Hindawi
Security and Communication Networks
Volume 2021, Article ID 6690988, 11 pages
https://doi.org/10.1155/2021/6690988

Research Article

A Practical Format and Semantic Reverse Analysis Approach for Industrial Control Protocols

Qun Wang,[1] Zhonghao Sun,[2] Zhangquan Wang,[1] Shiping Ye,[1] Ziyi Su,[1] Hao Chen,[3] and Chao Hu[4]

[1]*Zhejiang Shuren University, Hangzhou, Zhejiang, China*
[2]*National Computer Network Emergency Response Technical Team, Coordination Center of China (CNCERT/CC), Beijing, China*
[3]*China Telecom Jiangxi Branch Provincial Network Operation and Maintenance Department, Nanchang, Jiangxi, China*
[4]*PLA Army Engineering University, Nanjing, Jiangsu, China*

Correspondence should be addressed to Chao Hu; huchaonj@126.com

Received 20 October 2020; Revised 23 February 2021; Accepted 4 March 2021; Published 13 March 2021

Academic Editor: Petros Nicopolitidis

Industrial control protocol is the basis of communication and interaction of industrial control system, and its security is related to the whole industrial infrastructure. Because many industrial control systems use proprietary protocols, it is necessary to adopt protocol reverse analysis technology to parse them and then detect whether there are secure vulnerabilities in the protocols by means of fuzzy testing. However, most of the existing technologies are designed for common network protocols, and there is no improvement for industrial control protocol. Therefore, we propose a multistage ensemble reverse analysis method, namely, MSERA, which fully considers the design concept of industrial control protocols. MSERA divides the traditional reverse analysis process into three stages and identifies the fields with different semantic characteristics in different stages and combines with field rectification to effectively improve the results of reverse analysis of industrial control protocols. Through the experimental comparison of some public and proprietary industrial control protocols, it is found that MSERA not only outperforms Netzob in the accuracy of field split but also far exceeds Netzob in semantic recognition accuracy. The experimental results show that MSERA is very practical and suitable for reverse analysis of industrial control protocols.

1. Introduction

Industrial control system is the automatic control system composed of computer equipment and industrial process control components. It is widely used in electric power, oil and natural gas, chemical, transportation, manufacturing, and other industries. It is the "brain" and "center pivot" of key national infrastructure. A complete industrial control system usually includes supervisory control and data acquisition (SCADA), distributed control system (DCS), programmable logic controller (PLC), and process control system (PCS). The control information and monitoring data transmission between SCADA and PLC are all based on industrial control protocols. The analysis and processing of the message of industrial control protocols affect the safety of industrial control systems. Compared with the common network protocols, industrial control network protocols have stronger control function. By sending special function code instructions, it can control the start, operation, and stop of physical equipment such as PLC and even affect the production process monitoring view of the control center.

With the development of industrial Internet, more and more industrial control systems begin to connect to the network and realize remote control of industrial control equipment through the network. This operation mode not only greatly promotes the industrial production efficiency but also brings opportunities for network attackers. Although the industrial control network is relatively closed, the occurrence of Stuxnet, Ukraine, and Venezuela power grid incidents and other security incidents show that attackers

can still attack industrial control facilities by using various technologies. Many of these network security incidents are attackers who use protocol vulnerabilities to achieve the invasion and destruction of industrial control facilities. Therefore, it is necessary to carry out corresponding interactive tests for industrial control equipment, so as to dig out the potential protocol vulnerabilities of the equipment. At present, the typical method is fuzzy testing technology, which uses a large number of semieffective data as the input of the target program and detects potential vulnerabilities by monitoring the anomalies of the program. However, many industrial control devices use nonpublic protocols for communication, which brings technical challenges to fuzzy testing, because if some key fields in the protocol are filled with incorrect data, the input will be directly discarded by the equipment as wrong messages, which will reduce the effect of fuzzy testing. In order to avoid the test case failure caused by some errors with special semantics in the input data, it is necessary to obtain the communication protocol knowledge between the industrial control computer and the client in advance and utilize this knowledge to improve the correctness of the input data and the effectiveness of the test cases, and inferring the specification of industrial control protocols relies on the protocol reverse analysis. Therefore, protocol reverse analysis plays a fundamental role in fuzzy testing and vulnerability mining of industrial control equipment.

The purpose of protocol reverse analysis is to obtain the protocol specification of unknown protocols, including protocol syntax, semantics, and synchronization information. Protocol syntax is embodied in the control information and the structure and format of control message. It defines the key words, data type, and length of each field in the protocol, and protocol semantics describes the meaning of each field and the corresponding constraints on the content of the field. The inference of protocol format and semantics is a very basic and important part in protocol reverse process, which determines the correctness of the input sample in fuzzy testing. The current protocol reverse analysis is based on network traffic and instruction execution trace, respectively. Relatively speaking, protocol reverse based on network traffic has strong generality and can be effectively applied to the analysis of unknown protocols. The work of this paper is to infer the format and semantics of industrial control protocol on the basis of network traffic.

Most of the existing protocol reverse approaches are based on the analysis of common network protocols. Some feature words or field separator (FD) with high average rate are mined out by statistical analysis and other methods and then as the basis of field division. There is a lack of special optimization for the industrial control protocols. Generally speaking, industrial control protocol is mainly designed for industrial control system, and its function is to perform corresponding operation on industrial control equipment. Therefore, the form is relatively simple and the content is relatively fixed. In structure, it is usually composed of control part and data part. Each byte or even each bit of the control part has a specific function and meaning. Therefore, it is difficult to use special characters or feature words to divide the fields of industrial control protocols, but we can use this feature to analyze the fields with significant semantic features in advance, and then these analysis results are substituted into the subsequent reverse process.

In this paper, we propose a multistage ensemble reverse analysis method, namely, MSERA, which fully considers the design concept of industrial control protocols and divides the traditional single protocol reverse process into the following three stages:

(i) First stage: based on the heuristic analysis of typical semantic fields of industrial control protocols, the corresponding analysis methods are adopted to infer the location, scope, and semantics of these fields, and the semantic priority method is proposed to solve the overlapping problem of some data belonging to two semantic fields at the same time.

(ii) Second stage: the semantic fields extracted in advance are substituted into the analysis process, and then the high similarity groups are divided into the same category via sequence alignment, and the packets in the same category are divided into fields according to the dynamic change characteristics of the fields.

(iii) Third stage: according to the division results of the second stage, some semantics which cannot be obtained by premining are reanalyzed, and the corresponding semantics are obtained. In addition, field rectification is presented to correct some abnormal fields, so as to improve the accuracy of protocol reverse.

Through the integration of multistage analysis results, most of the field formats of industrial control protocols can be correctly divided and the semantic information can be mined out. This information will be helpful to improve the effectiveness of fuzzy test cases.

The rest of this paper is organized as follows: Section 2 introduces the challenges of protocol reverse analysis problems and the inspiration of industrial control protocol characteristics on the reverse process. Section 3 introduces the technical details of multistage ensemble reverse analysis method. Section 4 analyzes some public and proprietary industrial control protocols by using our analysis tools. Section 5 introduces the related work and Section 6 summarizes the full paper.

2. Challenges and Opportunities

2.1. Practical Problems in Protocol Reverse. The reverse analysis of protocol syntax usually includes two parts: dividing the message fields and identifying the field semantics after division. The typical protocol reverse process based on sequence alignment generally includes several parts as shown in Figure 1, and the main work is as follows:

(i) Message clustering: by calculating the similarity between messages, the messages with high similarity are divided into one kind of packets, so as to aggregate the messages with the same format as far as

possible, which is helpful to improve the accuracy of the subsequent analysis process.

(ii) Field split: match the packets in the same category, and divide the parts with the same characteristics into the same field according to the change characteristics of the packet content, so as to realize the division of message fields in each category.

(iii) Semantic inference: the message content in the same field is analyzed and matched with the specific semantic features to determine the semantics of the field, so as to ensure that the generated test message has the syntax characteristics of the real message.

However, the abovementioned reverse process faces many challenges which are difficult to overcome in practice.

(i) The accuracy of clustering results is the basis of field split and semantic inference. Once the result of clustering is not accurate, some packets that do not belong to the same message format are grouped together or the packets belonging to the same message format are classified into different categories, which will lead to incorrect field split. Moreover, the data field in the real communication message of industrial control system can easily affect the accuracy of message clustering.

(ii) The results of field split have a decisive impact on the final semantic recognition. The incorrect division of message structure will lead to the failure of semantic analysis. Some fields are formally characterized as the combination of fixed fields and variable fields (such as time stamp and multibyte sequence number), and some fields have the same characteristics as data fields (such as CRC fields). Only according to the change characteristics of data content, the results of field split will be impacted.

2.2. Inspirations from the Characteristics of Industrial Control Protocols. Through the practical problems faced by the reverse analysis of industrial control protocol, we can find that the root of the problem presented in the previous section is that the traditional analysis process separates the split of protocol format from the inference of field semantics. The calculation of message similarity and clustering results determine the result of field split, and the result of field split determines the result of semantic inference. The pipeline process does not feedback some information that can be discriminated in the later stage to the previous stage, which makes it difficult to analyze the format, position, and semantics of each field accurately and effectively.

Compared with many complex network protocols, the message format of industrial control protocol is relatively simple. Most of the protocol messages are relatively short, the field division is relatively clear, and the protocol function is relatively fixed. Some fields show obvious semantics, especially many fields with special semantics are located in the same position in different messages, which gives us the opportunities to accurately analyze the protocol format. Although many industrial control protocols adopt proprietary protocols and their message format is not open to the public, they still have the inherent characteristics of industrial control protocols. Table 1 lists some widely used fields and their characteristics in industrial control protocols.

In addition to the typical fields listed in Table 1, some industrial control protocols, such as S7Comm and IEC 61850 MMS, all adopt the block-oriented ISO transmission service. The protocol data unit (PDU) of these protocols is encapsulated in TPKT and ISO-COTP protocols. Therefore, according to the format characteristics of TPKT and ISO-COTP protocols, we can analyze whether the industrial control protocols adopt ISO transmission services in advance, so as to identify the fields contained in these protocols.

In the abovementioned typical fields, protocol identifier, sequence number, site identifier, and other fields can be identified according to the characteristics of the messages, but the structure identifier field can be determined according to the relationship between the field value and the message structure after the reverse analysis of the message structure. In addition, the length reference field can only be determined according to the calculation results of length value after field division.

Therefore, during the analysis of industrial control protocol, the process should be divided into different stages according to the semantic characteristics of the fields. Firstly, the fields with obvious semantic features in the protocol are identified, and this information is put into the subsequent analysis process. After the field division is completed, the semantics that can only be determined after the message structure are determined can be mined, so as to improve the performance. This paper discusses the integration of format inference and semantic mining in reverse analysis to improve the accuracy.

3. Multistage Ensemble Reverse Analysis on Industrial Control Protocols

Based on the abovementioned ideas, we propose a multistage ensemble analysis method, i.e., MSERA, for industrial control protocols. The whole analysis process is divided into three stages: presemantic mining of fields state, sequence alignment and field split stage, resemantic mining, and field rectification stage.

3.1. Presemantic Mining of Fields. The task of presemantic mining of fields is to analyze the samples by comparing the characteristics of typical semantic fields in industrial control protocols, so as to find out whether there are some significant semantic fields in this protocol and then deliver the information of these semantic fields to the next stage for analysis. Therefore, two problems need to be solved in this stage: the first is how to identify the semantic fields, while the second is how to transmit the identified field information to the subsequent analysis.

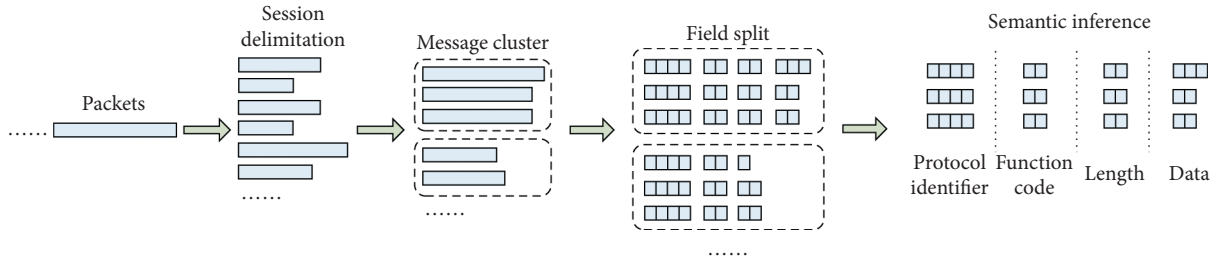

Figure 1: The typical protocol reverse process based on sequence alignment.

Table 1: Typical fields and their semantic features in industrial control protocols.

No.	Fields	General characteristics	Detailed features
1	Protocol identifier	Fixed and static	A number of bytes at the beginning of all messages and the length and content of this field are fixed
2	Length	Fixed and dynamic	The value of this field is related to the length of some fields or whole message
3	Sequence number	Fixed and dynamic	After all packets are sorted by the captured timestamp, the value of this field will be accumulated by 1
4	CRC	Fixed and dynamic	The value of this field is the same as the CRC calculation value of some fields or whole message
5	Timestamp	Fixed and dynamic	After all packets are sorted by the captured timestamp, the difference of this field between adjacent packets is close to that of captured timestamp
6	Function code	Fixed and optional	The value of this field is related to the operation function of industrial control equipment, and the value of message has several options
7	Structure identifier	Fixed and optional	The value of this field is related to the structure of the message, and the value of the message has only a few options
8	Site identifier	Fixed and optional	The value of this field is related to the source IP address, and the source site and destination site identifier usually appear in the message at the same time
9	Directory	Variable and dynamic	This field is generally used for indicating file position, which is formally represented as printable strings
10	Parameter	Variable and dynamic	The entropy of this field is very high, and its length is not fixed. It usually appears in the middle of the message
11	Data	Variable and dynamic	The entropy of this field is very high, and its length is dynamic. It usually appears at the end of the message

3.1.1. Identification of Semantic Fields. In many industrial control protocols, the fields with special semantics are located in the same position in most or even all messages, so a heuristic method can be used to identify these semantic fields.

(1) Identification of protocol identifier:

Field description: some industrial control protocols have several bytes of protocol identifier field at the beginning of each message, which is used by the receiver to confirm whether the received message belongs to the industrial control protocol. For example, the first byte of messages in IEC104 is 0 x 68, the first byte of messages in BACNet is 0x81, the protocol identifier field of DNP3 is 0 x 0564, and the first two bytes of a proprietary heat metering protocol are 0 x f0f0.

Identification approach: since the protocol identifier field is static at the beginning of the message and the length of the field is also fixed, the several bytes in the front of all messages are compared. If the first N bytes of all messages are the same, the first N bytes can be determined as the protocol identification field.

(2) Identification of length field:

Field description: most of the industrial control protocols have a length field to describe the size of a message. Generally, the length field can be divided into two categories: one is to indicate the length of the whole message, while the other is to calculate some variable fields (such as data fields). The former type of length field can be identified in the presemantic mining stage, while the latter type relies on the result of field split. Therefore, in this stage, the former type of length field is mainly recognized.

Identification approach: firstly, all packets are clustered according to their length, so that packets with the same length are divided into one category, and then the contents of consecutive N bytes of messages in the same position in each category are analyzed (because the general length field is not too long, N is not larger than 4 bytes). If all the messages

in the same category have the same value and the values of different categories are different, we calculate the difference of the values of different categories. If the difference equals to the gap of message length and the field value does not exceed the total length of the message this field is determined as the length field.

(3) Identification of sequence number field:

Field description: many industrial control protocols use the sequence number field to deal with packet loss and disorder. Different from the traditional TCP protocol, the sequence number field in industrial control protocol is usually used to count the number of packets. Therefore, when the packets with the same flow direction are sent in sequence, 1 will be added in turn.
Identification approach: the packets with the same five tuple information, i.e., source/destination IP, source/destination port, and transport protocol, are clustered and sorted according to the timestamp marked when the packets are captured. Then, the value of consecutive N bytes in the same position of the message is analyzed (because the sequence number field is not too long, N is limited to 4 bytes), and the difference of this part of adjacent messages is calculated. Considering that packet loss may occur in the actual process, the field is determined to be the sequence number field if 90% of the differences is 1.

(4) Identification of CRC field:

Field description: in order to ensure that there is no error during the data transmission, some industrial control protocols will verify the message via CRC. Generally speaking, CRC is designed in the last part of message structure. In addition, some industrial control protocols will have multiple CRC fields at the same time. For example, DNP3 will add a two-byte CRC 16 check field after each data block. Therefore, CRC field can be found by calculating the CRC value and comparing with subsequent content.
Identification approach: first, we extract the last two bytes of the message and derive 2 to N bytes from the back to the front, where the value of N is the message length minus 2, and calculate the CRC16 value of these bytes. If they are equal, record the CRC16 field, the position of the reference field, and the CRC16 type. At the same time, we derive the two bytes before the found reference field to find out if there exist other CRC16 fields. If more than 90% of all packets have CRC16 field in the same type, the field is determined as CRC16 field. For the CRC32 field, except its length of 4 bytes, the rest of the determination process is consistent.

(5) Identification of timestamp field:

Field description: some industrial control protocols will load the timestamp information into the message to indicate the time of the data. If the data are real-time response, the timestamp field has a strong correlation with the timestamp information added by packet capture tools. Therefore, this feature can be used to identify the timestamp field.
Identification approach: all packets are sorted according to the captured sequence, and then the bytes of two adjacent packets at the same position from the beginning are converted into timestamps. If both of them can be converted into valid timestamps with a time difference of no more than 5 years from the current time, the difference between the two timestamps is calculated. At the same time, the difference between the captured timestamps of two adjacent packets is calculated. These bytes are marked as possible timestamp fields if the difference between them is no more than 5 seconds. If more than 90% of the bytes in the same position of all packets are marked as a timestamp field, the field is determined as a timestamp field.

(6) Identification of function code field:

Field description: function code field is the most common field in industrial control protocols, which is usually used for the client to perform particular function operation on industrial control equipment. Function code field usually has only a few available options, which makes its entropy value different from other fields, so it can be used to locate function code field.
Identification approach: entropy is calculated for the content of all messages in the same location. At the same time, an empirical value is used to set the threshold of the entropy value of the function code. Specifically, the frequency of the occurrence of the field value should be evenly distributed among the 1/3 options of all possible values. The entropy value obtained in this case is compared with the actual entropy value of the message. If the entropy value of the message is lower than the threshold value and the message content is not a fixed field, the field is determined as a function code.

(7) Identification of site identifier field:

Field description: the site identifier field is a field for the client and server to identify each other's identity information in industrial control protocols. The site identifier field of both sides of the communication

usually exists in a message at the same time, and the value of the site identifier is related to the message's source address information, so this feature can be used to classify the field.

Identification approach: all messages are classified according to source addresses, and then the contents on the same location are matched. If the contents of messages in the same category are equal and the contents of different categories and locations are different, the field is determined as the site identifier field.

3.1.2. The Priority Settings of Semantic Fields. In the process of analyzing the above semantic fields, because the semantic features are not orthogonal to each other, some fields will be consistent with multiple semantic features at the same time. For example, in the heartbeat message of a proprietary heat metering protocol, the length of request message and response message is inconsistent, and the whole heartbeat protocol only has two message formats, which will make the length field determined as the site identifier field and function code field. Therefore, it is necessary to set priorities for fields that meet multiple semantic characteristics at the same time, so as to classify them into the correct field types.

In order to achieve more accurate analysis of field semantics, we set up the judgement matrix as shown in Table 2. The unit value of the matrix indicates that when the field conforms to two semantics at the same time, the element should be selected as the result. The principle of setting the table is to match the more complex semantics of features first and combine with the general rules of industrial control protocol in design.

In Table 2, we do not add the protocol identifier, sequence number, CRC, and timestamp fields with strong unique semantic characteristics, because these semantics basically do not have the same feature as other semantic fields.

3.1.3. Transformation of Identified Fields. After the heuristic method is used to identify the fields with typical semantics, the information needs to be transmitted to the subsequent analysis process, and the transmission of this information should be closely coordinated with the method to be adopted in the second stage, so as to ensure that the method in the later stage will not destroy the analysis results of the previous stage but also effectively use the analysis results to improve the accuracy of field split.

In the current major protocol reverse methods, sequence alignment algorithm is a classic algorithm. This algorithm can effectively determine the similarity between two or more sequences, so as to determine whether they are homologous. Moreover, in the process of alignment, replacing some of them with clear field content will not affect the analysis of other fields.

Therefore, after identifying the fields with particular semantics, we firstly replace the contents of these fields with special characters that are not often seen in industrial control protocols and record the original content and location of each converted field. Since these semantic fields are matched in a more rigorous way in the pre-semantic mining stage, the value in the field not only is required to fully conform to the semantic characteristics but also has the same relative position in the message. The identified contents with the same semantics will be divided into the same cluster in the sequence alignment stage. In addition, these converted special characters can effectively improve the similarity of messages with the same structure, which not only ensures that the previous analysis results are not damaged but also further improves the accuracy of analysis.

3.2. Sequence Alignment and Field Split. Because of the high computational complexity of sequence alignment, we first use DBSCAN clustering algorithm to divide packets with the same message structure into a set before similarity calculation and then conduct sequence alignment, so as to reduce the number of matches between messages not belonging to the same structure and speed up the analysis.

3.2.1. DBSCAN-Based Packet Clustering. DBSCAN is a density-based clustering algorithm. It defines a cluster as the largest set of density connected points. It can divide the region with enough high density into clusters. Under the initial conditions, it is impossible to know exactly how many classes of packet structure there are, and DBSCAN is a clustering algorithm independent of the number of categories, so it can effectively adapt to this scenario. When DBSCAN is used to cluster groups, the length is the only attribute, so it is used as the attribute for clustering.

The steps to preprocess the sample set with DBSCAN method are as follows:

(i) There are two parameters to be input for DBSCAN, namely, scanning radius eps and minimum number of included points minPts, which need to be determined in advance.

(ii) Select a packet that has not been visited and find out all nearby packets whose packet length difference is within eps. If the number of adjacent packets is not less than minPts, then the current point and its nearby points form a cluster, and the starting packet is marked as visited. Then, recursion is used to process all the groups in the cluster that are not marked as visited in the same way, so as to expand the cluster.

(iii) If the number of adjacent packets of the departure packet is less than minPts, the packet is temporarily marked as a noise point.

(iv) If the cluster is fully extended and all packets in the cluster are marked as visited, the same algorithm is used to process the other unvisited packets.

In the practical implementation, the length of industrial control packets is generally less than hundreds of bytes, the scanning radius is set to 10 bytes, and the minimum number of included points is set to 1.

Table 2: Judgement matrix of industrial control protocol field.

Semantic 2 Semantic 1	Length	Function code	Site identifier
Length	—	Length	Length
Function code	Length	—	Site identifier
Site identifier	Length	Site identifier	—

3.2.2. Sequence Alignment-Based Packet Structure Reverse Analysis. Sequence alignment algorithm is used to realize the protocol reverse and field split, and we choose Needleman-Wunsch algorithm to find out the similarity between the sequence to be tested and the target sequence. The main process includes three parts, which are similarity scoring, scoring and summation, and optimal backtracking, respectively.

(i) Similarity scoring: for two message sequences with the length of m and n, the algorithm first constructs a similarity matrix S of $(n+1)^*(m+1)$, and the subscripts are marked by i and j, respectively. Generally, for the matched symbols, the similarity score is 1; otherwise, the similarity score is 0.

(ii) Scoring and summation: iteratively sum the similarity matrix to obtain a new matrix M. The summation formula is as follows, where the space penalty w is set to 0:

$$M_{ij} = \max \begin{cases} M_{i-1,j-1} + S_{ij}, \\ M_{i,j-1} + w, \\ M_{i-1,j} + w. \end{cases} \quad (1)$$

(iii) Optimal backtracking: the matrix element with the highest score is traced back to the starting position. The left side, left upper diagonal, and upper side of the element were examined, respectively, and moved to the adjacent element with the highest score. When the three scores are the same, move to the upper left diagonal element first. If moving to the left, insert a space in the vertical sequence; if moving to the upper side, insert a space in the horizontal sequence; otherwise, do nothing.

In the abovementioned process, it is necessary to set the similarity threshold in advance. When the similarity value of two packet sequences exceeds the threshold, they can be divided into the same message category. Next, when dividing the fields of groups in the same message category, static data and dynamic data are used to identify, and the classification is based on the recognition results of static data and dynamic data. The specific example is shown in Figure 2.

In Figure 2, the length and content of static fields are fixed, while the length and content of dynamic fields may change. Therefore, after the static fields are aligned, all groups belonging to the message type can be split into corresponding fields according to the abovementioned field split method.

3.3. Resemantic Mining and Field Rectification

3.3.1. Remining on Field's Semantics. Although some typical semantic fields have been analyzed in the presemantic mining stage, some semantic fields cannot be identified. Only after all the fields are split, their semantics can be inferred. These semantics include structure identifier field, control part length field, and length reference field.

(1) Identification of structure identifier field

Field description: some industrial control protocols define structure identifier field to help the receiver to analyze the message content. Therefore, the content of this field determines the format of the whole message structure, and different field contents correspond to different message structures. In addition, in order to ensure that the receiver can parse the message successfully, the field is located in the same position in different structure messages.
Identification approach: the field in the same position after division is analyzed. If the field value of the message with the same structure is equal and the field value of message structure is different, then the field is determined as the structure identifier field. It should be noted that in some cases, the structure identifier field overlaps with the function code field, which is the analysis deviation caused by the design of industrial control protocols.

(2) Identification of control part length field

Field description: some industrial control protocols not only use length field to indicate the size of the whole message or the size of the data part but also use the control part length field to specify the size of the control part, so as to ensure the scalability and backward compatibility of the protocol. However, the size of the message control part is often same, which leads to the characteristics of a static field. Moreover, the length field cannot be identified in the presemantic mining stage, and it can only be determined by comparing the value of the field with the length of the control part after the message field is split.
Identification approach: analyze the fields in the same position after division. If the values of the field in all messages are same and equal to the sum of several fields in the front of the message, then the field is determined to be the control part length field.

(3) Identification of length reference field

Field description: the length reference field is not a specific field like other fields, but the message area corresponding to the length field. This area may only be a data field, or it may cover several fields. Although this field will not have a great impact on the results of protocol reverse analysis, it plays an important role in the fuzzy test. It is necessary to

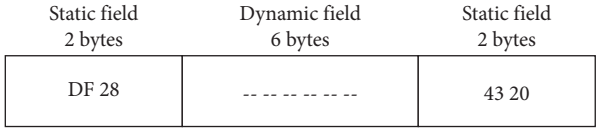

Figure 2: Example of field split.

identify the message area referenced by the length field.

Identification approach: the content length of all consecutive fields is counted from the first field. If the statistical value of all messages is equal to the value of length field, the area covered by these consecutive fields is determined as the length reference field.

3.3.2. Field Rectification. In practical protocol reverse analysis, some function code, sequence number, structure identifier, and other fields are composed of two or more bytes. In the case of insufficient samples, this field will show the characteristics of "static byte + dynamic byte," in which the static byte is often 0x00. After the analysis of the first two stages, the field in this form is often divided into two fields. The first field is a static field with all zeros, and the second field is a field with special semantics. However, in the actual industrial control protocols, useless fields are often not designed, so the two fields need to be combined. It should be noted that the industrial control protocol may adopt either high-order or low-order. When merging, it is necessary to determine whether to merge with the left field or the right field according to the position of the static field. Generally speaking, the merged fields usually start from odd byte and end with even byte. Therefore, this heuristic method can be used to correct the fields.

4. Performance Evaluation

In order to evaluate the performance of the multistage ensemble reverse analysis method, i.e., MSERA, proposed in this paper, based on Netzob [1], we implement field split and semantic recognition. Netzob is a protocol reverse analysis tool developed by Bossert et al. It has been packaged as a python third-party library. We can realize the reverse analysis of protocol by calling the interface in the library. At present, we mainly use Netzob to implement the sequence alignment algorithm in the second stage.

In addition, in order to evaluate the final analysis effect, we select some typical and proprietary industrial control protocols for reverse analysis and evaluate the analysis results from the accuracy of field division and semantic recognition. The definition of accuracy rate of field split is the percentage of correctly identified fields in all fields of industrial control protocol, and the definition of semantic recognition accuracy rate is the percentage of correctly recognized field semantics in all fields of industrial control protocol. Their specific descriptions are shown in the following equations:

$$\text{Field split accuracy} = \frac{\#\text{correctly identified fields}}{\#\text{all fields in protocol}}, \quad (2)$$

$$\text{Semantic recognition accuracy} = \frac{\#\text{correctly identified fields}}{\#\text{all fields in protocol}}. \quad (3)$$

At the same time, in order to verify the effectiveness of MSERA, we also compared it with Netzob. The specific analysis results are shown in Table 3.

It can be found from Table 3 that MSERA method is obviously better than Netzob because MSERA has premined the semantics of industrial control protocols and successfully identified some fields with obvious semantic features. At the same time, combining with sequence alignment and field rectification, the recognition accuracy is effectively improved, especially Netzob basically only has length field; MSERA can identify most of the typical semantics in industrial control protocols.

For several protocols with relatively poor analysis results, the reason for the low recognition accuracy of OMRON_FINS is that the second to fourth fields are reserved, gateway count, and destination address, respectively. However, from the final results, the target network address is composed of three bytes, and the first two bytes of the former source address and destination address are the same, which leads to their incorrect combination with reserved bytes and gateway count, which reduces the accuracy of analysis. The reason for the low recognition accuracy of BACnet is that the difference of byte number between different format packets is very small, and the accuracy of analysis results is reduced after sequence alignment. Heidenhain protocol is a proprietary protocol; the sample only contains the message to transfer the spindle rotation operation, resulting in fewer changes in the message. Some of the field contents have not changed and are combined together in the test process, which reduces the analysis accuracy.

Through the analysis of these samples, it can be found that MSERA can better analyze the field format and semantics of industrial control protocols, but to achieve higher accuracy, we need to resort to the diversity of samples to ensure that it can reflect the inherent characteristics of the protocol. In addition, when the contents of several consecutive fields are not changed during protocol design, MSERA's analysis method will be invalid for these fields, which is a inherent difficulty that cannot be overcome by network traffic-based protocol reverse.

5. Related Works

Protocol format analysis mainly completes protocol field split and field semantic inference. Generally speaking, it can be divided into four categories: format inference algorithm based on sequence alignment, format inference algorithm based on probability model, format extraction algorithm based on frequent sets, and format inference algorithm based on semantics.

Table 3: Analysis results of industrial control protocol.

Protocol	Field split accuracy		Semantic recognition accuracy	Semantic recognition accuracy
	Netzob	MSERA	Netzob (%)	MSERA (%)
OMRON_FINS	37.5	62.5	0	62.5
IEC104	38.5	74.03	14.1	49.6
BACNet	50	66.7	0	66.7
DNP3	44.4	77.8	11.1	77.8
Modbus	50	75	12.5	75
S7Comm	52	84	12	76
Huada Zhibao heat metering system	53.6	86.6	13.4	67.0
Germany Heidenhain numerical control system	50	66.7	0	50

Among the format inference methods based on sequence alignment, PI is one of the earliest successful aprE tools [2]. It pioneers the introduction of bioinformatics methods into protocol reverse. They all recognize patterns in long data sequences, and the message alignment method used in PIP algorithm has been reused in many tools. Cui et al. proposed a recursive clustering-based format extraction scheme Discoverer [3], which uses the format flag field to analyze the message substructure. In the initial clustering stage, firstly, according to the text and binary attributes, the message text throttling is segmented to get the message attribute sequence. Then, the sequence alignment algorithm is used to compare the message attribute type sequence, and the initial clustering of the message is carried out according to the comparison result. Bossert et al. proposed a semantic PI [4], which based on the existing PIP framework, added semantic considerations in sequence alignment and message format clustering. Meng et al. [5] proposed a multiple sequence alignment based on HMM (MSA-HMM). Firstly, all the groups were divided by K-means clustering algorithm, and then the contour coefficient (siloette) in the algorithm was evaluated The K value is adjusted and the UPGMA algorithm is used to iterate to get the final classification number. Secondly, for each group, the improved guide tree progressive multiple sequence alignment method is used for grouping. The transition probability of HMM matrix is calculated for the two compared sequences, and then the progressive multiple sequence comparison tree is constructed to segment the groups. Finally, the fields are merged according to the change rate, mean value, and variance of adjacent fields in the segmented group. Esoul and Walkinshaw [6] proposed a format extraction algorithm based on message segmentation (segment-based NW) in order to avoid the decision error caused by different message lengths. The basic idea is that the packets are compared by multiple sequences based on segmentation, and then the analysis results are combined by weighted coefficients.

In the format inference algorithm based on probability model, Wang et al. proposed two methods, Biprominer [7] and ProDecoder [8], to find keywords through statistical methods. Biprominer is developed by pointer to binary protocol. It includes three stages: (1) determining the field length and keywords by counting the correlation of frequency dependent patterns of each byte. (2) In the second stage, different messages are associated with keywords, which introduce Lempel Welch compression algorithm [9] applied in the field of information compression and find keywords by increasing the value of N in n-gram algorithm. ProGraph [10] is the first tool that can perform format inference for bit and byte at the same time, which is mainly based on the dependency between different fields in the same group. ProGraph constructs graph model to analyze the new correlation among multiple packets of the target protocol. The algorithm is based on each packet and assumes that the information of the protocol is always contained in the first few bits of the packet, so it can be used to analyze the encryption protocol. Cai et al. [11] proposed a method based on hidden semi-Markov modeling (HSMM) to determine the length of keywords and protocol fields. The algorithm obtains the original packet by Tshark and then completes the protocol analysis through session reconstruction, message reorganization, HSMM, message segmentation, and type inference.

In the format extraction algorithm based on frequent sets, Krueger et al. [12] proposed an automatic semantics aware analysis of network payloads (ASAP), first extract the relevant string table from the network traffic and then match these data. If it can be matched, add 1 to the corresponding bit in the matrix to get a high-dimensional vector matrix; second, decompose the obtained matrix and use PCA and NMF algorithm to get the vector corresponding to the main components of the matrix; third, decompose the matrix to get the moment arrays which are represented as disjunctive expressions and merged to get a semantic template that can represent typical communication content. AutoReEngine algorithm [13] refers to the alphabet method of ASAP, regards each byte as a keyword, and then calculates whether the frequency of each string exceeds a certain threshold; if it exceeds, it will be added to the frequent set; then, add a byte to recount whether it exceeds a certain threshold, merge the overlapping keywords, calculate their frequency, and determine whether it is a keyword until the frequent set is a closed string. Wang et al. [14] proposed a format inference algorithm (ac-fp algorithm) for wireless networks. The improved AC algorithm was used to determine the frame boundary, Apriori algorithm was used to extract character features, and FP growth algorithm was used to realize field association analysis. Based on similar ideas, Hei et al. [15] proposed a new method based on AC and AC apriori algorithm, by changing the length of keywords, using Zipf

distribution to count the keywords, only the first ranked as keywords, and then using AC algorithm to count the frequency of keywords in all groups, and then mining frequent sets through apriori algorithm, so as to obtain the possible message format, and then according to the support degree of the message, it carries on the statistics reduction. Fan et al. [16] thought that the previous format inference algorithms ignored the correlation between the front and back order of the fields, so they proposed an improved algorithm SPREA based on FP-tree [17]. The algorithm used information entropy to infer the encrypted fields in the group, studied how to determine the boundary of encrypted word segments, and conducted state machine inference based on Prospex algorithm [18]. IPART algorithm [19] is an algorithm designed for reverse analysis of industrial control protocol. Firstly, the grouping is decomposed into different field order, and then the group type is identified, and then the type inference and state machine construction are carried out.

In the semantic-based format extraction algorithm, FieldHunter [20] includes two module field extractors and field type inference tools. The field extractors are divided into key value extractors and single word extractors, and n-gram extraction extractors handle text protocols and binary protocols, respectively. In the extraction of binary character keywords, high-frequency strings are found as alternative protocol keywords by changing the value of N. The key to text protocol analysis is to determine the separator, which is mainly obtained by counting the symbol frequency of nonletters. After the fields are determined, the field type inference tool is responsible for determining the semantics of each field. Choi et al. proposed WASp (WPAN automatic spoofer) [21] for wireless protocol based on IEEE 802.15.4. The tool can identify the fields in byte-level and generate packets for attack. The algorithm cannot recognize the encryption protocol, which is due to the large amount of energy consumed by encryption computing in low-power protocol.

On the basis of protocol reverse analysis results, combined with fuzzy testing technology, it can effectively improve the safety detection effect of industrial control equipment. Li proposed a vulnerability mining method that can automatically identify various network protocols and generate fuzziers for fuzzy testing [22]. This method can automatically identify network protocol packet structure and conduct fuzzy testing through multiple stages of packet classification, multiple sequence alignment, specific domain identification, and fuzzier generation. Wang proposed an improved effective technical method based on local greedy algorithm to improve the accuracy of reverse keyword extraction [23]. Combined with the fuzzy testing framework Sulley, the vulnerability of industrial control protocol was mined and integer overflow was found. In order to solve the problem of low degree of automation in application layer network protocol fuzzy testing scheme, Zhang proposed an application layer protocol fuzzy testing scheme based on network protocol reverse, which can automatically construct test cases and effectively improve the efficiency of protocol vulnerability mining [24]. In addition, quantum communication also brings a severe test to the network communication security, and we need to deal with these new challenges [25].

6. Conclusions

It is very important for protocol analysis and testing to infer the format structure and field semantics of industrial control protocol, which is related to the security of the whole industrial control system. Current protocol reverse analysis is mainly designed for common network protocol, which is lack of pertinence of industrial control protocol characteristics. Therefore, the results of field reverse analysis are not accurate and semantic recognition is difficult. In order to solve the abovementioned problems, we propose a practical reverse protocol method MSERA for industrial control protocols. This method mines these fields with special semantics through the heuristic method for typical fields often used in industrial control protocol design and then combines sequence alignment and field rectification to achieve accurate split of fields in the protocol. Through the analysis of some public and proprietary industrial control protocols, it can be found that MSERA is not only more accurate than Netzob in the field split results but also greatly improves the semantic mining results. At the same time, we also analyze the problems of some protocols which are not good. The fundamental reason is that network traffic-based method is difficult to overcome. In our future work, we plan to combine it with an active interactive method to overcome the influence of sample on the reverse results of protocol.

Data Availability

The sample data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (nos. 2017YFB0801703 and 2017YFC1201204), CERNET Innovation Project (no. NGII20170406), Foundation for Training Postdoctoral Innovative Talents in Southeast University (no. 2242019R20024), and Zhejiang Province Public Welfare Research Project (no. LGG20F020014).

References

[1] F. Guihéry and G. Bossert, "Netzob documentation [EB/OL]," 2014, http://www.netzob.org.

[2] M. Beddoe, "The protocol informatics project," 2004, http://phreakocious.net/PI/.

[3] W. Cui, J. Kannan, and A. H. J. Wang, "Discoverer: automatic protocol reverse engineering from network traces," in *Proceedings of the Usenix Security Symposium*, Boston, MA, USA, August 2007.

[4] G. Bossert, F. Guihéry, and G. Hiet, "Towards automated protocol reverse engineering using semantic information," in *Proceedings of the 9th ACM symposium on Information, Computer and Communications Security*, pp. 51–62, Kyoto, Japan, June 2014.

[5] F. Meng, C. Zhang, and G. Wu, "Protocol reverse based on hierarchical clustering and probability alignment from network traces," in *Proceedings of the 2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA)*, pp. 443–447, IEEE, Shanghai, China, March 2018.

[6] O. Esoul and N. Walkinshaw, "Using segment-based alignment to extract packet structures from network traces," in *Proceedings of the 2017 IEEE International Conference on Software Quality, Reliability and Security (QRS)*, pp. 398–409, IEEE, Prague, Czech Republic, July 2017.

[7] Y. Wang, X. Li, J. Meng et al., "Biprominer: automatic mining of binary protocol features," in *Proceedings of the 2011 12th International Conference on Parallel and Distributed Computing, Applications and Technologies*, pp. 179–184, IEEE, Gwangju, Korea, October 2011.

[8] Y. Wang, X. Yun, M. Z. Shafiq et al., "A semantics aware approach to automated reverse engineering unknown protocols," in *Proceedings of the 20th IEEE International Conference on Network Protocols (ICNP '12)*, pp. 1–10, IEEE, Austin, TX, USA, November 2012.

[9] T. A. Welch, "A technique for high-performance data compression," *Computer*, vol. 17, no. 6, pp. 8–19, 1984.

[10] Q. Huang, P. P. C. Lee, and Z. Zhang, "Exploiting intra-packet dependency for fine-grained protocol format inference," in *Proceedings of the 14th IFIP Networking Conference (NETWORKING '15)*, Toulouse, France, May 2015.

[11] J. Cai, J. Z. Luo, and F. Lei, "Analyzing network protocols of application layer using hidden semi-Markov model," *Mathematical Problems in Engineering*, vol. 2016, Article ID 9161723, 14 pages, 2016.

[12] T. Krueger, N. Krämer, and K. Rieck, "ASAP: automatic semantics-aware analysis of network payloads," in *Proceedings of the International Workshop on Privacy and Security Issues in Data Mining and Machine Learning*, pp. 50–63, Springer, Barcelona, Spain, September 2010.

[13] J.-Z. Luo and S.-Z. Yu, "Position-based automatic reverse engineering of network protocols," *Journal of Network and Computer Applications*, vol. 36, no. 3, pp. 1070–1077, 2013.

[14] Y. Wang, N. Zhang, Y. Wu et al., "Protocol formats reverse engineering based on association rules in wireless environment," in *Proceedings of the 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, pp. 134–141, IEEE, Melbourne, Australia, July 2013.

[15] X. Hei, B. Bai, Y. Wang et al., "Feature extraction optimization for bitstream communication protocol format reverse analysis," in *Proceedings of the 2019 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/13th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE)*, pp. 662–669, IEEE, Rotorua, New Zealand, August 2019.

[16] Y. Fan, Y. Zhu, and L. Yuan, "Automatic reverse engineering of unknown security protocols from network traces," in *Proceedings of the 2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, pp. 1139–1148, IEEE, Chengdu, China, December 2018.

[17] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *ACM Sigmod Record*, vol. 29, no. 2, pp. 1–12, 2000.

[18] P. M. Comparetti, G. Wondracek, C. Kruegel et al., "Prospex: protocol specification extraction," in *Proceedings of the 2009 30th IEEE Symposium on Security and Privacy*, pp. 110–125, IEEE, Oakland, CA, USA, May 2009.

[19] X. Wang, K. Lv, and B. Li, "IPART: an automatic protocol reverse engineering tool based on global voting expert for industrial protocols," *International Journal of Parallel, Emergent and Distributed Systems*, vol. 35, no. 3, pp. 376–395, 2020.

[20] I. Bermudez, A. Tongaonkar, M. Iliofotou, M. Mellia, and M. M. Munafò, "Towards automatic protocol field inference," *Computer Communications*, vol. 84, pp. 40–51, 2016.

[21] K. Choi, Y. Son, J. Noh et al., "Dissecting customized protocols: automatic analysis for customized protocols based on IEEE 802.15. 4," in *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, pp. 183–193, Darmstadt, Germany, July 2016.

[22] W.-M. Li, A.-F. Zhang, J.-C. Liu, and Z.-T. Li, "An automatic network protocol fuzz testing and vulnerability discovering method," *Chinese Journal of Computers*, vol. 34, no. 2, pp. 242–255, 2011.

[23] H. X. Wang, C. Y. Zhu, H. Ying et al., "A fuzzy testing method of industrial control protocol based on reverse analysis," *Chinese Journal of Electric Power Information and Communication Technology*, vol. 17, no. 4, pp. 1–9, 2019.

[24] G. H. Zhang and X. M. Shi, "An automated fuzzy test scheme for application layer protocol," *Chinese Journal of Microelectronics and Computer*, vol. 35, no. 3, pp. 99–103, 2018.

[25] K. Li, P.-G. Yan, and Q.-Y. Cai, "Quantum computing and the security of public key cryptography," *Fundamental Research*, vol. 1, no. 1, pp. 85–87, 2021.