Eng Int Syst (2022) 3: 195–200

Engineering
Intelligent Systems

Cloud-Based Malware Detection of Smart Meters in Advanced Metering Infrastructure

Zuo Jian[1],*, **Ziwen Cai**[2],*, **Bin Qian**[2] **and Yong Xiao**[2]

[1] *Power Dispatch and Control Center of Guangdong Power Grid Corporation, Guangzhou 510600, China*
[2] *Power Research Institute of China Southern Power Grid, Guangzhou 530600, China*

In an Advanced Metering Infrastructure (AMI), smart meters implement encryption/decryption with Embedded Secure Access Module (ESAM) to secure communication. However, meters can be attacked by malicious adversaries once the ESAM is cracked. Since smart meters have limited communication and computing resources and cannot detect malicious code, a cloud-based cyber security protection approach is proposed to detect malware online. Firstly, the closed and fixed operating environment is utilized to establish and maintain a white list of legal processes in the cloud security server of the metering center. Thereafter, the operating process detection agent is installed in the smart meters, and all operating processes are recorded by it. The hash code of each operating process can be established as its identity and submitted to the cloud security server. The meter containing illegal processes can be identified by comparing it with the white list. The smart meter needs only to calculate and upload the hash code of processes, which is affordable for smart meters with limited computing and communication resources. The proposed approach can help strengthen the cyber security defense of an AMI.

Keywords: advanced metering infrastructure (AMI), cloud security, hash code, white list, smart meters

1. INTRODUCTION

An Advanced Metering Infrastructure (AMI) consists of smart meters, communication networks, and the front end system of the metering servers. AMI enables interactions between the power grid and power users to optimize resource allocation (Rinaldi et al., 2019; Montazerolghaem et al., 2018; Bohn et al., 2013). In order to achieve real-time, two-way, and reliable data transmission, the AMI adopts an open system and shared information mode to transmit sensitive information such as electricity price and electricity consumption, and controls instructions in a heterogeneous communication environment. However, the AMI can be compromised by security risks posed by hackers who seek information (Siryani et al., 2017; Spanò, L. Niccolini et al., 2015).

*Corresponding Author e-mail: 346078890@qq.com

Cyber-attacks on smart meters can breach the user's privacy and power consumption data could be altered, resulting in the potential loss of revenue for power utilities. Moreover, an attack can send erroneous information to many smart meters and produce in a large number of power outages (Sun et al., 2018). In 2009, researchers at the Black Hat Conference in the United States demonstrated that the worm virus could infect the smart meters of 15,000 households in one day and cause widespread power failure. Unlike a dispatch and substation automation system, the protection of the network security of an AMI has distinct characteristics:

- Smart meters are distributed on the users' side. Most meters communicate with a meter collector via power line communication and meter collectors communicate with the head end via General packet radio service (GPRS) wireless communication.

- Smart meters transmit sensitive information related to finance and the users' privacy, which is vulnerable to attacks by stakeholders.
- Generally, smart meters are developed with an embedded system which integrates functions such as metering, display, and communication (Wolf et al., 2018). Smart meters have limited computing and communication resources and it is difficult to implement complex measures to safeguard smart meters against cyber-attacks.
- To enable the integration of new applications, such as high-density measurement storage, two-way billing, etc. (Liu and Jiang, 2016; Mosenia et al., 2017), 32-bit embedded processors, such as ARM, have been utilized for smart meters with embedded operating systems such as μClinux to facilitate multi-process management (Malone et al., 2013; Golde). However, the bugs in the operating system will exacerbate the risk of cyber-attack.

The hardware encryption security module has been used for identity authentication and encrypted communication of smart meters. The built-in encryption algorithm is used for symmetric encryption with a key length of 128 bit, which can be used for communication identity authentication and transmission encryption (Soltan et al., 2018; Haddad et al., 2015). Since the cyber security of smart meters depends on the security of the encryption key, scholars have conducted in-depth research on different types of key negotiation and distribution. According to the literature, a lightweight key agreement protocol with a low resource overhead which takes into account the limited computing and communication resources (Keemink et al., 2008). (Bingjie et al., 2021 and Zheng, 2021) also proposed an improved key distribution management mode for AMI.

It should be noted that once the encryption algorithm is cracked, smart meters face the threat of cyber-attacks. At the 2014 Black Hat Conference in Europe, researchers cracked the encryption of the smart meters of a Spanish power grid company using an AES-128-bit symmetric encryption algorithm. This invades the meter after injection and executes malicious code. It can tamper with the id code and adjust power readings for the purpose of power theft. Moreover, it can attack neighboring meters by cutting out the electricity service. Due to the limited computing and communication resources of smart meters, there is a lack of in-depth research on methods for detecting the intrusion of malicious software. This paper proposes an approach for detecting malware in smart meters based on cloud security. Malware detection is implemented over the cloud to address the problems of detecting malware in smart meters with limited computing and communication resources (Adelmira, 2020; Alberto, 2020; Janzene, 2019).

2. MALWARE DETECTION BASED ON CLOUD SECURITY

Since the first decade of this century, computing, storage, and communication resources gathered by networks have increased substantially. The communication network has evolved from a traditional communication platform to a ubiquitous computing platform. Cloud computing integrates a large amount of data, storage, and computing resources distributed over the network, and creates an integrated and collaborative working environment. Services can be provided on demand online, which enables users to share and utilize resources on the open network conveniently. Since cloud computing can reduce costs through the sharing of large-scale resources, which can meet the transient peak demand, it has developed rapidly in recent years (Jaatun et al., 2013). A cloud-based security service is the application of cloud computing in the field of cyber security defense.

Traditionally, code signatures are widely used to detect malware. In order to keep up with the evolution of malware, a huge code signatures library containing numerous malicious codes should be updated frequently in computers. Thereafter, this enables the suspected files to be examined to determine whether or not they are malicious (Illera et al., 2014; Guezguez et al., 2017). With the ongoing development of the Internet and associated technologies, new viruses continue to emerge. The capacity of the library of malicious codes is expanding rapidly and consumes more and more system resources. The capacity of a typical library of malicious codes is around 100 megabytes, and contains millions of malicious code signatures. The explosive growth in the number of viruses is gradually decreasing the efficiency of traditional anti-virus software which, in turn, greatly reduces system performance (Chen et al., 2014).

In order to help detect malware in mobile intelligent terminals with limited computing, storage, and communication resources, a cloud-based security service is proposed by Namboodiri et al., (2014). Cloud security has evolved from a combination of P2P, grid, cloud computing, and other distributed computing technologies. With the malware detection mechanism for cloud-based security, a computer can maintain a simple library of local malicious codes, that contains the most common malicious codes. The local agent scans the operating program, local files, and website. The most common malware can be detected within the local malicious code library, while any other suspicious malware can be detected with a security vendor server cluster via the Internet. Since the cloud server has tremendous resources, it can be connected in real-time and can analyze and process malicious code, and can capture and analyze more complex malicious code. In this way, the local malicious code library can be rather small or not required at all. The composition of the malware detection system based on cloud security is shown in Figure 1.

According to the cloud-based security malware detection system, an agent should be deployed in the user's computer. When the client agent identifies a suspicious file, it sends the hash code of the suspicious file to the cloud server, which analyzes and judges in three ways: file hash code comparison, file sample heuristic analysis and rule-based behavior monitoring analysis.

- When a suspicious file is found, the hash code of the file is uploaded to the cloud server, and the server compares the hash code with the blacklist. It generates a solution and sends it to the client user once the file is on the blacklist.

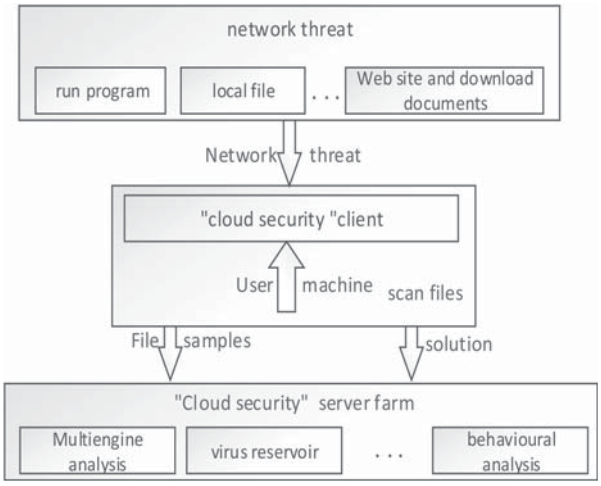

Figure 1 Composition of cloud security system.

- When no abnormalities are found in the hash code analysis of the file, the client agent extracts the file samples, and the cloud server carries out heuristic analysis and detection on the file samples, and generates a solution and pushes it to the client agent if any abnormalities are found.
- If no abnormalities are found in the analysis of the file samples, the client agent collects the behavior features of the file and uploads them to the cloud server. The cloud server determines whether the file is credible with regard to its behavior features and sends a solution to the client.

A cloud-based security malware detection system can significantly reduce the resource demand on intelligent mobile devices. Moreover, it can direct the mass of intelligent devices to a malware monitoring site, monitoring the abnormal behavior of the network in time, intercepting the latest status of various malware and pushed to the server for in-depth analysis. Thereafter, the information is processed and the relevant solutions are distributed to highly-intelligent devices, which speeds up significantly the response to emerging malwares and shortens the life cycle of a malicious code. It can effectively help to cope with serious attack threats.

3. CLOUD-BASED SECURITY MALWARE DETECTION IN AMI

Cloud security is characterized by a black list of illegal programs in a cloud-based security server, so the computing demand of intelligent devices can be reduced greatly. Unlike average computers, smart meters have limited computing resources and cannot conduct heuristic analysis and sample analysis of suspicious malware.

It should be pointed out that blacklist-based cloud security protection is a passive security defense mechanism. It must detect, identify and analyze malware features once the malware has initiated an attack and caused the damage. Thereafter, cloud security-based blacklist comparison and malware detection can limit the scope of malware and reduce associated attacks and damage.

Unlike the blacklist, the whitelist is an active approach to cyber security defense. It adopts a mechanism of authorization and allows only the authorized legal processes to operate. All the unauthorized processes are treated as illegal. Because the operation environment of smart meters is closed and fixed, the same batch of meters has similar operating processes. The number of operating processes in each batch of smart meters together with the name of each process, its size, hash code, and other relevant information can be recorded in the cloud security server database to create a white list. When the legitimate process information in the security server database changes, it can maintain and update the white list.

At present, smart meters upload the metering data and tariff data to the database of the metering center. A cloud security server can be installed in the metering center to detect malware in smart meters as described in the previous section (He and Yan, 2016).

- According to the cloud security, the procedure to detect malware that requires computing resources is transferred to the cloud security server via a communication network. Thus, malware in smart meters can be detected despite limited computing resources.
- Most smart meters operate with a μClinux operating system, which is a closed fixed environment. The number of operating processes is limited, including the embedded operating system and the business process developed by the manufacturer. Therefore, it is much less difficult to create a whitelist of legal processes than a blacklist of illegal processes for smart meters.
- The smart meter client can record the operating processes, calculate 128-bit hash codes of each process, and then upload the hash codes of all processes to the cloud security server in the metering center. The hash function can guarantee the uniqueness identity of the process. Hence, a smart meter with limited computing resources can meet the computing performance requirements.
- Despite the continuous upgrading of malicious code, as long as the hash code calculated in the smart meter is uploaded to the cloud security server, an illegal process

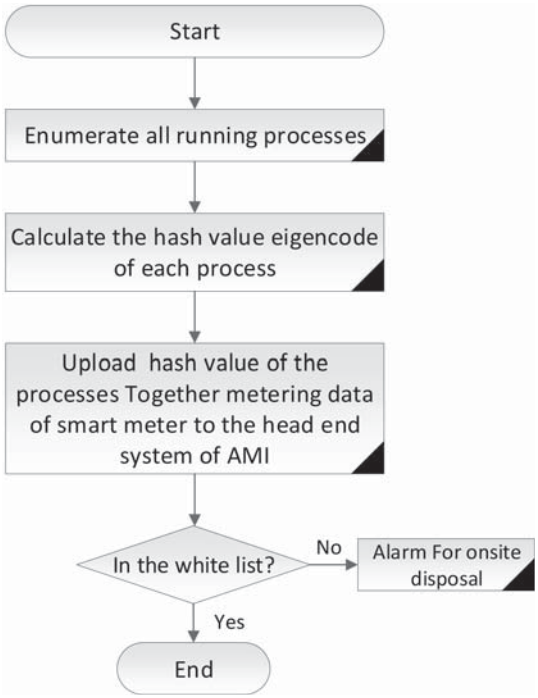

Figure 2 Flowchart of malware detection of smart meters with Cloud security.

that is not on the white list can be detected. The upgrading of malicious code within the life cycle of a smart meter does not affect the detection of malicious code.

It should be pointed out that due to the limited computing resources of smart meters, when the cloud security center detects malicious code in a meter, the meter cannot deal with the malicious code by itself. A field staff should handle it on site.

Building a complete and accurate white list of legitimate program information is essential to the cloud-based cyber security protection of smart meters. As a closed fixed environment, AMI is notably different from the average computer system. When constructing a whitelist, one can make full use of this to detect and record the name, size, hash code, and other information about all the processes of a smart meter. After the initial whitelist is established, the smart meter malware can be monitored online and the whitelist can be updated dynamically as shown in the flowchart depicted in Figure 2.

Cloud security-based malware detection in AMI includes the following steps:

- Smart meters regularly record all operating processes, calculate the hash code of each process and submit it to the cloud security server in the head end.
- The malware detection system of the cloud security server compares the hash codes of each process with the whitelist one by one.
- If the hash code of a process is not in the whitelist, the cloud security server requires the corresponding smart meter to report to the corresponding process module for functional and security detection. If it passes the security test, it will be put on the white list. Otherwise, an alert is issued and on-site staff are advised of the forthcoming investigation and consequences.

When detecting malware in a AMI based on cloud security, there are several problems to be noted:

- In order to reduce the communication pressure on the AMI system, it is necessary to reduce the frequency and data quantity of smart meter to the cloud. One feasible method is to periodically compare the history kept locally in the meter with the history kept in the cloud, and syncronize the 2 by uploading the difference to the cloud. This could be done instead of the complete history having to be repeatedly uploaded. In this way, the traffic pressure on the AMI can be reduced significantly.
- Due to limited changes in the closed environment, if the smart meter cannot upload the processes that are not on the whitelist, this can be done by a human on site, who can perform functional and security detection and update the whitelist in the cloud.

4. EXPERIMENTS

Smart meters are mainly operated with μClinux, μC/OS and other embedded operating systems. The following example, combined with μClinux, illustrates the concrete implementation of the recording process of the smart meter and the determination of the hash code.

Derived from the Linux 2.0/2.4 kernel, μClinux has most of the features of Linux and can use almost all Linux API functions. μCLinux is tailored and optimized for the embedded micro-control field, forming a highly optimized

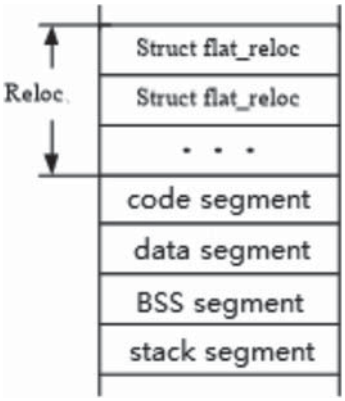

Figure 3 Format of executable file.

and compact embedded operating system with the advantages of small size, stability, good portability, excellent network functions, and extensive support for various file systems. This system is generally utilized for embedded operating systems with little memory or Flash, and has been successfully applied in routers, set-top boxes, PDAs and other devices (Hussain et al., 2018).

In order to reduce the complexity of the system, as well as the hardware development cost and power consumption, the memory management unit module is omitted in the hardware design of the embedded system of μClinux. Hence, the system cannot automatically allocate memory space for the process when the executable file is started. The system developer needs to manage the required physical memory and its storage. In order to adapt to the unique operating environment without memory management unit, μClinux system adopts the flat executable file format, the structure of which is shown in Figure 3.

In a Linux system, a process generally consists of a code segment and a data segment that contains initialized global variables, a BSS segment that contains uninitialized global variables and static variables, and a stack segment that contains automatic variables and local variables. When a process starts loading memory, the memory management unit carries out the virtual address space. Because the embedded system has no memory management unit for virtual memory management, μClinux adopts the real memory management strategy to directly access the physical memory through the address bus. All the addresses accessed by the process are actual physical addresses in a running space. Due to the lack of a system memory management unit, dynamic libraries cannot be used in μClinux, so the loaded applications are all statically compiled and connected. The connector only gives an offset to the segment base address. This offset is put into a unified reloc segment, and when the program is loaded, the actual physical address is obtained by adding the actual segment base address and offset.

In the μClinux system, the program first scans all processes, and then obtains information about each process one by one, including the memory size and other details. It can find the process of the code corresponding to the memory range according to the reloc offset. After reading this memory series, MD5 and SHA algorithm can be used to calculate the codes of the process. Thereafter, the smart meter uploads the hash codes of all processes to the cloud security server to detect the malware.

5. CONCLUSIONS

A cloud security-based approach is proposed in this paper to strength the cyber security defense of smart meters that have limited computing and communication resources. Since smart meters are in a fixed and closed environment, the whitelist is used to provide active cyber security defense. The smart meters record all operations and calculate the hash codes of all processes. Thereafter, the hash codes are uploaded to the security server in the metering center. If the hash code of a process is not in the whitelist, a malware can be identified as it is not recorded in the security server. Since smart meters with limited computing resources can record and calculate the hash codes of all processes, the proposed approach can strengthen the cyber security of smart meters and detect malware online.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support given by National Natural Scientific Funding of China (51777015), Scientific Research Funding of Hunan Education Department (15A0015).

REFERENCES

1. Adelmira A. (2020). Research on Multi-Agent Distributed Application Systems Based on WWW Platform. *Acta Electronica Malaysia, 4*(2): 31–34.
2. Alberto M. (2020). Elements of Music Based on Artificial Intelligence. *Acta Informatica Malaysia, 4*(2): 30–32.
3. Bohn, S., M. Agsten, O. Waldhorst, A. Mitschele-Thiel, D. Westermann, and P. Bretschneider, An ICT architecture for managed charging of electric vehicles in smart grid environments, *Journal of Engineering,* 2013, Article ID 989421, 11, 2013.
4. Chen, C.M., Y.H. Chen, Y.H. Lin, H.M. Sun. Eliminating rouge femtocells based on distance bounding protocol and geographic information. *Expert Systems with Applications, 41*(2), 426–433, 2014.

5. Golde, N., K. Redon, R. Borgaonkar. Weaponizing Femtocells: The effect of rogue devices on mobile telecommunication. https://www.tu-berlin.de/fileadmin/fg214/Papers/femto_ndss12.pdf
6. Guezguez, M. J., S. Rekhis, N. Boudriga. Observation-based detection of femtocell attacks in wireless mobile networks, in *Proc. Symp. Appl. Comput*., 2017, 529–534.
7. Haddad, Z., M. Mahmoud, S. Taha, I. A. Saroit, Secure and privacy-preserving AMI-utility communications via LTE-A networks, *2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications* (WiMob), Abu Dhabi, 2015, 748–755.
8. Halifax Regional Water Commission. *AMI Technology Assessment & Feasibility Study Consolidated Report, 2014*. Available online: https://www.halifax.ca/sites/default/files/documents/home-property/ water/AMI Technology-FianlReport.pdf
9. He, H., J. Yan, Cyber-physical attacks and defences in the smart grid: a survey, *IET Cyber-Physical Systems: Theory & Applications, 1*(1), 13–27, 12 2016.
10. Hussain, S. R., O. Chowdhury, S. Mehnaz, et al. LTE Inspector: A Systematic Approach for Adversarial Testing of 4G LTE. *Network and Distributed Systems Security Symposium 2018* 18–21 February.
11. Illera, A. G., J. V. Vidal. Lights off! The darkness of the smart meters. *2014 Europe Black Hat Conference*. Available online: https://www.blackhat.com/eu-14/archives.html#lights-off-the-darkness-of-the-smart-meters
12. Jaatun, M., I. Tøndel, G. Køien. GPRS Security for Smart Meters. Alfredo Cuzzocrea; Christian Kittl; Dimitris E. Simos; Edgar Weippl; Lida Xu. *1st Cross-Domain Conference and Workshop on Availability, Reliability, and Security in Information Systems* (CD-ARES), Sep 2013, Regensburg, Germany. Springer, Lecture Notes in Computer Science, LNCS-8127, 195–207, 2013
13. Janzene L. A. (2019). Sustaining The Supply Chain Management System of A Multi-Purpose Cooperative in Tiaong, Quezon. *Information Management and Computer Science, 2*(1): 04–09.
14. Keemink, S., B. Roos. *Security analysis of Dutch smart metering systems*, University van Amsterdam, Amsterdam, Netherlands, 2008 July.
15. Liu, G. Y. and D. J. Jiang, 5G: Vision and requirements for mobile communication system towards Year 2020, *Chinese Journal of Engineering*, 2016,
16. Malone, D., D. F. Kavanagh and N. R. Murphy, Rogue femtocell owners: How Mallory can monitor my devices, *2013 Proceedings IEEE INFOCOM*, Turin, 2013, 3387–3392.
17. Montazerolghaem, A., M. H. Yaghmaee, A. Leon-Garcia, Open AMI: Software-defined AMI load balancing, *IEEE Internet of Things Journal*, 5(1), 206–218, Feb. 2018.
18. Mosenia, A., N. K. Jha, A comprehensive study of security of Internet-of-Things, *IEEE Transactions on Emerging Topics in Computing, 5*(4), 586–602, 1 Oct.–Dec. 2017.
19. Namboodiri, V., V. Aravinthan, S. N. Mohapatra, B. Karimi, W. Jewell, Toward a secure wireless-based home area network for metering in smart grids, *IEEE Systems Journal, 8*(2), 509–520, June 2014.
20. Rinaldi, S., P. Ferrari, A. Flammini, E. Sisinni, A. Vezzoli, Uncertainty analysis in time distribution mechanisms for OMS smart meters: The last-mile time synchronization issue, *IEEE Transactions on Instrumentation and Measurement, 68*(3), 693–703, March 2019.
21. Siryani, J., B. Tanju, T. J. Eveleigh, A machine learning decision-support system improves the Internet of Things' smart meter operations, *IEEE Internet of Things Journal, 4*(4), 1056–1066, Aug. 2017.
22. Soltan, S., P. Mittal, H. V. Poor. BlackIoT: IoT Botnet of high wattage devices can disrupt the power grid. *Proc. of the 27th USENIX Security Symposium.* Aug 15–17, 2018, Baltimore, USA
23. Spanò, E., L. Niccolini, S. D. Pascoli, G. Iannacconeluca, Last-meter smart grid embedded in an internet-of-things platform, *IEEE Transactions on Smart Grid, 6*(1), 468–476, Jan. 2015.
24. Sun, Y., L. Lampe, V. W. S. Wong, Smart meter privacy: exploiting the potential of household energy storage units, *IEEE Internet of Things Journal, 5*(1), 69–78, Feb. 2018.
25. Wolf, M., D. Serpanos, Safety and security in cyber-physical systems and internet-of-things systems, *Proceedings of the IEEE, 106*(1), 9–20, Jan. 2018.
26. Xenakis, C. Malicious actions against the GPRS technology, *Journal Computer Virology, 2*(1), 121–133, Aug. 2006
27. X. Zheng. Massive High-Dimensional Big Data Feature Selection Algorithm in a Cloud Computing Environment. *International Journal of Engineering Intelligent Systems*, 29(5), 323–330, May 2021.
28. Y. Bingjie, Y. Huifeng, S. Chenjun, Z. Zhi, F. Jinghang. Research on Network Security Collaborative Defense Technology Based on Swarm Intelligence and Big Data Network Security. *International Journal of Engineering Intelligent Systems*, 29(6), 379–386, June 2021.
29. Zhu, D. L., N. Pang and Z. M. Fan. A self-testing approach defending against rogue base station hijacking of intelligent terminal. *2015 International Conference on Applied Science and Engineering Innovation*. 929–937.