Smart Water Distribution System Communication Architecture Risk Analysis Using Formal Methods

Stefana Krivokuća, Branka Stojanović, Katharina Hofer-Schmitz, Nataša Nešković and Aleksandar Nešković

***Abstract*—Given the rapid development of systems with critical infrastructures, the chances of cyber-attacks are also growing. As the smart water distribution system is one of such structures, security concerns play a significant role in the process of designing them. This paper presents a risk analysis of smart water distribution system example architecture through threat modeling and model checking. The results of the model checking process demonstrate how security requirements can be achieved in a Security-by-Design approach.**

***Keywords*—security, risk analysis, Microsoft Threat Modeling Tool, formal verification, PRISM model checker**

I. Introduction

CONSIDERING current tendencies towards greater automation in cyber-physical systems, many security concerns are surfacing [1]. Water distribution system, an example of CPSs with critical infrastructure, is becoming sensitive to cyber-attacks. In [2] the effects that cyber-attacks can have on a water distribution system are described. In [3] ways of attack detection in these systems are introduced. In [4] a procedure for risk analysis of the smart home automation system through the process of probabilistic model checking [5], is presented.

This paper provides an analysis of the smart water distribution system using probabilistic model checking. The process of threat modeling is automated and it is accomplished using Microsoft Threat Modeling Tool[1].

The paper is organized as follows. Section 2 introduces the system architecture. Section 3 provides vulnerabilities detection using threat modeling. Section 4 provides a risk analysis using formal methods. Section 5 contains the obtained results and the related discussion. Section 6 concludes the paper and points out directions for future work.

This work was funded by the Austrian Federal Ministry of Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK).

Stefana Krivokuća is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11120 Beograd, Serbia (e-mail: stefanakrivokuca96@gmail.com).

Dr. Branka Stojanović is with the JOANNEUM RESEARCH Forschungsgesellschaft mbH, Steyrergasse 17, Graz, Austria (e-mail: branka.stojanovic@joanneum.at).

Dr. Katharina Hofer-Schmitz is with the JOANNEUM RESEARCH Forschungsgesellschaft mbH, Steyrergasse 17, Graz, Austria (e-mail: katharina.hofer-schmitz@joanneum.at).

Prof. dr Nataša Nešković is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11120 Beograd, Serbia (e-mail: natasha@etf.rs).

Prof. dr Aleksandar Nešković is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11120 Beograd, Serbia (e-mail: neshko@etf.rs).

[1] https://docs.microsoft.com/en-us/azure/security/develop/threat-modeling-tool

II. System Architecture

A water distribution system is composed of pumps, reservoir tanks, pipes, and valves that are in charge of water delivery. Various elements are added to these systems, among there are water quality sensors, water level sensors, water pressure sensors, PLCs (Programmable Logic Controllers), and SCADA (Supervisory Control And Data Acquisition). These additional elements allow a more automated mode of operation. As the number of system elements increases, so does the total amount of communication that takes place. This leads to an increase in the number of potential weak points that could be exploited by attackers.

If the attacker happens to get access to a cyber-physical system element or elements, he could harm the overall process. Depending on the interests of the attacker, attacks can vary from data misuse, false alarms, water delivery shut down, tank overflows, or even water contamination.

A smart water distribution system is primarily responsible for the delivery of clean water, as well as for the delivery of the required amount of water. Considering these requirements, the analysis presented in this paper includes the following scenarios:

- Water contamination scenario, and
- Water tank overflow scenario.

III. Detection of System Vulnerabilities Using Threat Modeling

The process of providing security starts with detecting weak points of the system architecture and discovering methods in which they can be exploited.

A. Threat modeling

The threat modeling process gives an overview of how parts of the structure can be abused with an aim of achieving a cyber-attack. This procedure results in a list of threats for the considered smart water distribution system. These threats emphasize system vulnerabilities, therefore, they serve as a starting point for further analysis. Numerous threat modeling methodologies are currently available, among which there are STRIDE, DREAD,

978-1-6654-0499-0/20/$31.00 ©2020 IEEE

PASTA, CAPEC, OWASP, TRIKE, VAST, LINDDUN, Persona Non Grata, etc. The Microsoft Threat Modelling Tool, selected for the proposed research, identifies threats based on the STRIDE method, one of the most commonly used methods in research involving threat modeling [6].

B. WDS threat model

Firstly, a system data flow diagram (DFD) has to be created. Fig. 1 shows the diagram for the smart water distribution system example architecture.

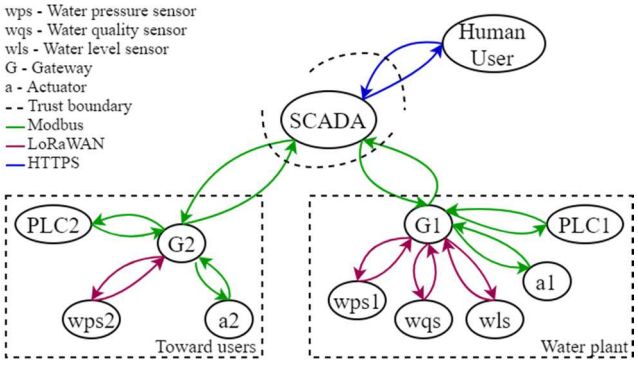

Fig. 1. Data flow diagram - smart water distribution

After creating a DFD, Microsoft Threat Modeling Tool displays a list of threats, and for the proposed example, a list of 228 threats was generated.

The introduced threats point out how different attacks can be carried out by exploiting certain system vulnerabilities. Through the analysis of threats associated to case scenarios of interest, the following vulnerability points were extracted: sensor-gateway links, gateway-actuator links, field gateways, and SCADA.

References [7] and [8] also point out certain security concerns in systems containing these vulnerabilities.

IV. Risk Analysis Using Formal Methods

Security by design is an approach where ensuring security in the system is covered during the system design stage. The results of risk analysis indicate how safety and security requirements can be fulfilled, therefore, it represents an essential part of this approach.

A. Formal verification

Formal verification is a practice that checks whether a certain system works in accordance with existing requirements. In [9] and [10] a review of formal verification methods for diverse protocols used in the IoT environment is presented. A way of achieving formal verification is using model checking [5]. It consists of modeling system abstraction as finite state space and examining it against suitable properties that are defined using temporal logic [11]. Upon completion of model checking quantitative results are acquired and risk assessment is possible.

PRISM model checker [12] is a probabilistic model checker used for systems with probabilistic or non-deterministic behavior and it supports Markov decision processes (MDPs) [13], among others. The PRISM language is based on writing modules that include defining variables and commands for transitions from one state to another.

B. Model generation - Water contamination scenario

Fig. 2 and Fig. 3 show configurations that are analyzed against cyber-attacks regarding the water quality aspect of the system. In light of the non-deterministic nature of cyber-attacks, the system is modeled as MDP within the PRISM model checker.

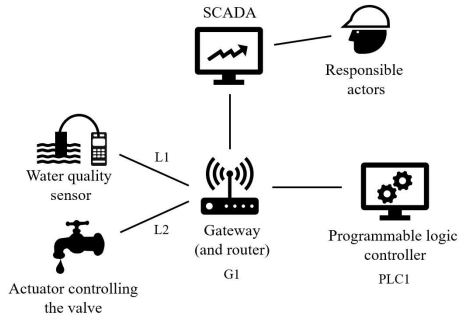

Fig. 2. Water contamination scenario - configuration 1

Configuration 2 contains the same elements as configuration 1 with an additional valve near users that prevents further distribution of contaminated water.

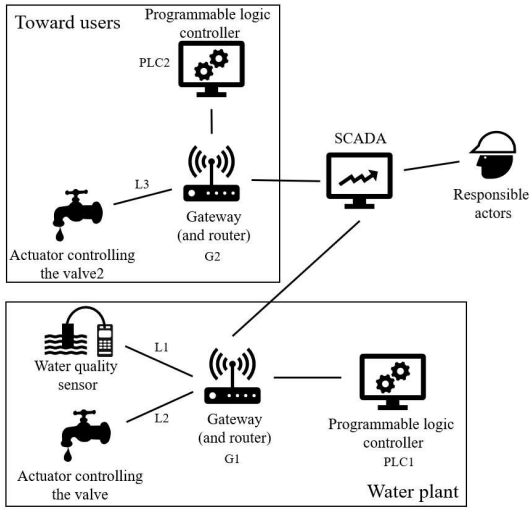

Fig. 3. Water contamination scenario - configuration 2

The presence of an attack usually leaves traces in the system in the form of unusual values. In order to provide security, it is necessary to define an appropriate response in cases these values appear. Consequently, the system model includes the following service policies:

- PLC1 policy: If water contamination is sensed, actuator A1 closes the valve.
- Maintenance policy: If water contamination is sensed, the maintenance team is informed.

Additionally, for configuration 2:

- PLC2 policy: If water contamination is sensed (received value from SCADA), actuator A2 closes the valve2.

In order to carry out an attack, the attacker must manipulate the information flow through the system by exploiting existing vulnerabilities. The process of threat modeling pointed out that system vulnerabilities are

sensor/actuator to gateway links, gateway, and SCADA. These vulnerabilities are included in the process of model generation, as well as characteristic attacker behavior. It is assumed that the attacker knows about existing system vulnerabilities and that in order to carry out an attack, the exploitation of one or more vulnerabilities is necessary. The model implies that the choice of vulnerabilities that are going to be exploited is random. In the event of an attack, the attacker will try to exploit the chosen vulnerability or vulnerabilities and he will succeed with the respective probability for each vulnerability. These values are based on the ones introduced in [4] and [14], for similar device types, and they are shown in Table 1.

TABLE 1. SUCCESSFUL EXPLOITATION PROBABILITIES

Vulnerability	***Exp. prob.***	***Related threat***
Sensor/actuator-gateway links	0.4	Link jammed: Denied Transmission
Gateway	0.4	Modified Observation/ Incorrect Actuation
SCADA	0.2	Modified Observation

Since the attacker's capabilities aren't unlimited, the number of exploitation attempts is restricted. An additional variable 'cost' is introduced and its role is to track the attacker's actions regarding the number of exploitation attempts.

After the system modeling is completed it is necessary to define attack properties against which the model will be tested. Properties are developed using the Probabilistic Computation Tree Logic (PCTL) [11], within the PRISM model checker. A system is considered to be under attack when a set of pre-conditions, an attack property, is fulfilled. By verifying these properties maximum likelihoods of an attack occurrence are obtained.

Table 2 shows properties for attacks with low, medium, and high impact, respectively.

TABLE 2. ATTACK PROPERTIES, WATER CONTAMINATION – CONFIGURATION 1 (CONFIGURATION 2)

Attack	***Attack pre-conditions***
False alarm	Valve (and valve2) remain open and team is called, while water is not contaminated.
Water stop	Valve (or valve2) closes and team is not called, while water is not contaminated.
Water poisoning	Valve (and valve2) remains open, while water is contaminated.

C. Model generation – Tank overflow scenario

For the tank overflow scenario, the modeling of the system is similar, with the difference that the water level aspect is examined, and relevant configurations are shown in Fig. 4 and Fig. 5.

Configuration 2 includes an additional safety valve that prevents overflow by releasing excess water from the tank.

The attacker behavior is included in the model in the same way as for the contamination scenario, for vulnerabilities listed in Table 1.

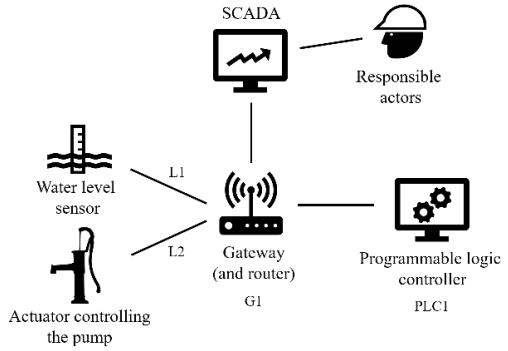

Fig. 4. Tank overflow scenario - configuration 1

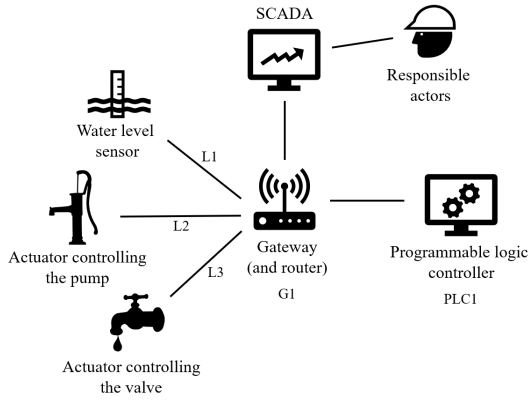

Fig. 5. Tank overflow scenario - configuration 2

Similar to the water contamination scenario, service policies have to be determined, for configuration 1:

- PLC1 policy: If a low water level is sensed actuator A1 turns on the pump.
- Maintenance policy: If a low/high water level is sensed the maintenance team is informed.

And additionally, for configuration 2:

- PLC1 - valve policy: If a high water level is sensed actuator A2 opens the safety valve (the pump remains off). Note: If the water level is optimal, the pump remains off and the valve remains closed.

Attacks considered for tank overflow scenario are shown in Table 3, for attacks with low, medium, and high impact, respectively.

TABLE 3. ATTACK PROPERTIES, TANK OVERFLOW - CONFIGURATION 1 (CONFIGURATION 2)

Attack	***Attack pre-conditions***
False alarm	Team is called and pump doesn't pump water (and valve is closed), while level is optimal (everything is okay, but team is called).
Water stop	Pump doesn't pump water (and valve is closed or pump pumps water and valve is open), all while water level is low.
Tank overflow	Pump pumps water (and safety valve is closed), while the water level is high.

V. RESULTS AND DISCUSSION

The process of model checking is done for water contamination and tank overflow scenarios. For each scenario, two system configurations are introduced and

appropriately modeled, and the comparison of their risk exposure scores is made.

A. Water contamination scenario

Fig. 6 shows verified risk exposure scores for the water contamination scenario.

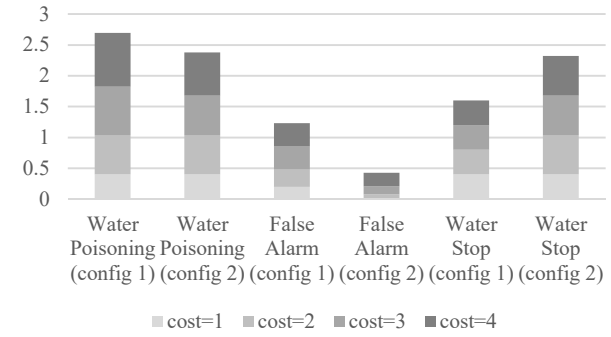

Fig. 6. Risk exposure scores, water contamination scenario

PRISM model checker provided maximum likelihood of occurrence for each attack with a respective cost. Variable 'cost' indicates how many attempts of exploit an attacker has had during an attack. The attacker wants to carry out the attack by exploiting a smaller amount of system vulnerabilities, so the maximum number of exploitation attempts is four (cost=4), as it was proposed in [4].

Config. 1 represents a bigger risk to contamination since there is no safety valve near users that would additionally prevent the delivery of contaminated water. Config. 2 is more sensitive to a potential water stop, due to the incorporation of secondary protective measures. A false alarm is based on changing the value of the water quality observation while other parts of the system work by the fact that there is no contamination. Aside from modified water quality observation, a smaller scale system requires fewer additional pre-conditions, so config.1 is more likely to experience a false alarm.

B. Tank overflow scenario

Fig. 7 shows verified risk exposure scores for the tank overflow scenario.

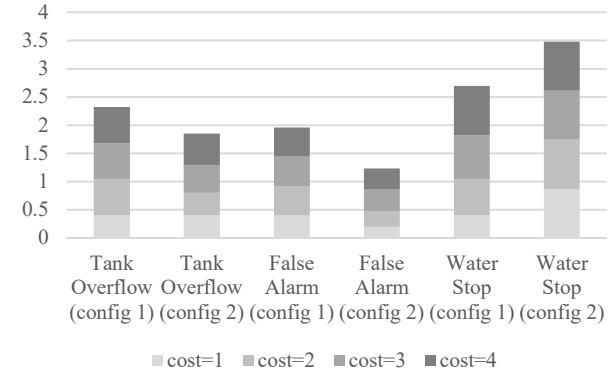

Fig. 7. Risk exposure scores, tank overflow scenario

A scenario with an additional valve that can release excess water, config. 2, imposes less risk to overflow than the one without it. Due to its complexity, config. 2, is more likely to experience a water stop. Similar to the water contamination scenario, a case of false alarm is more likely to occur in a smaller scale system that is config. 1.

VI. Conclusion

This paper presented a smart water distribution system security analysis. The process started with Microsoft Threat Modeling Tool that outlined existing system vulnerabilities. With the knowledge of the weak points of the system, risk analysis using model checking is accomplished for scenarios of water contamination and tank overflow. Results show risk exposure scores and they give an indication of which configurations are recommended considering safety and security concerns of a system with critical infrastructure.

A possibility of further research is exploring ways in which more accurate values of successful exploitation likelihoods of specific system elements can be obtained.

Acknowledgment

We would like to thank JOANNEUM RESEARCH Forschungsgesellschaft mbH for providing an opportunity to conduct this research.

References

[1] A. Humayed, J. Lin, F. Li and B. Luo, "Cyber-Physical Systems Security – A Survey," *IEEE Internet of Things Journal,* vol. 4, no. 6, 2017.

[2] S. Adepu, V. R. Palleti, G. Mishra and A. Mathur, "Investigation of Cyber Attacks on a Water Distribution System," *iTrust Center for Research in Cyber Security, Singapore University of Technology and Design,* 2019.

[3] C. M. Ahmed and J. Ruths, "Model-based Attack Detection Scheme for Smart Water Distribution Networks," *2017 ACM.*

[4] M. Mohsin, M. U. Sardar, O. Hasan and Z. Anwar, "IoTRiskAnalyzer : A Probabilistic Model Checking Based Framework for Formal Risk Analytics of the Internet of Things," *IEEE Access,* pp. 5494-5505, 2017.

[5] A. Bianco and A. de Luca, "Model Checking of Probabilistic and Nondeterministic Systems," *vol 1026. Springer..*

[6] N. Shevchenko, T. A. Chick, P. O'Riordan, T. P. Scanlon and C. Woody, "Threat Modeling: A Summary of Available Methods," *Software Engineering Institute | Carneige Mellon University,* 2018.

[7] J. Slay and M. Miller, "A Security Architecture for SCADA Networks," *ACIS 2006 Proceedings. Paper 12.,* 2006.

[8] K. Khan, W. Goodridge and D. Ragbir, "Security in Wireless Sensor Networks," *Global Journal of Computer Science and Technology Network, Web & Security,* vol. 12, no. 16, 2012.

[9] K. Hofer-Schmitz and B. Stojanović, "Towards Formal Verification of IoT Protocols: A Review," *Computer Networks,* p. 107233, 2020.

[10] K. Hofer-Schmitz and B. Stojanović, "Towards Formal Methods of IoT Application Layer Protocols," *CMI Conference on Cybersecurity and Privacy (CMI) (pp. 1-6). IEEE.,* 2019.

[11] H. Hansson and B. Jonsson, "A Logic for Reasoning about Time and Reliability," *Formal Aspects of Computing 6, 512–535 (1994).*

[12] M. Kwiatkowska, G. Norman and D. Parker, "PRISM 4.0: Verification of Probabilistic Real-time Systems," *In Proc. 23rd International Conference on Computer Aided Verification (CAV'11), Springer, 2011.,* vol. 6806, pp. 585-591.

[13] M. G. Garcia-Hernández, A. Reyes, E. Onaindía, S. Ledesma and G. Canedo, "Abstraction in Markov Decision Processes," *IASK International Conference, E-Activity and Leading Technologies.*

[14] J. Andreas, M. Boldt and B. Carlsson, "A risk analysis of a smart home automation system," *Future Generation Computer Systems. 56.,* 2015.