Can Replay Attacks Designed to Steal Water from Water Distribution Systems Remain Undetected?

Venkata Reddy Palleti  
Indian Institute of Petroleum and Energy-Visakhatpatnam  
Singapore University of Technology and Design  
venkat_palleti.che@iipe.ac.in

Chuadhry Mujeeb Ahmed  
Singapore University of Technology and Design,  
University of Strathclyde  
chuadhry@alumni.sutd.edu.sg

Vishrut Kumar Mishra  
Singapore University of Technology and Design  
kumar_mishra@sutd.edu.sg

Aditya Mathur  
Singapore University of Technology and Design  
aditya_mathur@sutd.edu.sg

ABSTRACT
Industrial Control Systems (ICS) monitor and control physical processes. ICS are found in, among others, critical infrastructures such as water treatment plants, water distribution systems, and the electric power grid. While the existence of cyber-components in an ICS leads to ease of operations and maintenance, it renders the system under control vulnerable to cyber and physical attacks. An experimental study was conducted with replay attacks launched on an operational water distribution (WADI) plant to understand under what conditions an attacker/attack can remain undetected while stealing water. A detection method, based on an input-output Linear Time-Invariant system model of the physical process, was developed and implemented in WADI to detect such attacks. The experiments reveal the strengths and limitations of the detection method and challenges faced by an attacker while attempting to steal water from a water distribution system.

KEYWORDS

1 INTRODUCTION
Public utilities, often referred to as critical infrastructure, such as a water treatment plant, water distribution system, power generation etc., play an important role in a nation’s economy. Industrial Control Systems (ICS) are used for the monitoring and controlling of these infrastructures. An ICS integrates networking, sensing, and computing technologies with the physical systems to provide efficient and reliable operation. Often the physical processes and cyber components (computers and networks) interact with each other in a feedback loop. Also, some of these systems are spatially distributed rendering them vulnerable to cyber and physical attacks [20]. Several attacks on water distribution systems have been reported in recent years. Two of the often cited such attacks include one at the Maroochy Shire Water Services [36] and another at the Kemuri Water Company (KWC). Industrial Control Systems Cyber Emergency Response Team (ICS-CERT) has emphasized the importance of understanding these attacks against critical infrastructures [18, 21].

The purpose of a water distribution system is to fulfill customer’s demand while ensuring appropriate quality of the delivered water. However, due to the complex nature of these systems, they are vulnerable to attacks such as intentional contamination, leaks, etc. Among such attacks, leakage in water distribution systems is of significant importance for effective management and water quality control [16, 32]. Leaks in water distribution systems could be accidental or intentional. Accidental leaks can occur due to the high pressure in pipelines. In contrast, intentional leaks occur when an attacker wants to steal water from a pipeline [5]. Leaks can be detected through the appropriately installed pressure sensors and flow meters. In order to increase the impact of a physical attack (water leaks in this work), an attacker might compromise a subset of sensors and alter their measurements [11]. Therefore, leaks on water distribution systems can be considered as a cyber-physical attack.

In this paper, we report water leakage experiments conducted on an operational water distribution plant named WADI [8]. The leaks are simulated through a valve that bypasses water from the main pipeline. It is assumed that the primary goal of the attacker is to steal water while remaining undetected. Thus, to remain undetected during leak simulation, the attacker launches replay attacks on a strategically selected subset of sensor and actuator readings. However, the choice of sensors and actuators also depends on the knowledge and capabilities of an attacker. The objective of this paper is to experiment with different attacker models and to determine whether an attacker is successful in remaining undetected against the statistical attack detection model.

Contributions: Experimental study in a water distribution system to understand (a) the requirements for detecting cyber and physical attacks with the intention to steal water and (b) effectiveness of a Linear Time Invariant and CUSUM-based method for detecting cyber attacks.

Organization: The remainder of this paper is organized as follows. Literature related to the work reported in this paper is reviewed in Section 2. Architecture and operation of WADI used for experimentation is in Section 3. An attack detection framework, based on Linear Time Invariant model of a water distribution system together with the CUSUM detector, used in this work are described in Section 4. Water leakage experiments and results are in Section 5. Conclusions, that map to the contributions above, derived based on the analysis of the data obtained are in Section 6.
2 RELATED WORK

Supervisory control and data acquisition (SCADA) systems are widely used to monitor and control operations in critical infrastructure assets such as electrical power distribution facilities, oil and gas pipelines, water distribution systems and sewage treatment plants. Several attacks on water distribution systems have been reported in recent years [18]. In 2016, hackers breached a water utility (Kemuri water company) and manipulated systems responsible for water treatment and flow control. A famous SCADA attack incident on Maroochy Water Services in Queensland, Australia caused extensive sewage spilling [36]. It is to be noted that such attacks can have a huge impact on critical infrastructure assets leading to economic loss and service disruption on the consumers. In recent years there is a significant growth on gaining knowledge on critical infrastructure attacks, its impacts and different attack detection mechanisms. Some of these research works are discussed below.

Attack detection in canal networked systems using hydro-dynamic models has been reported in [11, 12]. An investigation into the challenges in the security of control systems when sensor and actuator data are compromised has been reported in [9, 15]. The general approach in the literature is to study the effect of specific attacks against a particular system. The specific attacks include denial of service and deception attacks against a networked control system. Denial of service attack refers to the compromise of the availability of resources by jamming communication channels [10, 19]. Deception attacks refer to the compromise of the integrity of sensor and actuator data. Specific types of deception attacks include false data injection, replay, and stealthy attacks. In [17, 23], false data injection attacks are studied in the power networks assuming an attacker has perfect knowledge of the system model. In [24] authors have demonstrated the effect of replay attacks on the sensor measurements and proposed a methodology to detect such attacks. Various cyber attacks on networked control systems are studied in [37] on a quadruple tank testbed.

There are several works on using the model of a physical process [13, 27, 31]. Most of these works borrow ideas from fault detection literature and has also contributed towards the limitations of fault detectors to be used as attack detectors. Towards that end, secure state estimation has extensively been studied. Recently, a research work in [35] proposed a search algorithm based on Satisfiability Modulo Theory (SMT) to speed up the search of possible sensors sets. An anomaly-based methodology for the detection of a wide range of integrity attacks in cyber-physical critical infrastructure is reported in [28]. As stated previously, leakage in water distribution systems is of significant importance. [14, 40] studies leak detection and isolation in open water channels.

Research on cybersecurity in CPSs through experimental investigation of real testbeds has recently received more attention, especially in public critical infrastructures. Authors in [3] experimentally investigated the impact of various cyber attacks on an operational water treatment plant. Also, in their subsequent work [2] they designed a distributed mechanism for detecting cyber attacks on a water treatment system by systematically deriving invariants. Recently, in [6, 7, 30, 33] researchers report an investigation into the effectiveness of yet another detection scheme that uses the Kalman filter on an operational water treatment testbed system. Experimental investigation on operational and real testbeds provides how an attacker can perform attacks on these systems to achieve his goal. Attacker’s intent realization depends on his/her capabilities such as network and process knowledge. In this work, an operational water distribution testbed is used to show how an attacker performs cyber-physical attacks to realize his intent. The intent of the attacker is assumed as stealing water from the pipeline of a water distribution system.

3 WATER DISTRIBUTION PLANT

Water Distribution (WADI) plant is an operational test bed supplying 10 US gallons/min of filtered water [8]. WADI is open for researchers to study and design the safe and secure large-scale cyber physical systems. It represents a scaled-down version of a large water distribution network in a city. WADI consists of three stages, namely the primary grid (P1), secondary grid (P2), and return water grid (P3). Figure 1 is an overview of the processes involved in the WADI testbed. The arrows in this figure show the flow of the water from one stage to another stage. For simplicity, we divide P2 into three parts, namely P2A, P2B and P2C. For recycling, the return water grid pumps water to the primary grid. Water quality analyzers are installed in return water grid to check water quality before pumping it into the primary grid. In order to measure the water levels in tanks, level sensors, namely, 1_LT_001, 2_LT_002 and 3_LT_001 are installed in the stages P1, P2 and P3 respectively. The detailed architecture of the WADI testbed was explained in [8].

The list of all sensors ($i_1$) and actuators ($i_2$) is in Table 1 and is segregated by the different stages. The list of sensors and actuators used in our experiments is shown in the last column of the table.

In WADI, P2b is equipped with a leak detection line. Water leak can be simulated by a transparent double containment piping section fitted with a modulating control valve (2_MCV_007) that opens into the outer pipe. Valve 2_MV_008 is used to bypass water from the main pipeline section and the bypass water drains into the return water grid. Figure 2 shows a picture of the leak setup in WADI. The process flow diagram of the leak setup is represented in Figure 3. Two pressure meters, 2_PIT_001 and 2_PIT_002 are also installed in P2a and P2b, respectively. These sensors are located at, respectively, upstream and downstream of the leak point. A flow meter 2_FIT_002 is installed on the gravity line to measure the amount of water flowing into the consumer tanks.

Three Programmable Logic Controllers (PLCs) are installed to control each stage of WADI. These PLCs use National Instruments Compact RIO as RIO (Remote Input Output) devices. The communication network contains three layers namely, layer-0 (L0), layer-1 (L1) and layer-2 (L2). L0 is at the process level and connects actuators/sensors and I/O modules via RS485-Modbus protocol. L1 is the plant control network where all PLCs are connected to a central node in a star topology. The third layer L2 is a communication network between a touch panel Human-Machine Interface (HMI) and the plant control network.
Can Replay Attacks Designed to Steal Water from Water Distribution Systems Remain Undetected?

Conference’17, July 2017, Washington, DC, USA

Figure 1: Three stages in WADI are shown. Solid arrows indicate flow of water and sequence of processes.

Table 1: List of sensors and actuators in WADI

<table>
<thead>
<tr>
<th>Stages</th>
<th>Sensors (𝑖𝑠)</th>
<th>Actuators (𝑖𝑎)</th>
<th>Used sensors and actuators</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>1_LT_001, 1_FIT_001, 1_AIT_001, 1_AIT_002, 1_AIT_003, 1_AIT_004, 1_AIT_005</td>
<td>1_LT_001, 1_LT_002, 1_FT_001, 1_P_001, 1_P_002, 1_P_003, 1_P_004, 1_P_005, 1_P_006</td>
<td>1_LT_001, 1_FIT_001, 1_MV_001, 1_MV_004, 1_P_005</td>
</tr>
<tr>
<td>P2A</td>
<td>2_LT_002, 2_LT_003, 2_FIT_001, 2_AIT_001, 2_AIT_002, 2_AIT_003, 2_AIT_004, 2_AIT_005</td>
<td>2_MV_003, 2_MV_004, 2_MV_005, 2_MV_006</td>
<td>2_LT_002, 2_LT_003, 2_MV_006, where x={1,2,3,4,6}</td>
</tr>
<tr>
<td>P2B</td>
<td>None, 2_FIT_002, 2_FIT_003, 2_LT_001, 2_LT_002, 2_LT_003, 2_LT_004, 2_LT_005</td>
<td>2_MV_008, 2_MV_009, 2_MV_010, 2_MV_011, 2_MV_012, 2_MV_013, 2_MV_014, 2_MV_015</td>
<td>2_LT_002, 2_LT_003, 2_LT_004, 2_LT_005</td>
</tr>
<tr>
<td>P2C</td>
<td>2_LS_x01, where x={1,…,6}, 2_FQ_x01, where x={1,…,6}</td>
<td>None, 2_MCV_x01, 2_MCV_x02, 2_MCV_x03, 2_MCV_x04, 2_MCV_x05</td>
<td>2_MCV_x01, 2_MCV_x02, 2_MCV_x03, 2_MCV_x04</td>
</tr>
<tr>
<td>P3</td>
<td>3_LT_001, 3_LT_002, 3_LT_003, 3_LT_004, 3_LT_005</td>
<td>3_MV_001, 3_MV_002, 3_MV_003, 3_MV_004, 3_MV_005</td>
<td>3_MV_002</td>
</tr>
</tbody>
</table>

4 SYSTEM MODEL BASED ATTACK DETECTION FRAMEWORK

In this section, a system-model based attack detection framework is explored. First, a system-model is derived based on state dynamics followed by its validation. A statistical detector namely CUMulative SUM (CUSUM) is used for attack detection.

4.1 System model

A system model represents the dynamics of a physical process as a mathematical model. Figure 1 shows the physical process in WADI. The state of the system is measured using sensors. Actuators are controlled by the PLCs to affect process state. Control actions are governed by user demands labeled as consumer tanks in Figure 1. For example, for an elevated tank, the level of water is considered as its state measured by a level sensor. Such a state in a water distribution process is dynamic and is governed by the actuators at the inlet and outlet of the process. We can consider these actuation signals as inputs to the control system and sensor measurements as outputs. The goal is to obtain a system model of the following form,

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

where \( x \in \mathbb{R}^n \) is system state vector, \( A \in \mathbb{R}^{nxn} \) is state space matrix, \( B \in \mathbb{R}^{nxp} \) is the control matrix, \( y \in \mathbb{R}^m \) are the measured outputs, \( C \in \mathbb{R}^{mxn} \) is measurement matrix, and \( u \in \mathbb{R}^p \) denote the system control. The state space matrices \( A, B, C \) capture the system dynamics and can be used to find a specific system state given an initial state. It is possible to predict the system state for a physical process with a precise analytic model. The system model can be used to model the normal behavior of a dynamic physical process, and in the case of anomalies, system behavior can be observed as deviating from the expected.

Figure 1 shows the infrastructure for WADI testbed. The main components are primary grid storage tanks, elevated reservoir and...
sion to the sensor measurements obtained from WADI indicates considering the system state evolution based on the identified state states has an acceptable trade-off between prediction error and order models were also identified. Eventually the model with the matrices are reproduced in Appendix A. A few higher and lower generated using the approximated model, with \( x_k \) written as, \( \hat{x}_k \) denotes the optimal estimate for \( x_k \). Thus, an error can be defined as, \( e_k = \hat{x}_k - x_k \). where \( \hat{x}_{k|i} \) denotes the optimal estimate for \( x_k \) given the measurements \( y_1, ..., y_j \). Let \( P_k \) denote the error covariance, \( \text{Cov}(e_k) = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] \), and \( \hat{P}_{k|i} \) the estimate of \( P_k \) given \( y_1, ..., y_j \). Prediction equation for state variable using Kalman filter can be written as, \( \hat{x}_{k+1|k} = A\hat{x}_{k|k} \) \( P_{k+1|k} = AP_kA^T + Q \), where \( \hat{x}_{k|k} \) is the estimate at time step \( k \) using measurements up to time \( k \) and \( \hat{x}_{k+1|k} \) is \( (k+1)^{th} \) prediction based on previous \( k \) measurements. Similarly, \( \hat{P}_{k|k} \) is the error covariance estimate until time step \( k \). \( Q \) is the process noise covariance matrix. The next step in Kalman filter estimation is time update step using Kalman gain \( L_k \). \[
l_k = P_{k|k-1}C^T(CP_{k|k-1}C^T + R)^{-1}
\] \[
\hat{x}_{k+1|k} = \hat{x}_{k+1|k} + L_k(y_k - C\hat{x}_{k+1|k})
\] \[
\hat{P}_{k+1|k} = [(I - L_kC)\hat{P}_{k+1|k}\]
where \( \hat{x}_{k+1|k} \) and \( \hat{P}_{k+1|k} \) are the updates for the \( k + 1 \) time step using measurements \( y_k \) from the \( k^{th} \) sensor and Kalman gain \( L_k \). \( R \) is the measurement noise covariance matrix. The initial state can be selected as \( x_0 = x_0 \) with \( P_0 = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] \). Kalman gain \( L_k \) is updated at each time step but after a few iterations it converges and operates in a steady state. Kalman filter is an iterative estimator and \( \hat{x}_{k|k} \) in Eqn. (4) comes from \( \hat{x}_{k-1|k} \) in Eqn. (7). It is assumed that the model is a faithful representation of the water distribution network (see Figure 4). The top pane in the figure shows the sensor readings from WADI as well as the estimated output by the model for the water level sensor using system matrices. One may observe that the modeled output is close to the sensor readings resulting in small residual shown in the bottom pane.

4.2 Attack detection framework

In this section, details of the proposed attack detection scheme are described. First, we discuss the Kalman filter based state estimation and residual generation. Next, we present a residual-based attack detection procedure, namely the CUSUM detector.

4.2.1 Kalman filter. To estimate the state of the system based on the available output \( y_k \), we use a linear filter with the following structure:

\[
\hat{x}_{k+1} = A\hat{x}_k + Bu_k + L_k(y_k - C\hat{x}_k),
\]

with estimated state \( \hat{x}_k \in \mathbb{R}^n \), \( y_k = E[x(t_k)] \), where \( E[\cdot] \) denotes expectation, and gain matrix \( L_k \in \mathbb{R}^{n\times m} \). Define the estimation error \( e_k = x_k - \hat{x}_k \). In the Kalman filter, the matrix \( L_k \) is designed to minimize the covariance matrix \( P_k := E[e_k e_k^T] \) (in the absence of attacks).

From (2), we can get an overview of the system model where the Kalman filter is being used for estimation. The estimator makes an estimate at each time step based on the previous readings up to \( x_{k-1} \) and the sensor reading \( y_k \). The estimator gives \( \hat{x}_k \) as an estimate of state variable \( x_k \). Thus, an error can be defined as, \( e_k = \hat{x}_k - x_k \).

Figure 2: Leak setup in WADI.

Figure 3: Process flow diagram of leak setup in WADI.

The system identification problem is to determine the system matrices \( A, B, C \) from input-output data. The model so obtained provides a good fit between measurements and simulated outputs, generated using the approximated model, with 10 states, i.e., \( n = 10 \); the matrices are reproduced in Appendix A. A few higher and lower order models were also identified. Eventually the model with 10 states has an acceptable trade-off between prediction error and its dimension. The quality of the identified model is validated by considering the system state evolution based on the identified state space matrices and initial state \( x_1 \). The proximity of the system evolution to the sensor measurements obtained from WADI indicates that this model is a faithful representation of the water distribution network (see Figure 4). The top pane in the figure shows the sensor readings from WADI as well as the estimated output by the model for the water level sensor using system matrices. One may observe that the modeled output is close to the sensor readings resulting in small residual shown in the bottom pane.
the system is in a steady state before attacks are launched. Kalman filter gain is represented by $L$ in steady state.

4.2.2 Residuals and hypothesis testing. The model based fault detection is common in fault detection literature [39]. Using residual vector for fault detection focuses on a particular structure of the faults. However, an intelligent adversarial attack is challenging to detect. In this study, the focus is on the performance of the residual based anomaly detection scheme against strategically designed attacks, such as for example, replay attacks [6, 25]. In this work, we assess the performance of the CUSUM model-based fault detection procedure for a variety of attacks. These procedures rely on a state estimator, e.g., Kalman filter, to predict system state. The estimated state is compared with sensor measurements $\hat{y}_k$, which may have been compromised. The difference between the two should stay within a certain threshold under normal operation, otherwise, an alarm is triggered. The residual random sequence $r_k, k \in \mathbb{N}$ is defined as

$$r_k := \tilde{y}_k - C\hat{x}_k = C\gamma_k + \eta_k + \delta_k.$$  

If there are no attacks, the mean of the residual is

$$E[r_{k+1}] = CE[\epsilon_{k+1}] + E[\eta_{k+1}] = \tilde{r}_{mx1},$$  

where $\tilde{r}_{mx1}$ denotes an $m \times 1$ matrix composed of mean of residuals under normal operation, and the co-variance is given by

$$\Sigma := E[(r_{k+1} \; r_{k+1}^T)] = CPC^T + R_2.$$  

For this residual, we identify two hypothesis to be tested, $H_0$, the normal mode, i.e., no attacks, and $H_1$, the faulty mode, i.e., with attacks. In our case study, the pressure at the nodes and the water level in the tank are the outputs of the system. The residuals are obtained using this data together with the state estimates. Thus we can state the two hypotheses as follows.

$$H_0 : \begin{cases} E[r_k] = \tilde{r}_{mx1}, \\ E[r_k r_k^T] = \Sigma, \end{cases} \quad \text{or} \quad H_1 : \begin{cases} E[r_k] \neq \tilde{r}_{mx1}, \\ E[r_k r_k^T] \neq \Sigma. \end{cases}$$

In the following, we formulate the hypothesis testing in a more formal manner using existing change detection techniques based on the statistics of the residuals.

4.2.3 Cumulative Sum (CUSUM) detector. The residual vector is given as an input to the CUSUM procedure, also known as the stateful detector. The input to the CUSUM procedure can be considered as a distance measure, i.e., a measure of how far the estimate is from the expected measurements. For this work, all the sensors and actuators are considered together while creating a system model. The Kalman filter gain is presented in form of a matrix as all the sensors together constitute the system outputs. A dedicated detector for each sensor is designed. The index $i$ denotes the sensor/detector, $i \in \mathbb{J} := \{1, 2, \ldots, m\}$, where $m$ is the number of sensors. Thus, the attacked output vector can be partitioned as $\tilde{y}_k = \text{col}(\tilde{y}_{k,1}, \ldots, \tilde{y}_{k,m})$ where $\tilde{y}_{k,i} \in \mathbb{R}$ denotes the $i$-th entry of $\tilde{y}_k \in \mathbb{R}^m$, then

$$\tilde{y}_{k,i} = C_i x_k + \eta_{k,i} + \delta_{k,i}.$$  

with $C_i$ denotes the $i$-th row of $C$, and $\eta_{k,i}$ and $\delta_{k,i}$ denote the $i$-th entries of $\eta_k$ and $\delta_k$, respectively. $\delta_k$ is the value added by the attacker to the sensor measurements to achieve a specific attack objective. The residual vector for each sensor can be given as,

$$r_{k,i} = C_i x_k + \eta_{k,i} + \delta_{k,i}.$$  

The standard CUSUM [26] procedure is explained using the following equations:

\textbf{CUSUM:} $S_{0,i} = 0, \; S_{0,i}^+ = 0, \; \hat{k}_i = 0, \; \tilde{k}_i = 0,$

$$S_{k,i}^* = \max(0, S_{k-1,i}^* + r_{k,i} - \tilde{T}_i - \kappa_i), \quad \text{if} \quad S_{k-1,i}^* \leq \tau_i^*,$$

$$S_{k,i}^* = S_{k-1,i}^* + \kappa_i, \quad \text{if} \quad S_{k-1,i}^* > \tau_i^*.$$  

\textbf{CUSUM:}$S_{0,i} = 0, \; S_{0,i}^+ = 0, \; \hat{k}_i = 0, \; \tilde{k}_i = 0,$

$$S_{k,i}^* = \max(0, S_{k-1,i}^* + r_{k,i} - \tilde{T}_i - \kappa_i), \quad \text{if} \quad S_{k-1,i}^* \leq \tau_i^*,$$

$$S_{k,i}^* = S_{k-1,i}^* + \kappa_i, \quad \text{if} \quad S_{k-1,i}^* > \tau_i^*.$$  

The standard CUSUM [26] procedure is explained using the following equations.
Design parameters: Bias $\kappa_i > 0$; threshold $\tau_i > 0$.
Output: $\text{Alarm}(s) = k_i^+ + k_i^-$. 

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 ($\bar{r}_i$). To tune the CUSUM detector there is also a slack variable $\kappa$ chosen to be $\frac{1}{2} * \sigma_i$ in this study. $\tau_i = \pm \Gamma * \sigma_i$, where $\Gamma$ is a multiplier to the standard deviation ($\sigma$) and usually taken between 3 and 5 [26]. 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 zero 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 $A^+_t$, $A^-_t \in [0, 1]$ denotes the false alarm rate for the CUSUM procedure defined as the expected proportion of observations which are false alarms [1, 38]. In this study a false alarm rate of 5% is considered under the normal operation. Table 2 shows a list of CUSUM parameters for all the sensors used in this study. Following metrics are used while assessing the effectiveness of the attack detection procedure.

- True Positive Rate (TPR): This rate is defined as the ratio of data detected as attack to the total number of observations when the anomaly actually existed.
- False Positive Rate (FPR): This is defined as the ratio of data detected as attack to the total number of observations when the plant was under normal operation.

I n d e x e d, F P R s h o u l d b e a s m a l l a s p o s s i b l e a n d T P R a s h i g h a s p o s s i b l e. B o th T P R a n d F P R b e i n g r a t i o s r a n g e b e t w e e n 0 a n d 1.

### Table 2: CUSUM design parameters

<table>
<thead>
<tr>
<th>Sensors</th>
<th>Design parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Bias ($\kappa$)</td>
</tr>
<tr>
<td>1_LT_001</td>
<td>0.1231</td>
</tr>
<tr>
<td>2_LT_002</td>
<td>0.0873</td>
</tr>
<tr>
<td>2_PIT_001</td>
<td>0.106</td>
</tr>
<tr>
<td>2_PIT_002</td>
<td>0.1412</td>
</tr>
<tr>
<td>1_FIT_001</td>
<td>0.0186</td>
</tr>
<tr>
<td>2_FIT_001</td>
<td>0.0361</td>
</tr>
<tr>
<td>2_FIT_002</td>
<td>0.007</td>
</tr>
<tr>
<td>2_FIT_003</td>
<td>0.0155</td>
</tr>
</tbody>
</table>
were input to the system model obtained using subspace system identification as discussed in Section 4. Based on the system model the residual ($r_i$) for each sensor ($i$) was generated using the Kalman filter based state estimation. Figure 6a shows the residual vector under normal operation of the plant for the pressure sensors. For the sake of brevity, plots are omitted for the remainder of the sensors; a complete list of threshold and bias selection for CUSUM is included in Table 2. These residuals are tested against the calculated thresholds for CUSUM procedure as explained in the previous section.

During normal operation, False Positive (FP) rates ($A_i^+$) are shown in Table 3. Three runs of normal operation were experimented with to validate the system model. As there were no attacks, any alarm raised by the CUSUM detector is considered as a false positive. For example in Run 1 for 1_LT_001, out of 7200 data points, 27 points (0.0379) are flagged as alarms. It is observed that these experiments have similar FP rates for different runs. Under normal operation, this false alarm rate validates the stability of system model and the CUSUM based detector.

Table 3: False positive rate under normal experiments

<table>
<thead>
<tr>
<th>Sensors</th>
<th>False Positive Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Run 1</td>
</tr>
<tr>
<td>1_LT_001</td>
<td>0.0379</td>
</tr>
<tr>
<td>2_LT_002</td>
<td>0.0372</td>
</tr>
<tr>
<td>2_PIT_001</td>
<td>0.0104</td>
</tr>
<tr>
<td>2_PIT_002</td>
<td>0.1145</td>
</tr>
<tr>
<td>1_FIT_001</td>
<td>0.0089</td>
</tr>
<tr>
<td>2_FIT_001</td>
<td>0.0193</td>
</tr>
<tr>
<td>2_FIT_002</td>
<td>0.0609</td>
</tr>
<tr>
<td>2_FIT_003</td>
<td>0.0582</td>
</tr>
</tbody>
</table>

5.5 Leak execution (Naive attack)

A naive attacker performs an attack to steal water from one of the main pipeline sections. Since the attacker has only physical access to a portion of the plant, i.e., a water distribution pipe, the attacker executes four different types of attacks in an effort to remain undetected. Note that valve 2_MCV_007 can be used to cut-off water supply. Steps described in Part (a) of Figure 5 are used for introducing a leak in the plant.

Before starting the leak process, for the sake of conserving water, another valve labeled as 2_MV_008 (shown in Figure 2) is kept open so that the leaked/stolen water can drain into the return water grid of the plant. The idea is to capture the behavior of a physical attacker. An attacker can choose to cut off water supply by opening or closing the valve 2_MCV_007 in a gradual or abrupt manner. The change could be observed by the operator or using pre-installed fault detectors. Thus, an attacker can launch any one of the following attacks labeled as L1, L2, L3 or L4. For each of these attacks, the CUSUM based attack detection is used to detect the abnormal behavior of the system.

1) Gradual increase and abrupt decrease (L1): In this attack scenario, the leak process begins by gradually opening the valve 2_MCV_007. The position of the valve was increased by 5% every 60 seconds until it reached 95%. Once it reached 95%, the system was run for 10 minutes and then the valve closed fully and abruptly.

2) Gradual increase and gradual decrease (L2): This experiment was performed by gradually opening valve 2_MCV_007. The position of the valve was increased by 5% every 60 seconds until it reached 95%. At this position, the system was allowed to run for 10 minutes. The valve was then closed gradually i.e., the position was decreased by 5% every 60 seconds until it reached 0%.

3) Abrupt increase gradual decrease (L3): In this experiment the valve (2_MCV_007) was opened abruptly, i.e. the position of the valve (2_MCV_007) set to 100%. The plant was left running for 10 minutes and then the valve was closed gradually. The position was decreased by 5% every 60 seconds until it reached 0%, i.e., until completely closed.

4) Abrupt increase and abrupt decrease (L4): In this leak experiment, the leak process started by opening the valve (2_MCV_007) completely, i.e. the position of valve 2_MCV_007 was set to 100%. The system continued to run for 10 minutes in this state and then the valve was immediately closed completely, i.e, the position of the valve (2_MCV_007) set to 0%.

Results and discussion. Table 4 shows the FPs and TPs for the four leak experiments. It can be seen that sensors 2_PIT_001, 2_PIT_002, 2_FIT_001, and 2_FIT_002 show the high true positive rate. This is due to the fact that these sensors are upstream and downstream from the leak point and are most affected by the leak. There are other sensors in the table which were never affected by the attack, for example, level sensors 1_LT_001 and 2_LT_002. However, these level sensors are included for comparison with the sensors under attack. The affected sensors also show a slightly higher FPs due to the data points that are physically located following the leak attack and carry the effect forward. For example, in experiment L1, the attack start and end points are at $k = 2103$ and $k = 3969$, respectively. Where, $k$ is the time at which sensor measurements are recorded. Each sensor measurement is stored at each second. The calculated values of FP and TP for 2_PIT_001 are 0.3641 and 0.8757, respectively. Visual impact of an attack is shown in Figure 6. By comparing Figure 6a (normal operation) with Figure 6b (under attack), one could observe that the sensor residuals deviate significantly from the normal thresholds.

Though the attacker attempts to steal water in four different ways, the attack detection mechanism is able to detect all attacks. The TP and FP alarm rates are found to be identical in all these cases. Therefore, it can be concluded that different procedures for performing the physical attacks has the same impact on the attack detection model. As it is not possible to successfully hide the physical attack alone, the attacker proceeds to perform a cyber attack simultaneously as explained in the next section.

5.6 Leak attacks with replay

Two attacks were executed in this set of experiments. In the first attack, referred to as Replay_1 and Replay_2, the attacker launches a physical attack on the system to steal water while simultaneously launching a cyber-attack to spoof the sensor measurements. The
Figure 5: Steps in performing leak and replay attack experiments.

(a) Normal operation: Residual estimation for Run 2 of the normal experiments. Figure also shows the upper and lower limits of the CUSUM detector for each sensor.

(b) Residual under Leak execution

Figure 6: Comparison of residual estimates between normal and naive attack experiment L2.
Can Replay Attacks Designed to Steal Water from Water Distribution Systems Remain Undetected? Conference’17, July 2017, Washington, DC, USA

Table 4: Alarm rates under leak experiments

<table>
<thead>
<tr>
<th>Sensors</th>
<th>L1</th>
<th></th>
<th>L2</th>
<th></th>
<th>L3</th>
<th></th>
<th>L4</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FPR</td>
<td>TPR</td>
<td>FPR</td>
<td>TPR</td>
<td>FPR</td>
<td>TPR</td>
<td>FPR</td>
<td>TPR</td>
</tr>
<tr>
<td>1_LT_001</td>
<td>0.0038</td>
<td>0.0086</td>
<td>0.0811</td>
<td>0.0721</td>
<td>0.0026</td>
<td>0.0019</td>
<td>0.0028</td>
<td>0.0052</td>
</tr>
<tr>
<td>2_LT_002</td>
<td>0.0064</td>
<td>0.0112</td>
<td>0.0472</td>
<td>0.0527</td>
<td>0.0063</td>
<td>0.0142</td>
<td>0.0056</td>
<td>0.0117</td>
</tr>
<tr>
<td>2_PIT_001</td>
<td>0.3641</td>
<td>0.8757</td>
<td>0.2192</td>
<td>0.8023</td>
<td>0.2763</td>
<td>0.7389</td>
<td>0.2845</td>
<td>0.6715</td>
</tr>
<tr>
<td>2_PIT_002</td>
<td>0.3979</td>
<td>0.9738</td>
<td>0.4303</td>
<td>0.8768</td>
<td>0.3451</td>
<td>0.9062</td>
<td>0.2979</td>
<td>0.9935</td>
</tr>
<tr>
<td>1_FIT_001</td>
<td>0.0160</td>
<td>0.0204</td>
<td>0.0729</td>
<td>0.0235</td>
<td>0.0156</td>
<td>0.0210</td>
<td>0.0142</td>
<td>0.0262</td>
</tr>
<tr>
<td>2_FIT_001</td>
<td>0.2328</td>
<td>0.0142</td>
<td>0.0262</td>
<td>0.0165</td>
<td>0.3804</td>
<td>0.5298</td>
<td>0.2753</td>
<td>0.7353</td>
</tr>
<tr>
<td>2_FIT_002</td>
<td>0.3560</td>
<td>0.8891</td>
<td>0.3994</td>
<td>0.6033</td>
<td>0.2802</td>
<td>0.76501</td>
<td>0.2874</td>
<td>0.7227</td>
</tr>
<tr>
<td>2_FIT_003</td>
<td>0.2566</td>
<td>0.4826</td>
<td>0.2569</td>
<td>0.7727</td>
<td>0.2355</td>
<td>0.4587</td>
<td>0.2194</td>
<td>0.6082</td>
</tr>
</tbody>
</table>

Figure 7: Residual estimates under Replay_1 attack

cyber attack was launched as a replay attack on the sensor measurements. The attacker compromised a set of sensors, observed and recorded their readings for a certain time duration and replayed the recorded measurements while carrying out the attack. These attacks were performed strategically to avoid leak detection. The target replay attack points were chosen based on the attacker’s knowledge of the process. Initially, the attacker chooses the upstream pressure sensor, namely 2_PIT_001 and the downstream pressure sensor (2_PIT_002), to perform the replay attack (Replay_1). Then the attacker launches the replay attacks on both pressure sensors and the flow sensor (2_FIT_002) labeled as Replay_2 in Table 5.

The replay attack procedure on sensors is as follows. As shown in part (b) of Figure 5, initially, the replay attack is launched at time $k_s$. This is followed by the launch of the actual physical attack at time $k_s + \Delta k$, where $\Delta k$ is the time difference between the launch of replay and leak attacks. These attacks are kept running simultaneously until time $k_e$, where $k_e$ denotes the end of the leak attack. Even after the leak attack has ended, the attacker continues to replay the sensors for a duration of $k_{af}$ to hide the after-effect of the attack.

Results and discussion. The alarm rates for the replay attacks are listed in Table 5. From the table it is observed that even after performing the replay attack on sensors 2_PIT_001, 2_PIT_002, and 2_FIT_002, the alarm rates are high implying that the CUSUM based detector can detect the attacks. This is due to the fact that the attacker chooses to replay the sensor measurements. However, the model considers actuation signals as input and sensor measurements as outputs as mentioned in Section 4. A visual inspection of Figures 7 and 8 reveals the residual vector of sensors during different replay attack scenarios. From this figure it can be seen that the residuals are out of bounds of the normal operation. As a result, the attacker cannot remain undetected by only replaying the sensor measurements. It can be observed from the results so far that replaying only sensor data is not adequate to hide cyber and physical attacks from the model based detectors.

Table 5: Alarm rates under replay experiments

<table>
<thead>
<tr>
<th>Sensors</th>
<th>Replay_1</th>
<th></th>
<th>Replay_2</th>
<th></th>
<th>Replay_3</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FPR</td>
<td>TPR</td>
<td>FPR</td>
<td>TPR</td>
<td>FPR</td>
<td>TPR</td>
</tr>
<tr>
<td>1_LT_001</td>
<td>0.0461</td>
<td>0.0689</td>
<td>0.0504</td>
<td>0.0815</td>
<td>0.0311</td>
<td>0.0625</td>
</tr>
<tr>
<td>2_LT_002</td>
<td>0.0427</td>
<td>0.0729</td>
<td>0.0295</td>
<td>0.0302</td>
<td>0.0243</td>
<td>0.0162</td>
</tr>
<tr>
<td>2_PIT_001</td>
<td>0.2347</td>
<td>0.6889</td>
<td>0.1598</td>
<td>0.4081</td>
<td>0.0822</td>
<td>0.2196</td>
</tr>
<tr>
<td>2_PIT_002</td>
<td>0.2416</td>
<td>0.8374</td>
<td>0.3250</td>
<td>0.7238</td>
<td>0.1736</td>
<td>0.0336</td>
</tr>
<tr>
<td>1_FIT_001</td>
<td>0.0648</td>
<td>0.0485</td>
<td>0.0813</td>
<td>0.0633</td>
<td>0.0033</td>
<td>0.0037</td>
</tr>
<tr>
<td>2_FIT_001</td>
<td>0.2718</td>
<td>0.7066</td>
<td>0.3325</td>
<td>0.5611</td>
<td>0.0617</td>
<td>0.1079</td>
</tr>
<tr>
<td>2_FIT_002</td>
<td>0.2064</td>
<td>0.6638</td>
<td>0.3184</td>
<td>0.5448</td>
<td>0.3320</td>
<td>0.3727</td>
</tr>
<tr>
<td>2_FIT_003</td>
<td>0.1645</td>
<td>0.6330</td>
<td>0.1503</td>
<td>0.4435</td>
<td>0.1299</td>
<td>0.0772</td>
</tr>
</tbody>
</table>
5.7 Powerful attack

A powerful attacker with complete knowledge of the system model and detection mechanism can identify the sensors and actuators on which replay attack needs to be launched. In the previous section we observed that only replaying sensor measurements could not hide the attack from a CUSUM detector. A powerful attacker with complete knowledge of system model, inputs, and outputs, should launch a replay attack not only on sensor (output) measurements but also on the actuators (valves). For this experiment, an attacker launches a replay attack on Modulating Control Valves (MCVs), together with the sensor measurements, as described in Section 5.6.

MCVs in WADI control the input flow to the consumer tanks. The opening position of an MCV is decided based on the current consumer demand. The system model uses this information as one of the inputs. Therefore the attacker chooses to launch replay on these valves, together with other sensors, to mask the attack. Figure 9 shows the residual estimates when the replay is performed on sensors and actuators. It is observed that the residuals do not deviate from the normal operation and are within the thresholds of the detector. Therefore the attack is not detected. Also, the FP and TP rates are low as compared to other two attack scenarios as shown in column labelled “Replay_3” in Table 5. However, it is important to mention that the false alarm rate for the pressure sensors is quite high even during the normal operation as shown in Table 3. The idea is to show how sensor instability can cause issues in designing the detectors. However, it is recommended to design process specific heuristics, for example, based on the frequency of false alarms to reduce the false positive rate and make it viable for the production in the field.

The attacker remains undetected by executing such a powerful attack. There exists a strong interdependence between MCVs and sensor measurements. Therefore, in this attack scenario, the attacker succeeds in stealing water without being detected.

6 SUMMARY, CONCLUSIONS AND FUTURE WORK

6.1 Summary

Replay attacks were launched on an operational water distribution system (WADI) to understand the effectiveness of a model-based method in detecting such attacks aimed at stealing water. Three attack scenarios, namely naive, cyber-physical, and powerful attacks were considered. From the experiments, it was observed that the powerful attacker succeeds in stealing water without being detected. It is also observed that if the system model was developed using only the sensor measurements, then the cyber-physical attacker can easily remain undetected. However, as the model is built...
using both sensor and actuator measurements, the cyber-physical attacker fails to mask the attack.

6.2 Conclusions
Analysis of data acquired during experimentation on WADI offers evidence in support of the following claims. (a) The LTI and CUSUM framework is effective in detecting attacks launched by an attacker with incomplete knowledge of the system and of the detection mechanism, and (b) an attacker could remain undetected, and successfully steal water, when the attacks are designed strategically using a complete knowledge of the water distribution system and the detection framework. The experimental evidence does point to the need for a more advanced method for detecting attacks aimed at stealing water in water treatment systems coupled with design enhancements that allow better detection rates.

6.3 Future Work
Effects of randomness in user demands: Consumers’ demands are kept constant in this study to avoid variation in the data due to variations in the consumer demand and to highlight the replay attack. Random consumer demands make it harder for the attacker to launch a successful replay attack. One can use this randomness to deceive an attacker and make it even harder to launch a replay attack.

Sensor ageing effects: Another issue is sensor/actuator wear and tear. Although this issue is not considered in the study reported here, it can play a significant role in similar future studies.

Speed of detection: The standard CUSUM method was used such that when an attacker makes a small change in a particular direction of the process, it will take some time for the effect to accumulate and cross the CUSUM threshold. There have been works to detect the changes faster than the standard CUSUM detector [34]. The quickest change detection with observation scheduling in the minimax setting can be used in future work to include the speed of detection as a parameter.

ACKNOWLEDGMENTS
This work was supported in part by the National Research Foundation (NRF), Prime Minister’s Office, Singapore, under its National Cybersecurity R&D Programme (Award No. NRF2015NCR-NCR003-001) and administered by the National Cybersecurity R&D Directorate.

REFERENCES
A STATE SPACE MATRICES FOR THE SYSTEM MODEL

In what follows, we present the state space matrices \( A, B, C \) and Kalman gain \( K \), obtained using subspace system identification. A 10th order model is used. Therefore, we have system matrix \( A \) as a 10 x 10. For a 15 inputs, matrix dimensions for \( B \) are 10 x 15. For 8 outputs, the dimensions for matrix \( C \) are 8 x 10. Using these state space matrices and system model of (1), one can find the dynamics of the system evolution.

\[
A = \begin{bmatrix}
0.995 & -1.171 \times 10^{-4} & 8.262 \times 10^{-3} & -2.54 \times 10^{-5} & -0.0015 & 8.718 \times 10^{-3} & -0.000117 & -0.000024 & 0.999 & -1.412 \times 10^{-4} \\
-0.0000124 & 0.999 & -1.412 \times 10^{-4} & 6.892 \times 10^{-3} & -2.42 \times 10^{-5} & -2.72 \times 10^{-3} & 0.0015 & -0.0038 & 7.641 \times 10^{-4} & 0.997 & 0.0033 & 0.0078 & -0.0319 & 0.1475 \\
0.0000079 & 7.47 \times 10^{-4} & -2.13 \times 10^{-3} & 0.936 & 1.982 \times 10^{-4} & 0.0073 & -0.0410 & 0.0000007 & 0.0026 & 0.0063 & 0.0178 & -0.0260 & 0.9237 & 0.0210 \\
0.0053 & -0.0045 & -0.0249 & 0.0491 & 0.0052 & 0.0077 & 0.885 & 0.0064 & -0.0035 & -0.0169 & 0.0489 & 0.0030 & 0.0148 & -0.0762 \\
-0.0223 & -0.0152 & -0.1805 & 0.1656 & 0.2589 & -0.0895 & -0.1766 & -0.00765 & 0.0024 & 0.0303 & 0.0179 & -0.019 & -0.0326 & 0.175
\end{bmatrix}
\]
Can Replay Attacks Designed to Steal Water from Water Distribution Systems Remain Undetected?

Conference'17, July 2017, Washington, DC, USA

\[ C = \begin{bmatrix}
-475.04 & -3976.80 & 46.14 & 21.01 & 1.82 & 1.95 & 0.34 & 2.11 & -0.12 & -0.26 \\
1519.69 & -102.13 & -17.89 & 4.54 & 3.000 & 0.344 & -1.33 & 0.14 & 0.65 & 1.80 \\
348.066 & -908.87 & 609.891 & -3742.44 & -19.061 & -426.50 & 2586.86 & -646.60 & -29.84 & -251.44 \\
372.86 & -569.67 & 225.67 & -4452.78 & -44.44 & -168.80 & 1002.75 & 319.95 & -22.82 & -192.38 \\
-1.407 & -1.08 & -0.703 & 0.307 & -7.706 & -42.83 & -6.95 & -6.042 & 0.62 & -0.012 \\
10.648 & 0.57 & -0.99 & 2.01 & -84.934 & 14.11 & 3.13 & 0.13 & -8.45 & 0.12 \\
-1.86 & -3.66 & -31.197 & -8.68 & -0.0443 & -0.377 & 2.38 & -4.58 & -0.032 & -0.59 \\
2.30 & 12.54 & 0.23 & 102.90 & 1.62 & -0.59 & 1.68 & 59.57 & 0.388 & 6.28 \\
\end{bmatrix} \]

\[ K = \begin{bmatrix}
-1.02 \times 10^{-5} & 0.000319 & -2.03 \times 10^{-5} & 8.47 \times 10^{-5} & -0.000103 & -2.34 \times 10^{-5} & -6.43 \times 10^{-5} & 2.26 \\
-8.66 \times 10^{-5} & -3.36 \times 10^{-5} & -4.67 \times 10^{-5} & 7.71 \times 10^{-5} & -8.57 \times 10^{-5} & -5.35 \times 10^{-5} & -2.76 \times 10^{-5} & 2.34 \\
-7.05 \times 10^{-5} & 4.16 \times 10^{-5} & -0.000435 & 0.000695 & -0.00514 & -0.00133 & -0.00436 & 0.0 \\
1.29 \times 10^{-5} & -8.43 \times 10^{-5} & 1.71 \times 10^{-5} & -0.000105 & 0.000173 & 0.000154 & 0.00118 & 0.0 \\
-0.000219 & -0.0001582 & 3.77 \times 10^{-5} & 0.000126 & -0.00248 & -0.00427 & 0.00256 & 0.0 \\
0.0001308 & 4.38 \times 10^{-5} & -5.63 \times 10^{-5} & -5.99 \times 10^{-5} & -0.0121 & 0.000444 & -0.00192 & -0.0 \\
4.17 \times 10^{-5} & -9.79 \times 10^{-5} & 0.000341 & -0.000183 & -1.97 \times 10^{-5} & 0.000251 & 0.00246 & 0.0 \\
6.64 \times 10^{-5} & -0.000446 & -0.000273 & 0.000525 & 0.00269 & 8.80 \times 10^{-5} & 0.00376 & 0.0 \\
-0.000581 & 0.00117 & 0.000173 & 0.000429 & -0.00496 & -0.00931 & -0.00938 & -0.0 \\
0.000131 & 0.000527 & -0.000214 & 2.90 \times 10^{-5} & -0.0129 & -0.0010 & -0.0025 & -0.0 \\
\end{bmatrix} \]

\[ B = \begin{bmatrix}
-0.130464 & -0.000019 & -0.000200 & -0.000020 & -0.000107 & -0.000610 & -0.000020 & -0.000107 & -0.000610 & -0.000020 \\
0.000044 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
0.000041 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
0.000041 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
0.000041 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
0.000041 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
0.000041 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
0.000041 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
0.000041 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
0.000041 & 0.000017 & -0.000218 & 0.000019 & -0.000948 & -0.000602 & 0.000019 & -0.000948 & -0.000602 & 0.000019 \\
\end{bmatrix} \]