matter at what initial state the process starts. For example, does it matter if the water filling process starts at 500 mm or 800 mm water levels? In this study, a particular process, for example, a water filling process started at different initial states depending on the control logic. The results presented in Table 3 is a combination of all possible initial conditions of a particular process and a fingerprint is created for all the runs of the process taken together. It can be seen from Table 3 that the process skew based fingerprint is stable over a range of process start and end conditions, making it robust to use in a real-world system.

5.2 Attack Detection

RQ3: Can the proposed process skew based fingerprint be used as an attack detection method? The performance of the proposed technique as an attack detection method is evaluated under a range of attack data collected from SWaT testbed. SWaT was subject to different attack scenarios for four days. This is to say that for four days there were a lot of runs of normal operation and then there were attack instances in between. A complete list of attacks is shown in Table 5 in the Appendix. An example of process skews for the process of tank4 in stage4 is shown in Figure 12. From Figure 12 it is evident that using process offsets and skews, it is easy to detect attacks. The attack scenarios deviate from the normal process offsets. Note that there are attack start and attack stop markers. In some cases when the attack was stopped the slope of the process offset, which is
Table 4: Evaluation of the proposed technique on the attack data from SWaT testbed. TPR presents the attacks which were detected accurately as percentage (attacks-detected/total-attacks-executed).

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Stage 1</th>
<th>Stage 3</th>
<th>Stage 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPR</td>
<td>100%(3/3)</td>
<td>100%(4/4)</td>
<td>100%(1/1)</td>
</tr>
<tr>
<td>FPR</td>
<td>3.2%(4/125)</td>
<td>1.65%(2/121)</td>
<td>8.06%(10/124)</td>
</tr>
</tbody>
</table>

Figure 14: Stealthy attack on the level sensor in Stage 1 of the SWaT testbed. The Stealthy attack is designed to spoof the values of the level sensor measurements so that the residual shown on the right does not surpass the threshold.

Figure 6.1 A Comparison with Model based Detectors

Can process skew fingerprint be used to detect attacks those are stealthy for the model based detectors [3, 5, 6]? Figure 14 shows the execution of such an attack on the SWaT testbed. On the left-hand plot, actual measurements and sensor estimates obtained from level sensor, LIT-101, using the system model have been plotted. On the right-hand side, respective residual (measured - estimated) values for the level sensor are shown. Upper and lower limits for a statistical detector can be seen. On the left-hand plot, the dotted green line shows the ground truth for the process state, while the attacker spoofed the sensor values and managed to derive the system away from the normal operation overtime during the attack period. The spoofed values are chosen such that the residual values never grow bigger than a model-based detector threshold and hence, could not get detected. But from the ground truth, we know that the process dynamics are not what the attacker is making PLC to believe. Using process skew, it is possible to detect the presence of such an attacker. The idea is if an attacker wants to deviate the process from its desired operation, it must defy the process dynamics and expose itself to process skew. In comparison, it can be concluded that the proposed process skew based technique can detect attacks that are stealthy for the system model based detectors.

6.2 Scalability

This case study is carried out on a water treatment plant but we believe that the technique itself is generalizable. The physical process discussed in this work is water/fluid dynamics but there are other similar processes, e.g., gas or other chemical fluids where the same techniques should work. Moreover, in this work, a range of different processes and process states are considered that points towards its scalability. On the top of it, the demonstration on a real system highlights its applicability in real-world applications.

7 CONCLUSIONS

We demonstrated that indeed a process skew exists for each process due to the deviations in the process from the design. The proposed technique can be used to fingerprint the different process states, for example, filling, emptying, or a combination of these process dynamics in a water treatment system. Hence, it is possible to detect attacks on the processes. An extensive evaluation of the proposed technique on a real-world water treatment system validates its applicability and practicality.

While carrying out this study, some useful observations have been made regarding the process transients. When a process changes from one state to another, the process dynamics are said to be in a transient state and it takes time to reach a steady-state. In the future, we would like to explore this transient feature of the processes for attack detection.

ACKNOWLEDGEMENT

This work was supported by the SUTD start-up research grant SRG-ISTD-2017-124.

REFERENCES

### Table 5: Executed Attacks on SWaT Testbed from reference [14]

<table>
<thead>
<tr>
<th>Attack Sequence Number</th>
<th>Start Time</th>
<th>End Time</th>
<th>Attack Point</th>
<th>Start State</th>
<th>Attack</th>
<th>Expected Impact or Attacker Intent</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>28/12/2015 10:58:50</td>
<td>Turn on P-102</td>
<td>P-101 is on</td>
<td>Where as P-102 is off</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>28/12/2015 11:28:22</td>
<td>Increase by 1 mm</td>
<td>LIT-101</td>
<td>Water level between L and H</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>28/12/2015 12:15:53</td>
<td>Water level in</td>
<td>LIT-301</td>
<td>Water level increased above BH</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>28/12/2015 13:26:13</td>
<td>Value of DPT-101</td>
<td>Open MV-101</td>
<td>Set value of DPT as +48kpa</td>
<td>Backwash process is started again, and again; Normal operation stops; Decrease in water level of tank 101; Increase in water level of tank 301</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>28/12/2015 14:19:00</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>28/12/2015 14:28:20</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>28/12/2015 14:46:20</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>28/12/2015 15:20:10</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>28/12/2015 16:08:25</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>28/12/2015 16:42:50</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>28/12/2015 17:01:25</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>28/12/2015 17:43:40</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>28/12/2015 18:26:13</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>28/12/2015 19:06:50</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>28/12/2015 19:46:50</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>28/12/2015 20:27:00</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>28/12/2015 21:08:18</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>29/12/2015 10:01:50</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>29/12/2015 10:46:50</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>29/12/2015 11:35:40</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>29/12/2015 12:15:53</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>29/12/2015 12:56:30</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>29/12/2015 13:37:40</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>29/12/2015 14:19:00</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>29/12/2015 14:58:50</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>29/12/2015 15:37:40</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>29/12/2015 16:16:25</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>29/12/2015 16:55:18</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>29/12/2015 17:44:30</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>29/12/2015 18:23:40</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>29/12/2015 19:02:50</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>29/12/2015 19:41:30</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>29/12/2015 20:20:40</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>29/12/2015 21:00:00</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>29/12/2015 21:39:40</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>29/12/2015 23:06:00</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>30/12/2015 00:44:20</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>30/12/2015 01:23:00</td>
<td>Value of FT-401</td>
<td>MV-101</td>
<td>Set value of FT-401 above 1</td>
<td>Tank Underflow; Damage P-101</td>
<td></td>
</tr>
</tbody>
</table>

- Attack Sequence Number: The sequence number of the attack in the testbed.
- Start Time: The start time of the attack.
- End Time: The end time of the attack.
- Attack Point: The specific point in the testbed affected by the attack.
- Start State: The initial state of the system before the attack.
- Attack: The action performed by the attacker.
- Expected Impact or Attacker Intent: The expected impact or the attacker's intent for each attack.