ISSN: 2664-9640

An Use Case Description Method Oriented to Industrial Control Protocol

Rui Zhang[1, a, *] and Guanyu Zhang[1]

[1]Shenyang Jianzhu University, College of Information and Control Engineering, Shenyang, 110168, China.

[a]945816869@qq.com

Abstract

In the test of the security and robustness of industrial control protocols, how to complete the test with fewer use cases is an important research topic for test workers, and the way in which test cases are described is the first important issue that affects the quality of use cases. factor. Taking Modbus protocol as an example, this paper proposes a calculation method of use case similarity and population dispersion based on weight division, which can more accurately describe the similarity of individual use cases and the degree of individual dispersion within the population. Experiments show that When the same use case generation algorithm is used, the method proposed in this paper can increase the population dispersion by 3.45% compared with the conventional method, and effectively increase the use case redundancy.

Keywords

Test case, protocol test, similarity, industrial control security.

1. Introduction

The industrial control system controls the data collection, image and sound signal processing, information transmission, and process control in the entire production process. The safety and reliability during operation are related to the stability of the entire system. In recent years, with the rapid popularization and application of computer networks, traditional industrial control systems have gradually developed in the direction of interconnection and intelligence, which has led to emerging concepts such as the Internet of Things, Industrial Internet of Things, and Industry 4.0. While injecting new vitality into industrial control systems, it also brings the same challenges [1].

In the security system of the industrial control system, the transmission protocol is an important guarantee to ensure the safe transmission of information. The low cost of attacks against the protocol is one of the most common attack methods. Especially the rapid development of the network makes remote attacks possible. [2]. Based on the data message structure characteristics of the test protocol, this paper proposes a method for calculating the similarity of use cases and population dispersion based on weight division, which can achieve a more accu

rate description of the similarity of test cases and effectively improve the test case in the test process. Generate and optimize effects to increase test coverage.

2. Modbus Test Case Design

2.1. Message Feature Analysis and Coding

In protocol testing, selecting a suitable use case encoding method can reduce the time complexity of generating use cases during the test process and complete the conversion of the

ISSN: 2664-9640

encoded file to the data message faster. Figure 1 is the data segment contained in the data packets of the more common Modbus communication protocol in the industrial control protocol and the byte length occupied by each field [3].

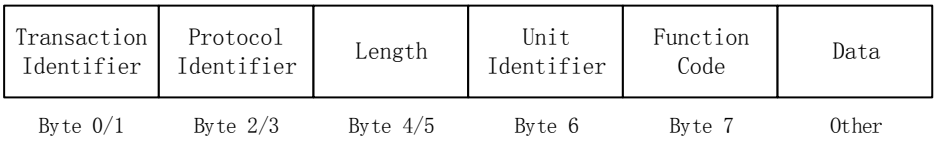

Fig 1. Modbus data message structure

In the Modbus protocol message, since the transmission identifier and the protocol identifier are not related to the message content, these two fields can be ignored when constructing the test case, so each test case can be described as Equation 1

$$case = [l \quad u \quad f \quad d] \tag{1}$$

Among them, l is the length of the data segment, and its value matches the length of the data contained in the following three fields; u is the address identifier, and the legal value is 0 to 255; f is the function code, which is divided into common function codes and User-defined function code, ranging from 1 to 127; d is the data segment. The data information of this field depends on the function code.

2.2. Method for Calculating Similarity of Test Cases with Weights

In Modbus, the length of each field of the message sequence is basically fixed, and the functions of each field and the impact on the security of the message are not the same. Some fields are related to each other. If you use the encoding characters of the two use cases directly The Hamming distance between strings is used as a similarity judgment, and there is some irrationality. In order to solve the above problems, this paper proposes a weight distance calculation method based on internal classification. The weights of different fields are set, and the distances of different fields are calculated according to the corresponding functions. The weight coefficients of each field are determined by analytic hierarchy process.

Assuming there are test cases A and B, first calculate the corresponding distances of each functional field of the two, then combine the weights of the fields, and then calculate the overall similarity between the two. The final calculation formula is shown in Equation 2.

$$dis_{AB} = \sum_{i=1}^{4} w_i \cdot d(A_{vi}, B_{vi}) \tag{2}$$

Where $w = [w_1, w_2, w_3, w_4]$ is the weight of each field; A_{vi} and B_{vi} are the corresponding fields of the two use cases, and $d(A_{vi}, B_{vi})$ is the distance between the two corresponding fields. The distance calculation method for different fields is slightly different.

The pairwise comparison matrix determined by the analytic hierarchy process is shown in Equation 3.

$$A = \begin{bmatrix} 1 & 3 & 1/3 & 1/2 \\ 1/3 & 1 & 1/5 & 1/3 \\ 3 & 5 & 1 & 3 \\ 2 & 3 & 1/3 & 1 \end{bmatrix} \tag{3}$$

ISSN: 2664-9640

The consistency check is performed on the pairwise comparison matrix, and the coefficient $CR = 0.0386 < 0.9$ is checked, and the consistency check passes. The calculated weight of each field is shown in Equation 4.

$$w = [0.1682 \quad 0.0769 \quad 0.5167 \quad 0.2382] \tag{4}$$

Formula 2 is the formula for calculating the distance between fields. Here, the Hamming distance is used for calculation [4].

3. Use Case Population Dispersion Calculation Method

In the process of generating test cases, iteration is performed by using population as a unit, so a generation of population needs to be described from the perspective of the entire population. Here the population dispersion sca is designed to describe the population state. Population dispersion refers to the overall degree of dispersion among individuals in a generation of population. When the dispersion is low, it means that the overall similarity of individuals within the population is too high, and the coverage of test cases is low [5]. At this time, the parameter information in the test case generation process, such as mutation probability and criticality threshold, can be appropriately changed to adjust the distribution of the generated test cases and improve the coverage of the test cases.

When describing the dispersion of individuals in the entire population, it can usually be described by the average distance between individuals. This method is feasible to a certain extent, but each individual needs to calculate the distance between itself and all other individuals, leading to this method There are a lot of repeated calculations, and the efficiency is low. In addition, if there is an extremely uniform edge distribution, it will lead to misjudgment. Therefore, the concept of population dispersion is proposed in this paper, and a new calculation method is designed to accurately reflect the distribution of individuals in the population and reduce the amount of calculation.。

First normalize the values of the individual fields in the population.

$$v_n = \frac{c_n - c_{n\_\min}}{c_{n\_\max} - c_{n\_\min}} \tag{5}$$

Where $c_{n\_\min}$ and $c_{n\_\max}$ represent the minimum and maximum values of this field in the population, respectively.

Calculate the average center use case $\overline{case} = \left[\overline{l_v} \quad \overline{u_v} \quad \overline{f_v} \quad \overline{d_v}\right]$ by summing and averaging each field, where the calculation formula of each field is as formula 6.

$$\bar{v} = \frac{1}{m} \cdot \sum_{i=1}^{m} v_i \tag{6}$$

At this time, the variance can be used to describe the degree of dispersion of individuals in the population, as shown in Equation 7.:

$$sca = \sum_{i=1}^{m} \left(\left(l_{iv} - \overline{l_v}\right)^2 + \left(u_{iv} - \overline{u_v}\right)^2 + \left(f_{iv} - \overline{f_v}\right)^2 + \left(d_{iv} - \overline{d_v}\right)^2 \right) \Big/ m \tag{7}$$

ISSN: 2664-9640

4. Experimental Verification

By designing test case description methods and similarity calculation methods among test cases, combined with the description of the dispersion within the test case population, theoretically, the efficiency of test case generation can be effectively improved, and the coverage of test cases can be improved. In order to verify the correctness of the proposed method, a set of comparative experiments is designed, and the genetic algorithm is used as the core generation algorithm of the test case. In the description of the use case description method, individual similarity, and population dispersion, the methods described in this article and conventional methods are used to describe the test cases generated by the two.

Genetic algorithm is an intelligent optimization algorithm, which is often used to find the global optimal solution, and adjust the optimization direction of the population by designing the corresponding fitness function [6]. In the genetic algorithm used in the test case generation method designed in this paper, the direction of population convergence is suspicious use cases existing in historical data. The significance of fitness function is that when there are suspicious points in the population, the population converges to the suspicious use case, and when it does not exist, it tends to have a higher dispersion of the population. Other parameter settings of genetic algorithm: mutation probability $P_m = 0.2$, cross probability $P_c = 0.6$.

The script development language of the experiment is Python 3.6; the test uses Modbus communication simulation software Modbus Poll and Modbus Slave. First use Modbus Poll to establish data communication with Modbus Slave, use Wireshark packet capture tool to obtain normal communication messages, select representative data messages from them to analyze the data characteristics, and construct the initial population; then send the initial population to the use case generation And optimization module to iterate, optimize and update use cases; finally send each generation of population to the target for testing. Statistical analysis was performed on the use case data generated by the two methods, and the dispersion of the current population was collected every 200 generations during the experiment. The results are shown in Figure 2.

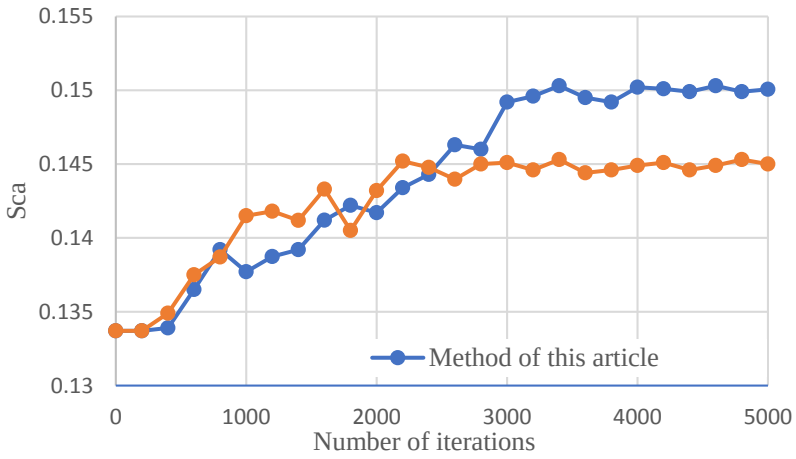

Fig 2. Trend of population dispersion

In two groups of experiments using different methods, during the population iteration process, the dispersion gradually increased and eventually stabilized. At the beginning of the experiment, since the same initial population was used, the dispersion of the two was the same. However, with the population iteration, when both are stable, the dispersion of the population produced by the improved method is increased by 3.45% compared with the conventional method. It is generally believed that the higher the dispersion between individuals within a population, the higher the coverage of test cases [7]. Therefore, it can be considered that the

ISSN: 2664-9640

coverage of test cases generated by the improved method is higher than that of the conventional method, and it also proves that the method proposed in this article has certain advantages over the conventional method.

5. Conclusion

This paper proposes a new use case similarity determination method and the concept of population dispersion, which provides a new idea and method for improving the use case coverage in the protocol testing process. In the determination of test case similarity, different weights and distance calculation methods are set according to different protocol fields, which can more accurately determine the similarity according to the function and data content of the use case, and the change of the encoding method is effectively resolved. The problem of inaccurate similarity determination caused by data mutation. In addition, the genetic algorithm is introduced into the use case generation algorithm, and the use case similarity and population dispersion are used as the fitness function of the genetic algorithm to realize the automatic optimization of use case generation. The use case data generated in the experiment shows the effectiveness of the method Sex. The next step is to improve the applicability of this method and apply it to the generation of test cases for other protocols.

References

[1] Yi Shengwei, Zhang Yibin, Xie Feng, et al. Security analysis of industrial control network protocols based on Peach [J]. Journal of Tsinghua University: Natural Science Edition, 2017, 57 (1): 50-54.

[2] Zhang Yafeng, HONG Zheng, WU Lifa, et al. Form-syntax based Fuzzing method for industrial control protocols [J]. Application Research of Computers, 2016, 33 (08):2433-2439.

[3] Cheng Yang, Liu Xueping, Zhan Tao. Design of an industrial control system based on MODBUS protocol [J]. Machinery Design and Manufacturing, 2011 (01): 1-3.

[4] Katsigiannis, Konstantinos & Serpanos, Dimitrios. MTF-Storm: a high performance fuzzer for Modbus/TCP[C]. 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA). 2018. 926-931.10.1109/ETFA. 2018. 8502600.

[5] Kang J, Park J H . A secure-coding and vulnerability check system based on smart-fuzzing and exploit [J]. Neurocomputing, 2017, 256(20):23-34.

[6] LI Zhu, Automatic Testing-Case Generation Based on Adaptive Genetic Algorithm [J]. Application of Computer System, 2016, 25 (01):192-19.

[7] VOYIATZIS A G, KATSIGIANNIS K, KOUBIAS S, et al. A Modbus/TCP Fuzzer for test internetworked industrial systems[C]. 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA). IEEE, 2015.