BATtle of the Attack Detection ALgorithms (BATADAL)

Annual Water Distribution Systems Analysis Symposium
Sacramento, California, U.S.A.
May 21-25, 2017

`http://www.ewricongress.org` **and** `http://www.batadal.net`

Detailed Problem Description and Rules—September 9, 2016

Riccardo Taormina	Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372 Email: riccardo_taormina@sutd.edu.sg
Stefano Galelli	Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372 Email: stefano_galelli@sutd.edu.sg
Nils Ole Tippenhauer	Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372 Email: nils_tippenhauer@sutd.edu.sg
Avi Ostfeld	Technion – Israel Institute of Technology, Haifa 32100, Israel Email: ostfeld@tx.technion.ac.il
Elad Salomons	OptiWater, 6 Amikam Israel st., Haifa, 3438561, Israel Email: selad@optiwater.com

Introduction

Modern Water Distribution Systems (WDS) rely on computers, sensors and actuators for both monitoring and operational purposes. This combination of physical processes and embedded systems—cyber-physical systems, in short—improves the level of service of water distribution networks but exposes them to the potential threats of cyber attacks. During the past decade, several water supply and distribution systems have been attacked, with the consequent creation of cyber-security agencies and international partnerships to defend water networks. Yet, little is known about the potential effect of these attacks as well as the design and implementation of attack detection algorithms—which identify anomalous behaviors of sensors, pumps and other components of water networks.

1 Approach and Schedule

The BATtle of the Attack Detection ALgorithms (BATADAL) will objectively compare the performance of algorithms for the detection of cyber attacks in water distribution systems. Participants will contribute an attack detection algorithm for a given water network following a set of rules (outlined below) that determine the exact goal of the algorithms. The algorithm development and testing will follow a phased approach. Participants will be first given two datasets characterized by the absence/presence of cyber attacks. These two datasets are to be used for the development of the detection algorithms. Then, a third dataset (the test dataset) will be shared for a period of about one week, during which the participants will use their algorithms to produce a detection report. The reports will be used by the organizers to rank the performance of all algorithms, which will be presented during the Annual Water Distribution Systems Analysis Symposium—to be held in Sacramento, California, U.S.A., May 21-25, 2017. The sessions will be followed by a panel discussion. A jointly authored journal publication will be prepared to archive the challenge and contributed solutions. The schedule of events for BATADAL is outlined in Table 1.

Date	Event
May 26, 2016	Initial announcement at EWRI 2016
September 9, 2016	Publication of problem details and competition rules + Release of the first dataset (with no attacks) + Release of the second dataset (with attacks)
October 2, 2016	Submission of abstracts to EWRI 2017 by participants
January 9, 2017	Submission of conference paper
February 20, 2017	Release of test dataset (unlabeled attacks)
February 27, 2017	Submission of detection report by participants
February 28, 2017	Release of labels for the test dataset
March 13, 2017	Submission of revised conference paper (to be confirmed by EWRI 2017 organizers)
May 21-25, 2017	Public presentation of results at EWRI 2017
July 1, 2017	Development of a jointly authored journal manuscript

Table 1: Schedule of events

2 How to participate

The problem data are available at `http://www.batadal.net`. Interested participants should contact the organizers (see above contact information) to receive a username and password—the data section of BATADAL website is password-protected. Each participating team must submit an on-line abstract for EWRI 2017 conference (`www.ewricongress.org`) that discusses briefly the proposed approach (e.g., machine learning techniques, Kalman filer-based approaches, CUSUM-based methodologies etc.). When submitting the abstract, the topic area must be identified as "Water Distribution Systems Analysis Symposium–BATtle of the Attack Detection ALgorithms (BATADAL)"—this will identify your team as a participant in the

competition.

Each team must summarize its final results in a conference paper, which must be uploaded to the same website no later than January 9, 2017. All conforming results will be included in the public presentation at the conference and will be published as part of the conference proceedings. Submitted papers should be brief and to the point. It is not necessary to describe the competition, as that will be included in the summary comparison paper. Each paper should briefly state that it is part of the BATtle of the Attack Detection ALgorithms (BATADAL) and include the following sections: Abstract; Introduction (brief); Methodology; Summary of results for the first two datasets (see below); Conclusions; and References.

The submission of the detection report based on the test dataset is compulsory—reports submitted with incomplete information may be excluded from the comparison. Note that the test dataset released on February 20, 2017, will not include labeled attacks, and will be used for ranking the designed algorithms. Once all detection reports have been received, the attack labels will be released. This will give the option to the participants to submit an optional revised version of their paper that includes the results on the test dataset (to be confirmed by EWRI 2017 organizers).

3 Design challenge

C-Town Public Utility (CPU) is the main water distribution system operator of C-Town (Fig. 1). For many years, CPU has operated a static distribution topology. In the last year, CPU has introduced novel smart technology to enable remote data collection from sensors in the field, and remote control of actuators. Shortly after that new technology has been introduced, anomalous low levels in Tank T5 and high levels in Tank T1 were observed. A month later, a water overflow in Tank T1 occurred. While CPU personnel at the control center were able to see the anomalous readings for the first two episodes, Tank T1 overflow took place unexpectedly while the water level readings were always below the alarm thresholds and pumping operations appeared to be normal. Searching for the causes, CPU engineers suspect potential cyber-attacks for all these episodes. In particular, they are considering adversaries that are able to activate and deactivate the actuators in C-Town, as well as altering the readings of the sensors deployed in the network and the reported status of actuators, and interfering with the connections established between networked components. The participants task is thus to develop an online alert system for cyber-physical attacks.

3.1 Development data

CPU will provide the participants with the following material.

C-Town model (EPANET ".inp" input format)

C-Town (Fig. 1) is based on a real-world medium-sized network. Water consumption is fairly regular throughout the year with no seasonal variations. Water storage and distribution across the demand nodes is guaranteed by seven tanks, whose water levels trigger the operations of one valve and eleven pumps distributed in five pumping stations (S1-S5). Pumps, valves and tank water level sensors are connected to nine PLCs (Programmable Logic Controller), which are located in proximity of the hydraulic components monitor/control. C-Town has a Supervisory Control And Data Acquisition (SCADA) system that collects the readings from all PLCs and coordinates the operations of the entire network. Table 2 reports the water level sensors and the hydraulic actuators controlled by each PLC. Most of the PLCs controlling the pumps are not directly connected to the water level sensors employed in the control logic, but receive the necessary information via other PLCs. Each PLC controlling a given actuator also reads its status (ON/OFF or OPEN/CLOSED), the flow passing through it, and the suction and discharge pressures.

Historical SCADA data

The following data on historical SCADA operations are provided.

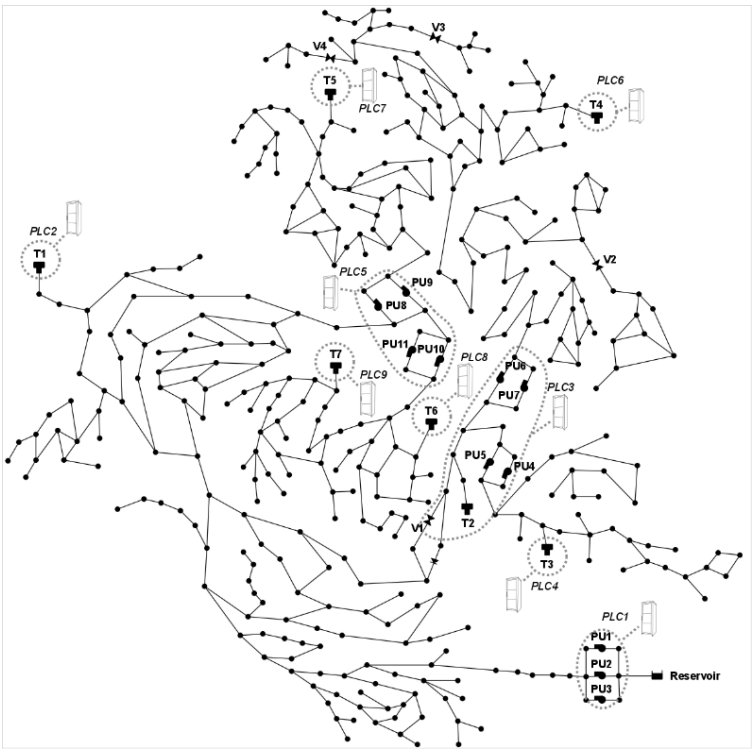

Figure 1: Graphical representation of C-Town water distribution system.

- First set: Data from six months preceding the installation of the smart devices (September 2015 to February 2016). These data are guaranteed to be without attacks and can be used to study the normal system operations. (Data availability: September 9, 2016.)
- Second set: A few months of data following the installation of the smart devices (April to June 2016). This dataset contains the attacks causing anomalous low levels in T5 (13-18 May) and high levels in T1 (21-22 May). CPU engineers were able to discover these attacks and label them properly. The last month includes the attack that caused Tank T1 to overflow. The engineers were able to label the attack only when the overflow occurred (22-23 June). They also suspect that other attacks might be contained in the remaining data of this second dataset. (Data availability: September 9, 2016.) Additional data may be released after the abstract submission (October 2, 2016).
- Test set: This dataset with unlabeled data will be released on February 20 2017. It will be used to quantitatively compare the performance of the algorithms (see Section 3.3).

All data are provided in tabular format with the first column reporting the timestamp of the readings and the remaining columns reporting the value of each different sensor. An additional column contains a binary flag to discriminate normal conditions (flag = 0) from under attack (flag = 1). This column will contain the value -999 for unlabeled data. The available SCADA readings are:

- Water level in each tank;
- Status (0 for OFF/CLOSED, 1 for ON/OPEN) for each pump and valve in the system;
- Flow through each pump and valve;
- Suction pressure and discharge pressure for each valve and pumping station.

The variables are indicated using the prefix L for water level, S for status, F for flow, and P for pressure. The sensor labels (i.e., header of the data files) are formed by linking the variable prefix with the label of a

node/component in the map using the underscore symbol _. For instance, L_T1 indicates the water level in T1, S_PU1 the status of pump PU1, while P_J280 and P_J269 are the suction and discharge pressures of pumping station S1 (as shown in the .inp file, junction J280 and J269 are respectively located at the inlet and outlet of pumping station S1).

PLC	Sensor	Actuators (controlling sensor)
PLC1	-	PU1(T1), PU2(T1)
PLC2	T1	-
PLC3	T2	V1(T2), PU4(T3), PU5(T3), PU6(T4), PU7(T4)
PLC4	T3	-
PLC5	-	PU8(T5), PU9(-), PU10(T7), PU11(T7)
PLC6	T4	-
PLC7	T5	-
PLC8	T6	-
PLC9	T7	-

Table 2: Controlling sensors and controlled actuators attached to each PLC

3.2 Goal of the attack detection algorithm

The primary goal of the detection mechanism is to reliably detect the presence of an ongoing attack from the SCADA readings, and to do so in the shortest amount of time. In addition, the algorithm should avoid false alarms and recognize when the threat is no longer in place. Due to the distributed nature of the WDS, an ideal detection mechanism should also be able to identify which components of the physical network are being attacked in order to facilitate and hasten incident resolution. Furthermore, the inherent interdependence of the elements in the water network should theoretically allow for the detection of anomalies even when the adversary tries to conceal his actions by altering the SCADA readings of one or a few deployed sensors. The control system will operate in the control room, based on the available SCADA data.

3.3 Test data and evaluation criteria

A test dataset will be made available for a short time window (see Table 1), during which the participants are required to run their algorithms and submit a detection report (see Appendix A for further details). The test dataset will contain a few months of data and contain attack instances that may differ from those of the development dataset. All algorithms will then be compared by adopting quantitative criteria based on the time-to-detection and events classification (confusion matrix), as outlined below.

Time-to-detection

The Time-To-Detection (TTD) is the time needed by the algorithm to recognize a threat. It is defined as the difference between the time t_d at which the attack is detected and the time t_0 at which the attack started:

$$TTD = t_d - t_0.$$

The lower the value of TTD, the better the algorithm performs. If the attack is detected, we then have:

$$0 \leq TTD \leq \Delta t,$$

where Δt is the total duration of the attack. If the attack is not detected while it is ongoing (or at all), we set $TTD = \Delta t$.

To facilitate the comparison of all detection algorithms under different attack scenarios, a performance score S_{TTD} will be computed as follows:

$$S_{TTD} = 1 - \frac{1}{n_a} \sum_{i}^{n_a} \frac{TTD_i}{\Delta t_i},$$

where n_a is the number of attacks contained in the test dataset, TTD_i is the time-to-detection relative to the *i*-th attack and Δt_i the duration of the *i*-th attack. S_{TTD} varies between 0 and 1—with $S_{TTD} = 1$ being the ideal case in which all attacks are immediately detected, and $S_{TTD} = 0$ the case in which none of the attacks is detected.

Confusion Matrix

The confusion matrix is a table used to describe the performance of a classifier on a set of data for which the true values are known. The columns of the matrix represent the instances in an actual class while the rows represent the instances in a predicted class. In BATADAL the confusion matrix is employed to assess the performances of the attack detection algorithms using two classes, UNDER ATTACK and SAFE, which yield the 2x2 matrix shown in Fig. 2. The cells of the confusion matrix are defined as follows:

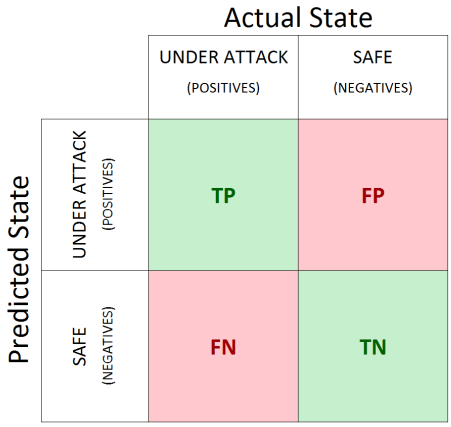

Figure 2: Confusion Matrix

- True Positive (TP): the system is under attack and the algorithm recognizes it.
- True Negative (TN): the system is not under attack and the algorithm recognizes it.
- False Negative (FN): the system is under attack but the algorithm fails to detect it.
- False Positive (FP): the system is not under attack but the algorithm detects a non-existent threat (false alarm).

The comparison across all detection algorithms will be based on the True Positive Rate $TPR = TP/(TP + FN)$—also known as *recall* or *sensitivity*—and the True Negative Rate or *specificity*, defined as $TNR = TN/(FP + TN)$. These metrics will be aggregated into a single score S_{CM} defined as the mean of TPR and TNR—aka Area Under the Curve [1]:

$$S_{CM} = \frac{TPR + TNR}{2}.$$

This measure accounts for both correct detection and false alarms, and is suited when the sampled distribution is biased towards one of the two classes (the SAFE class in this case). S_{CM} varies between 0 and 1—a value of S_{CM} equal to 0.5 corresponds to a naïve detection mechanism that predicts the system to be always in safe or under attack conditions.

Attack localization

Sophisticated detection mechanisms may localize which area of the network has been attacked and identify the targeted components. Such information is precious for preparing an adequate response to resolve

the incident. Having more (correct) details on the attacked area/component can indeed facilitate incident resolution and speed-up resuming of normal operations. Exact localization, however, will not be considered when ranking the designed algorithms.

3.4 Ranking of the submitted solutions

Each team is allowed to submit only one detection report. From the report an overall score will be computed using a combination of S_{TTD} and S_{CM}:

$$S = \gamma \cdot S_{TTD} + (1 - \gamma) \cdot S_{CM}.$$

where the coefficient γ (with $0 \leq \gamma \leq 1$) is used to define the relative importance of the time-to-detection and the confusion matrix criteria. This coefficient is currently set to 0.5—i.e., the two criteria are equally weighted—but the organizers might change its value before the release of the test data to the participants (see Table 1).

3.5 Example of algorithm scoring

This section exemplifies the scoring performed for a single algorithm and a single attack ($n_a = 1$) in a week-long test dataset (168 hours). Fig. 3 shows the comparison between the attack track (in blue) and the detection track (in red) reconstructed from an hypothetical detection report. The attack lasts 40 hours ($\Delta t = 40$) and starts at hour 60 of the test dataset. The algorithm detects the attack after 8 hours ($TTD = 8$) and stops signaling that the system is under attack 5 hours before the actual end of the attack. This yields the following confusion matrix:

$$\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} = \begin{bmatrix} 27 & 0 \\ 13 & 128 \end{bmatrix}$$

for which a value of $S_{CM} = 0.838$ can be computed. The overall score S is:

$$S = \gamma \cdot S_{TTD} + (1 - \gamma) \cdot S_{CM} = 0.5 \cdot (1 - \frac{8}{40}) + (1 - 0.5) \cdot 0.838 = 0.819$$

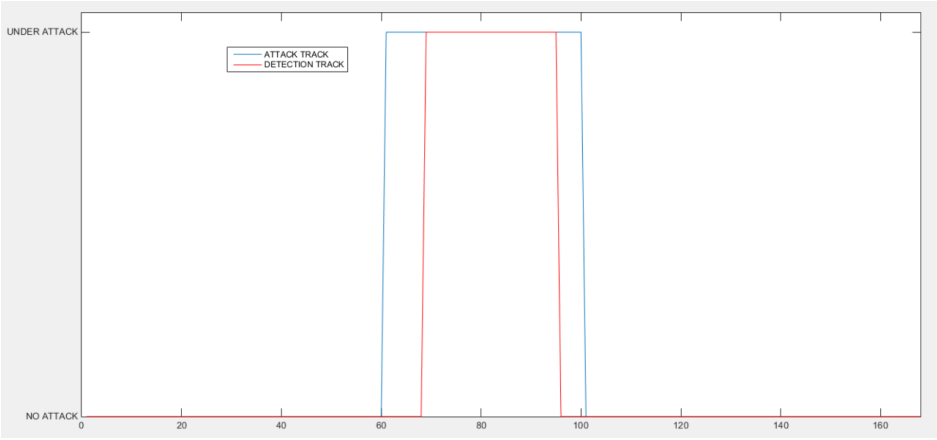

Figure 3: Comparison of attack and detection tracks

References

[1] D. M. Powers, “Evaluation: from Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation,” *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.

[2] R. Taormina, S. Galelli, N. O. Tippenhauer, E. Salomons, and A. Ostfeld, “Assessing the effect of cyber-physical attacks on water distribution systems,” in *Proceedings of World Congress on Environmental & Water Resources 2016*, pp. 436–442, doi: 10.1061/9780784479865.046.

A Appendix

The detection report should be structured as .csv file with as many rows as the number of detected attacks in the test dataset and four columns separated by commas. The first column should contain the attack id specified as an increasing integer starting from 1. The second and third columns will contain the date-time of the beginning and end of the attack, specified using the “DD-MM-YY hh” format. The fourth and last column should contain the labels of the attacked devices and sensors in the network. If the algorithm detects that more than one attacked device/sensor, these should be separated by a space. Figure 4 displays an example of detection report with four reported attacks.

report_example.csv

```
1,10-06-2016 10,13-06-2016 14,T1 PU1 PU2
2,20-06-2016 08,20-06-2016 18,T5
3,02-07-2016 23,03-07-2016 11,PU10 PU11
4,11-07-2016 03,12-07-2016 07,V1
```

Figure 4: Example of detection report