Modbus/TCP Communication Anomaly Detection  
Based on PSO-SVM  
Wenli Shang¹, a, Shengshan Zhang²,b and Ming Wan³,c

¹ Shenyang Institute of Automation Chinese Academy of Science, Shenyang 110016, China  
² Shenyang Institute of Automation Chinese Academy of Science, Shenyang 110016, China  
³ Shenyang Institute of Automation Chinese Academy of Science, Shenyang 110016, China  

a shangwl@sia.cn, b zhangshengshan@sia.cn, c wanming@sia.cn

Keywords: PSO, SVM, Modbus function code, Sequence anomaly detection, Industrial security gateway

Abstract. Industrial firewall and intrusion detection system based on Modbus TCP protocol analysis and “whitelist” policy cannot effectively identify attacks on Modbus controller which exactly take advantage of the configured rules. An Industrial control systems simulation environment is established and a data preprocessing method for Modbus TCP traffic captured is designed to meet the need of anomaly detection module. Furthermore a Modbus function code sequence anomaly detection model based on SVM optimized by PSO method is designed. And the model can effectively identify abnormal Modbus TCP traffic, according to frequency of different short mode sequences in a Modbus code sequence.

Introduction

Industrial control system originally adopted proprietary communication protocols, operating systems, hardware devices and so on. What’s more, it was strictly isolated from other networks, and much more attention was put on the physical security and functional safety. However, the industrial control system designers neglected the information security. With the huge demand of information technology, industrial control system is no longer as enclosed as before. Because of the widespread use of TCP/IP, open industrial communication protocol, and general operating system, industrial control system born with so many information security and network security flaws is becoming more and more fragile [1-4].

Modbus TCP protocol is utilized diffusely in the industrial control system or SCADA system of petrochemicals, energy, metallurgy, and electric power, many researchers and institutions have done some instructive work about vulnerability analysis, attack and protection of Modbus protocol, and several representative achievements are as bellow:

Sun Da-Lin etc. analyzed the security of Modbus TCP in the SCADA system, and proposed a lot technology and management solutions [5], which can effectively reduce the information security risk in industrial control system, but those methods mainly develop from traditional IT network security policy and does not involve Modbus application protocol information security issues.

Wang Ting-Ting designed an improved DES method that encrypts Modbus RTU communication [6], however, encryption/decryption need too much resource and may have a negative impact on the real-time performance. Zhang Yun-Gui etc. proposed a non-parametric CUSUM intrusion detection method based on industrial control model [7]. This method firstly computes the difference between industrial control model predicted output and signal from sensors, then form the time-based statistical series, and at last with the CUSUM method realize an online intrusion monitor and alarm. Nevertheless, this way should be quite familiar with procedures and parameters of the production process, and there may be large error in the control model established.

Javier Jiménez etc. proposed a method which according to signatures detects intrusion behaviors targeting the industrial control system based on Modbus TCP communications [8]. Obviously, signatures cannot describe all the kinds of attacks behaviors on the Modbus TCP communications,
and false negatives cannot be avoided. Venkat Pothamsetty etc. designed a Modbus application protocol packet filter firewall based on Netfilter/Iptables framework in the kernel of Linux [9]. But this sort of firewall cannot block all the malicious traffic without effecting regular communication requirements, because overly strict configuration may block the innocent traffic too. In practice, there has been industrial firewall for industrial control system and SCADA system, which analyze the packet according to Modbus application protocol based on deep packet inspection technology [10]. In summary, the above methods form relatively complete architecture from information policy, transmission encryption, intrusion detection, access control of application protocol and several other aspects. However, the intrusion and attack behavior with unknown feature, or attack behaviors in the “whitelist” of firewall cannot be identified. To solve this problem, Modbus function code is selected as the object of study, a Modbus TCP communication function code sequence anomaly detection method is proposed, and whose parameters are obtained with particle swarm optimization (PSO). This PSO-SVM module can identify aggressive behavior and unusual behaviors which cannot be detected by firewall or intrusion detect system.

Feature Selection of Modbus TCP Communication

Modbus is an application layer messaging protocol, meaning that it operates at layer 7 of the OSI model. It is the oldest and perhaps the most widely deployed industrial control communications protocol. It has been the de facto standard and enhanced over the years into several distinct variants, such as implementation over serial link (Modbus RTU/Modbus ASCII), implementation over TCP/IP (Modbus TCP), and implementation over high-speed token ring (Modbus Plus). Modbus TCP is focused, and its message format is shown in Fig. 1 [11-12].

As shown in the figure, MBAP HEADER stands for Modbus application protocol header, which contains four fields including transaction identifier, protocol identifier, length and unit identifier [11]. FUNCTION CODE stands for Modbus function code, which indicates what kind of operation a Modbus client wants a Modbus server to execute; therefore this field can describe the operation intent best. The DATA part is filled by the client according to a specific application and answered by the server.

Since Modbus TCP protocol is on the basis of TCP/IP [12], the underlying protocol security flaws are inherited. What is more, Modbus application protocol itself lack authentication, authorization, and encryption and other security mechanisms. Lacking of authentication means a Modbus session can be established easily if a Modbus address is supplied. Lacking of authorization means no access control mechanisms based on roles exist, and any user can execute arbitrary function. Lacking of encryption means address and command can be easily captured and parsed [13,14].

For example, a Modbus client compromised by some kind of industrial computer virus may sniff and analyze traffic in the network. Then malicious packets composed by Modbus function code and coil or registers are fabricated. This kind of attack cannot be blocked by a firewall even the firewall work at the application protocol level, because the attacker take advantage of what is need in the regular communication.
Industrial control system and SCADA system usually is programeed and configured in advance, and the communication behaviors have periodic characteristics. It is assumed that, a control system or SCADA system based on Modbus TCP communication has a certain sequence characteristics and behavior patterns. And if a system is invaded, the sequence characteristics and stable behavior patterns can no longer be maintained. Therefore, regular Modbus TCP communication model can be established to identify abnormal and aggressive behaviors.

Since Modbus function code field can effectively characterize the operation intention, Modbus function code is treated as object of study. Modbus TCP communication can be transformed into Modbus function code sequence, and thus the Modbus TCP communication anomaly detection is converted to anomaly detection of Modbus function code sequence.

```
\[
A_i = a_i^1 a_i^2 a_i^3 a_i^4 a_i^5 \\
A_s = a_i^6 a_i^7 a_i^8 a_i^9 a_i^{10} \\
A_n = a_i^{11} a_i^{12} a_i^{13} a_i^{14} ... a_i^{18} \\
\]
```

![Diagram](image)

Figure 2. Data preprocessing process

**Data Preprocessing**

Before Modbus TCP traffic gained from industrial control system is used for anomaly detection, it needs some preprocessing procedure. Firstly, with the chronological order of the packets unchanged, randomly the packets are divided into different sequences. And then different sequences are given two kinds of labels, one is the +1 class and another is -1 class. All the sequences which contain packets from simulated attack source are sentenced as -1 class. At last, all unnecessary packets that don’t contain Modbus function code are eliminated and Modbus function code sequence sample collection is obtained.

Because samples which those artificial intelligence algorithms like SVM should have the same dimensions, this kind of Modbus function code sequence cannot be dealt with directly, for a sequence may contain different number of Modbus function codes. Obviously, the way in which every single
Modbus function code is mapped to one dimension of the vector is not feasible. If every Modbus function code sequence has the same number of Modbus function code, the method may be a good one. However, it could cause the loss of randomness and flexibility during the sample selection process.

A method that transfer Modbus function code sequences containing different number of function codes into vectors of same dimensions is illustrated as Fig. 2 shows:

Step1: assign r as length of short sequence. For Modbus function code set \( \{A_n\} \), \( n = 1, \ldots, N \), \( N \) stands for number of samples, \( A_i = a_1^i a_2^i \ldots a_n^i \), \( a_j^i \in \{\text{Modbus function code}\} \), \( j = 1, \ldots, n_i \), \( n_i \) is the number of Modbus function code in the ith sample, and \( r \leq n_i, \ \forall i = 1, \ldots, N \).

Step2: Obtain the mode short sequence set \( M \). Handle every \( A_i \) in the Modbus function code sequence set \( \{A_n\} \) with a window whose length equals configured in the last step, and then for every single sequence \( A_i \), at most \( \sum_{i=1}^{N} n_i - rN + N \) Modbus function code sequences that contain \( r \) Modbus function codes can be got. After removing duplicate sequences, a mode short sequence set \( M = \{M_1, M_2, \ldots, M_m\} \), \( m \leq \sum_{i=1}^{N} n_i - rN + N \) can be obtained.

Step3: Express Modbus function code sequence in a vector. For sequence \( A_i = a_1^i a_2^i \ldots a_n^i \), a SVM vector \( x_i = (x_{i1}, x_{i2}, \ldots, x_{im}) \) is constructed, and \( x_{ij} \), the jth component of \( x_i \), stands for the frequency of the jth component of mode short sequence set \( M \), and the formula is \( x_{ij} = g(m_j)/(n_i - r + 1) \), where \( g(m_j) \) stands for number of times the jth component \( m_j \) appears in \( A_i \).

Obviously, the method mentioned above used for data processing have a lot advantages. Firstly, packets sequences of different lengths are mapped into vector with the same dimension, and the sample selection has great flexibility. Secondly, every component of a SVM vector describes the frequency characteristics of the corresponding mode short sequence; that is to say, the vector describes the sample more rationally.

**PSO-SVM Anomaly Detection Model**

Method like mathematical statistics, feature selection, neural networks, machine learning, data mining and so on almost need a large number of training samples, otherwise will cause larger false positives and false negatives.

Unlike methods mentioned above, SVM method has several unique advantages: first, SVM is good at obtaining the optimal solution in the case of limited samples, and the data captured in the experimental environment happens to belong to the limited sample situation; secondly, SVM algorithm can eventually be converted to convex quadratic programming problem, and falling into local extreme can be avoided; Thirdly, nonlinear separable problems could be converted into linearly separable problem by transformation from low-dimensional space into a high dimensional space; lastly, dimension disaster is avoided by introducing kernel function and complexity of the algorithm has nothing to do with the dimension of space [15].

In practical applications, the performance of SVM largely depends on the selection of parameter and kernel function. In most cases, the designer selects parameters and kernel functions based on his experience. Particle swarm optimization (PSO) algorithm is widely used in parameter optimization, and can effectively improve the efficiency of searching process [16].

So a PSO-SVM module is proposed for the anomaly detection of Modbus function code sequence, and the whole process is illustrated in Fig. 3. The parameters of SVM module are optimized by PSO method, and can enhance the classification accuracy.

A set of particles are initialized in the feasible solution space in PSO method, and every particle stands for a potential optimal solution. Position, speed and fitness value are three important features of a particle, and the fitness value should be calculated based on the actual situation, which describes the particle is good or not. Particles move in the solution space, individual extreme, group extreme, position, speed and other parameters are updated every loop. Individual extreme records the position which has the best fitness value, and group extreme records the best position which belongs to all the particles. Formula 1 and 2 explains how speed and position are updated [17,18].
In formulas above $k$ and $k+1$ stands for last loop and this loop, $V$ is the velocity, $P$ represents individual extreme, $G$ represents group extreme. Inertia factor $c_1$ and $c_2$ are nonnegative constants; $r_1$ and $r_2$ are the acceleration factor generated randomly between 0 and 1. Penalty factor $C$ and parameter $\sigma$ of radial basis function are optimized by the PSO process as illustrated in the next several steps.

\begin{align*}
V^{k+1} &= \omega V^k + c_1 r_1 (P^k - X^k) + c_2 r_2 (G^k - X^k), \quad (1) \\
X^{k+1} &= X^k + V^{k+1}. \quad (2)
\end{align*}

Step0: Set $k_{\text{max}}$ as the maximum number of iteration.

Step1: Generate particles and initialize associated parameters. $X = (X_1, X_2, ..., X_N)$ is generated randomly, and $N$ stands for the number of particles. Every particle is made up of two components, in other words, $X_i = (x_{ic}, x_{i\sigma})$, where $x_{ic}$ represents penalty factor $C$ and $x_{i\sigma}$ represent $\sigma$, the parameter of radial basis function. Correspondingly, velocity $V_i$ is made up of two components $V_{ic}$ and $V_{i\sigma}$. What’s more, components of $X_i$ should be constrained to $[X_{c\text{min}}, X_{c\text{max}}]$ and $[X_{\text{min}, \sigma}, X_{\text{max}, \sigma}]$.

Step2: Calculate fitness value $F(X_i)$ of particles. Fitness value $F(X_i)$ is classification accuracy in the sense of cross-validation under the condition where components $x_{ic}$ and $x_{i\sigma}$ respectively represent penalty factor $C$ and parameter $\sigma$.

Step3: Update individual extremes and group extreme based on fitness values. If $F(X_i^k) > F(X_j^{k-1})$, then $P^k = X_j^k$, else $P^k = X_j^{k-1}$. If $j$ exists and $F(X_j^k) > F(X_i^k)$, and $F(X_j^k) > F(G^{k-1})$, then $G_j^k = X_j^{k-1}$, else $G_j^k = G_j^{k-1}$.

Step4: Determine whether the conditions are met to exit iteration. If the number of iterations is more than $k_{\text{max}}$, or during 50 consecutive iterations group fitness value changes no more than 0.01%,
exits the iterative process and G stands for the best classification accuracy, and the corresponding particle $X_g$ records optimal parameters for the SVM module.

Step5: Update speeds and positions of particles based on formulas mentioned above. In the end of every iteration, check whether component of particles are constrained into $[X_{cmin}, X_{cmax}]$ and $[X_{armin}, X_{armax}]$, and those violate the limitations should be adjusted. For example, if $x_{ic} < x_{cmin}$, then $x_{ic} = x_{cmin}$, if $x_{ic} > x_{cmax}$, then $x_{ic} = x_{cmax}$.

In step 2, classification accuracy in the sense of cross-validation of the SVM module is selected as fitness value. The anomaly detection module based on SVM is established as described as follows:

Step1: Accept penalty factor C and parameter given by PSO parameter optimization process.

Step2: Give all samples category labels. Normal Modbus function code sequences belong to +1 class, and abnormal function code sequences belong to -1 class.

Step3: Solve the dual problem to get the SVM model [19]:

$$
\min Q(\alpha) = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^{n} \alpha_i. 
$$

s.t. $\sum_{i=1}^{n} \alpha_i y_i = 0, \ 0 \leq \alpha_i \leq C, \ i = 1, ..., n.$

The solution is $\alpha^* = (\alpha_1^*, \alpha_2^*, ..., \alpha_n^*)$.

Step4: Construct discriminant function [19]:

$$
b^* = y_j - \sum_{i=1}^{n} \alpha_i^* y_i K(x_i, x_j), \ j \in \{j|0 < \alpha_j^* < C\}. 
$$

$$
f(x) = \text{sgn}(\sum_{i=1}^{n} \alpha_i^* y_i^* K(x_i, x) + b^*).
$$

Step5: Calculate the classification accuracy of SVM according to discriminant function, and return the accuracy to PSO parameter optimization process as the value of fitness value calculation function $F(X)$.

Comparison and Analysis of Simulation Results

Experimental Data Acquisition. Currently, there are no data sets of industrial control systems available for testing and evaluation. For feasibility analysis and validation of various algorithms of Modbus function code sequence anomaly detection, a laboratory simulation environment is designed and established, and the topology is shown in Fig. 4.

Figure 4. Industrial control system simulation environment
Schneider M340 PLC, with CPU module 2020, is selected as Master controller in the control unit layer, communicating with monitoring computer in Modbus TCP protocol. SCADA monitoring screen is developed with KingSCADA software, and UnityPro software installed on the engineer station is used for the programming and configuring the PLC. Attack source simulates a node infected with a computer virus sending malicious instructions to important controller. This environment demonstrates how level of a certain container is controlled, by simulating actions of valve, values of level sensors, container model and control logic in the PLC.

When the environment is running, Modbus TCP traffic should be captured. Then packets related to three-way handshaking, confirmation and retransmission in the TCP mechanism should be eliminated. After that traffic between Modbus clients and servers are obtained, because servers respond the clients with the same Modbus function code, those packets from servers to clients can be eliminated also. The rest packets from Modbus clients to Modbus servers are the key factors for Modbus function code sequence anomaly detection.

**Results Analysis.** 114 Modbus function code sequences are obtained in the Industrial control systems simulation environment, the length of mode short $r$ is 3. After processing those short sequences can the samples can be got where there are 64 samples belonging to +1 class and 50 belonging to -1 class.

Inertia factor $C_1$ equals 1.5 and $C_2$ equals 1.7, number of particles is set as 20, and 100 is specified as the maximum number of iteration. The classification accuracy of SVM is calculated in 5-fold cross-validation method. At the same time, models like SVM traversing grid to find the optimal parameters [20], standard RBF and standard BP are established to compare with PSO-SVM model.

As shown in Fig. 7, in the Particle swarm optimization in iterative process, fitness value has a very fast convergence, and that means PSO-SVM is efficient. The classification accuracy reaches impressive 95.83% for test set, and for the train set the accuracy maintains 100%. To conclude, PSO-SVM model combines the advantage of PSO in parameter optimization and generalization capability of SVM generalization capability.

To make experimental data reflects objectively the performance of four models, 30 classification tests are done for each model, and key statistical indicators are shown in Table 1.

| Table I. Results of Four Models |
|---------------------------------|
| PSO-SVM | C-SVM | RBF | BP |
| $C_{best}$ | 1.22627 | 2.75197 | / | / |
| $\sigma_{best}$ | 0.1 | 0.04 | / | / |
| $Acc_{cv}$ | 83.54% | 91.85% | / | / |
| $Acc_{train}$ | 100% | 100% | 100% | 100% |
| $Acc_{test}$ | 84.17% | 79.58% | 50% | 54% |
| **Time** | 23.68[s] | 139.83[s] | 1.8[s] | 4[s] |
Because of limited number of samples, the train set classification accuracy could always maintain 100% for all these four kinds of models, but two kinds of SVM perform much better than RBF and BP in classifying the test set. PSO-SVM model performs a little better than SVM model that traverses grid to find the optimal parameters, but it cost far less time than the latter does. The penalty factor C obtained by PSO-SVM is much smaller, and that means fewer support vectors.

Conclusion
Intrusion and attacks targeting on industrial control system and SCADA system which take advantage of design flaws in Modbus application protocol or rules and whitelist policies configured in firewall and intrusion detection system cannot be detected or blocked. To solve this problem, it is suggested that Modbus function code field should be focused on, and a data processing method that converts Modbus TCP traffic into Modbus function code sequences and furthermore into vectors that can be handled by anomaly detection models is proposed. What is more, an anomaly detection model based on SVM which is optimized by PSO algorithm is put forward and achieved good classification and detection results.

The PSO-SVM model possesses the following advantages: firstly, it can detect unknown attacks and intrusion, and is a powerful supplement of application layer filtering firewall and intrusion detection system; secondly, can the data processing method be popularized where other industrial protocols and important fields need to be handled; thirdly, the model has great practical value in industrial system network auditing, intrusion detection and other fields.

However, real-time performance of the PSO-SVM module is weak. In the future, several fields in other industrial protocols could be handled simultaneously, and much work could be done in the aspects of dimensionality reduction to get data that both describes communication traffic efficiently and keep refined.
Acknowledgment

This work was supported by Hi-Tech Research and Development Program of China (No.2012AA041102-03) and by the National Natural Science Foundation (No.61164012).

References

[1] Q. Z. Wei, “Industrial network control system security and management,” Measurement & Control Technology, vol. 32, no. 2, pp. 87-92, 2013.
[2] Y. Peng, C. Q. Jiang, and F. Xie, “Industrial control system cyber security research,” Journal of Tsinghua University (Sci & Tech), vol. 52, no. 10, pp. 1396-1405, 2013.
[3] Q. Xiong, X. W. Jing, and F. Zhan, “Summary and implications for China of the information security work of the ICS system in the oil and gas industry in America,” China Information Security, vol. 27, no. 3, pp. 80-83, 2012.
[4] V. M. Iigure, S. A. Laughter, R. D. Williams, “Security issues in SCADA networks,” Computers & Security, vol. 25, issue 7, pp. 498-506, 2006.
[5] D. L. Sun and D. M. Jiang, “Modbus/TCP protocol security and its application in industrial monitoring and control system,” Journal of Safety Science and Technology, vol. 2, no. 2, pp. 92-95, 2006.
[6] T. T. Wang, “Security Research on SCADA system data transmission,” M.S. thesis, Dept. Electron. Chinese, East China University of Science and Technology, Shanghai, China, 2012.
[7] Y. G. Zhang, H. Zhao, and L. N. Wang, “A non-parametric CUSUM intrusion detection method based on industrial control model,” Journal of Southeast University (Natural Science Edition), vol. 42, no. 1, pp. 55-59, 2012.
[8] J. Javier (July, 2011). Using SNORT for intrusion detection in MODBUSTCP/IP communications. [Online]. Available: http://www.giac.org/paper/gcia/7218/snort-intrusion-detection-modbus-tcp-ip-communications/124438
[9] P. Venkat and F. Matthew. Transparent Modbus TCP filtering with Linux [Online]. Available: http://modbusfw.sourceforge.net/
[10] C. M. Xia, T. Liu, H. Z. Wang et al., “Industrial control system security analysis,” Information Security and Technology, no. 2, pp. 13-17, 2013.
[11] Modbus Industrial Automation Network Specification—Part 1: Modbus Application Protocol, GB/T 19582.1-2008.
[12] Modbus Industrial Automation Network Specification—Part 3: Modbus Protocol Implementation Guide Over TCP/IP, GB/T 19582.3-2008.
[13] E. Kanpp, Industrial Network Security Securing Critical Infrastructure Networks for Smart Grid, SCADA, and other Industrial Control System, MA: Syngress, 2011, ch. 4, pp. 55-60.
[14] P. Huitsing, R. Chandia, M. Papa et al., “Attack taxonomies for the Modbus protocols,” International Journal of Critical Infrastructure Protection, vol. 1, pp. 37-44, 2008.
[15] K. L. Li, J. Z. Zhao, H. K. Huang et al., “An intrusion detection method based on SVM,” Information and Control, vol. 32, no. 6, pp. 495-499, 2003.
[16] G. C. Chen and J. S. Yu, “Particle swarm optimization algorithm,” Information and Control, vol. 34, no. 3, pp. 318-323, 2005.
[17] B. Jiang, N. Wang, and L. P. Wang, “Particle swarm optimization with age-group topology for multimodal functions and data clustering,” Communications in Nonlinear Science and Numerical Simulation, vol. 18, no. 11, pp. 3134-3145, 2013.
[18] F. J. Cabrerizo, E. Herrera-Viedma, and W. Pedrycz, “A method based on PSO and granular computing of linguistic information to solve group decision making problems defined in heterogeneous contexts,” European Journal of Operational Research, vol. 230, no. 3, pp. 624-633, 2013.
[19] X. G. Zhang, “Introduction to Statistical Learning Theory and Support Vector Machines,” ACTA Automation, no. 1, pp. 32-40, 2006.
[20] L. Li and X. L. Zhang, “Optimization of SVM with RBF kernel,” Computer Engineering and Applications, no. 29, pp. 190-204, 2006.