计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.190600073

# 基于最长公共子串挖掘的未知链路层协议帧切割算法

陈庆超[1]　王　韬[1]　冯文博[2]　尹世庄[1]
1 中国人民解放军陆军工程大学装备模拟训练中心　石家庄 050003
2 中国人民解放军陆军工程大学指挥控制工程学院　南京 210007
(cqc62808@163.com)

**摘　要**　在日益激烈的现代电子对抗领域中,侦听方截获的原始数据一般是比特流的形式,将比特流划分为数据帧是处理截获数据的首要任务。现有方法虽然可以准确地提取相关序列实现帧切分,但是当需要处理的数据量较大时,时间和空间的消耗量过大,并且实验过程中常常需要预先设定一些阈值。为此,文中提出了一种基于最长公共子串挖掘的未知链路层协议帧切割算法,该算法通过统计一定长度的比特流的最长公共子串,逐步精确前导码和帧起始定界符,从而实现帧切分。实验数据表明,该算法相较于基于频繁序列挖掘以实现帧切分的算法,相关候选序列数量呈指数级下降,最终使得候选序列唯一。该算法的时间复杂度为 $O(n)$,且只需单次扫描,充分说明该算法可以高效地实现帧切分。

**关键词**:最长公共子串;前导码;帧起始定界符;帧分割;未知链路层协议
**中图法分类号**　TP393

# Unknown Link Layer Protocol Frame Segmentation Algorithm Based on Longest Common Substrings Mining

CHEN Qing-chao[1], WANG Tao[1], FENG Wen-bo[2] and YIN Shi-zhuang[1]
1 Equipment Simulation Training Center, Army Engineering University of PLA, Shijiazhuang 050003, China
2 College of Command and Control Engineering, Army Engineering University of PLA, Nanjing 210007, China

**Abstract**　In the increasingly fierce field of modern electronic countermeasures, the original data intercepted by the listener is generally in the form of bit stream. Dividing the bit stream into data frames is the primary task to process the intercepted data. Although the existing methods can accurately extract the related sequence to achieve frame segmentation, when the amount of data to be processed is large, the consumption of time and space is unacceptable and some thresholds need to be set in advance. For this reason, an unknown link layer protocol frame segmentation algorithm based on the longest common substring mining is proposed in this paper. By counting the longest common substring of the bit stream of a certain length, the preamble and the frame start delimiter become iteratively accurate. Thus, the frame segmentation is realized. The experimental data show that compared with the algorithm based on frequent sequence mining to achieve frame segmentation, the number of candidate sequences of the proposed algorithm is reduced exponentially, and the final candidate sequence is unique. The time complexity of the proposed algorithm is $O(n)$, and only a single scan is required, which fully shows that the proposed algorithm can realize frame segmentation efficiently.

**Keywords**　Longest common substring, Preamble, Frame starting delimiter, Frame segmentation, Unknown link layer protocol

## 1　概述

在日益激烈的现代电子对抗领域中,侦听方截获的原始数据一般是比特流的形式,将比特流划分为数据帧是处理截获数据的首要任务,在此基础上才能对协议的语法、语义等进行分析[1-2]。在通信系统中,一般通过帧同步[3-4]的方式提取帧结构,但是在链路层协议未知的情况下,前导码和帧起始定界符都是未知的,如果侦听方需要提取帧结构,就需要通过数据挖掘[5-6]等方式获得发送方规定的前导码[7]和帧起始定界符。

目前国内外关于帧切分的研究较少。Jin[8]通过实验验证了 AC(Aho-Corasick)算法对于提取比特流短频繁序列的有效性,并提出了新的剪枝方法以适应长频繁序列挖掘。Bai等[9]提出基于偏三阶相关函数峰值特性的识别方法,对同步码各位上的序列进行偏三阶相关函数运算,然后利用 $m$-序列采样后的偏三阶相关函数峰值相似度高的特性来区分信息码和同步码,通过进一步运算得到同步码的长度和起始位,该方法可以用较少的数据识别帧同步码。Wang等[10]提出基于频繁统计和关联规则的未知链路层协议比特流切割算法,通过频繁统计和关联规则验证,识别并提取标志帧开始的特征序列和关联规则序列,并给出 $N$ 种最具可能性的切割方案。Xue等[11]提出了一种在数据挖掘的基础上利用有向图构建多重关联规则进行比特流切分的算法,通过频繁序列统计、关

到稿日期:2019-06-23　返修日期:2019-08-29　本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:国家重点研发计划(2017YFB0802900)
This work was supported by the National Key R&D Program of China(2017YFB0802900).
通信作者:王韬(a13592247640@foxmail.com)

联规则分析以及关联规则整合，识别比特流中标识帧起始的多重关联规则序列，该方法复杂度较低且输出唯一。Lei等[12]提出基于前导码挖掘的未知协议帧切割算法，该方法利用AC(Aho-Corasick)算法从目标比特流中发现候选序列，基于候选序列集合大小的变化特征对序列选取方法进行改进，提出了未知前导码长度的判定与挖掘方法，从而提取前导码序列，实现帧切分。但是，该方法前导码长度的判定存在一定局限性。Wu[13]利用改进的AC算法实现对长度为 $m$ 的模式串进行统计，同时采用关联规则对模式串进行拼接以发现比特流的同步序列，最后利用汉明距离检测同步码出现的位置，实现帧定位。Yu等[14]通过在数据流中随机加入一些探针来提取正确的前导码，从中寻找连续的短模式串，提取最频繁的重复串并对其过滤，但是该算法实验操作复杂，且依赖硬件平台。目前的研究多是基于AC算法[15-16]统计频繁序列，从而获得标识帧开始的特征字段，实现帧切分。但是这类方法是一种基于大量统计的方法，需要统计和存储所有可能的特征字段，这意味着这类方法具有较高的时间与空间复杂度。

针对现有研究存在的问题，本文提出了一种基于最长公共子串[1-20]挖掘的未知链路层协议帧切分算法，能够高效准确地提取未知链路层协议帧的前导码和帧起始定界符，从而实现帧切分。该方法首先将比特流数据切割成多个具有一定长度的二进制序列，取两个二进制序列获得其最长公共子串集合，将集合中的所有公共子串与第3个序列进行比对，对公共子串集合进行更新，依次类推，直至完成对所有序列的比对，获得的公共子串集合即为前导码和帧起始定界符前后组合的候选序列集合。

## 2 未知链路层协议帧的切割原理

最长公共子串挖掘的目的是切割链路层协议帧，本节首先对链路层协议帧的结构进行简单介绍。

### 2.1 链路层协议帧的结构特征

将数据链路层比特流拆分成帧主要有4种方法：字节计数法、字节填充的标志字节法、比特填充的标志比特法和物理层编码违禁法[21]。目前，多数协议采用相同的分界模式，即利用前导码实现帧同步，利用帧起始定界符标识帧的开始，如图1所示。

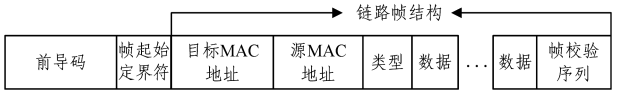



图1 Ethernet II 帧结构

Fig. 1 Frame structure of Ethernet II

### 2.2 相关定义和概念

**定义1** 若没有其他公共子串包含公共子串 $X$，则称 $X$ 为最长公共子串。

**定义2** 前导码序列和帧起始定界符序列的前后组合，称为帧同步序列。

在数据挖掘的过程中，可能挖掘到多个最长公共子串，这些最长公共子串均可视为帧同步序列的候选序列。

## 3 基于最长公共子串挖掘的未知链路层协议帧切割算法

本文提出的未知链路层协议帧切割算法主要分为4个步骤：1)将比特流数据进行分割，获得包含多个二进制序列的序列集合；2)求两个二进制序列的最长公共子串，组成初始最长公共子串集合，在序列集合中排除这两个序列；3)求最长公共子串集合中不被包含在第3个二进制序列中的子串与第3个二进制序列的最长公共子串，更新最长公共子串集合，在序列集合排除第3个序列；4)重复步骤3)，直至序列集合为空，所得最长公共子串集合中的序列即为帧同步序列的候选序列。算法流程如图2所示。

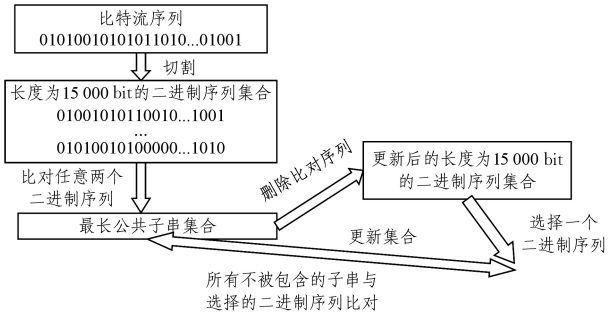



图2 本文算法的流程

Fig. 2 Flow chart of proposed algorithm

### 3.1 比特流数据分割为二进制序列集合

由于捕获数据时机的随机性，比特流数据的头部不一定从帧同步序列开始，为了确保二进制序列中包含帧同步序列，二进制序列的长度至少要大于一个帧的最大长度。常见的帧的最大传输单元约为1 500 Byte，即12 000 bit。将阈值设为15 000 bit对比特流数据进行分割。比特流数据切割后获得长度为15 000 bit的二进制序列的集合，切割效果如图3所示。

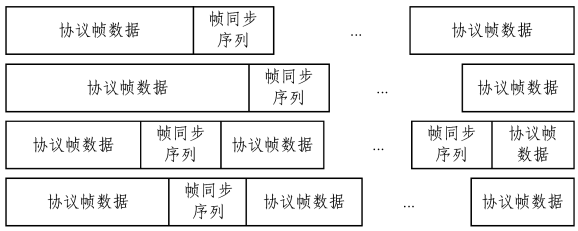



图3 二进制序列集合

Fig. 3 Binary sequence set

### 3.2 最长公共子串挖掘

设置足够大的阈值对比特流数据进行切割，可以保证帧同步序列被包含在二进制序列中，则两个二进制序列的公共子串一定包含帧同步序列。为了避免与协议用户数据冲突，帧同步序列的长度一般大于帧头的最长固定字段的长度，即MAC地址的长度(6 Byte)，因此可以只挖掘长度大于6 Byte的公共子串。两个二进制序列往往存在多个长度大于6 Byte的公共子串，剔除被包含的公共子串，剩下的公共子串组成最长公共子串集合，该集合中的元素包含帧同步序列。判断最长公共子串是否被包含在第3个二进制序列中，将不被包含的最长公共子串与第3个二进制序列进行比对，进一步缩小最长公共子串集合，更新最长公共子串集合。与其他二进制序列进行重复比对，更新最长公共子串集合，直至与所有长度为15 000 bit的二进制序列比对完全。根据局部性原理，最终的最长公共子串集合中的元素即为帧同步序列或者与帧同步序列没有区分度的序列。求两个二进制序列的最长公共子串集合的过程如算法1所示。

**算法 1** 最长公共子串集合算法

```
Input:Two binary sequence shortstr and longstr
Output:Longest common substrings set S
1. function FIND(shortstr, longstr)
2.     for i in range(shortstr.len)
3.       for j in range(longstr.len)
4.         n ← 1;substr ←""
5.         tempa ← shortstr[i];tempb ← longstr[j]
6.           while tempa==tempb
7.             substr← substr+tempa
8.               if i+n < shortstr.len and j+n < longstr.len
9.                 tempa←shortstr[i+n];
10.                  tempb← longstr[j+n];n ← n+1
11.              else
12.                break
13.            if substr.len > 48
14.              results.append(substr)
15.   S=results.sort()#将公共子串从大到小排序
16.   for i,item in enumerate(S)
17.     for smallitem in S[i+1:]
18.         if smallitem in item
19.           S.remove(smallitem)
20.   return S
21. end function
```

## 4 实验结果与分析

### 4.1 实验数据

实验所用数据来自某一主机在一段时间内捕获的同种协议的交互数据包，经过处理转换成的带帧头的连续比特流数据[11]，如表 1 所列。

表 1 实验数据集合

Table 1 Experiment dataset

| 序号 | 协议 | 包含帧数 | 比特流长度 |
|---|---|---|---|
| S1 | UDP | 1000 | 3101200 |
| S2 | UDP+TCP | 1000 | 3894736 |

### 4.2 帧同步序列挖掘

本文在配置为 2.60GHz 的 CPU、8GB 内存、操作系统为 Windows10 的 PC 上基于 Python 语言实现了所提方法。

实验 1 以 S1 数据集为研究对象，本文算法发现候选序列的个数及其变化规律如图 4 所示。

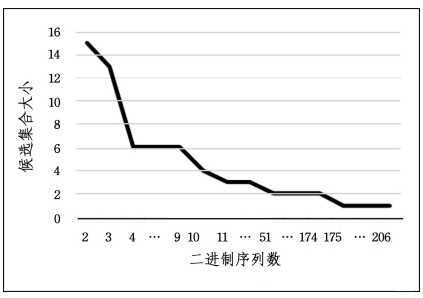



图 4 S1 数据集的候选序列集合大小随对比的二进制序列数的变化的特征

Fig. 4 Characteristics of size variation of candidate sequence set for S1 dataset with the number of binary sequences compared

由图 4 可知，当逐步进行比对时，候选序列集合会随之减小，而帧同步序列被逐步地精确提取，当算法运行结束时，候选序列集合减小至 1，挖掘出的候选序列为“10101010101010-10101010101010101010101010101010101010101011”，转换为 16 进制为“ AAAAAAAAAAAAAAAB”，其与已知以太网帧的帧同步序列相同。

实验 2 以 S2 数据集为研究对象，本文算法发现候选序列的个数及其变化规律如图 5 所示。

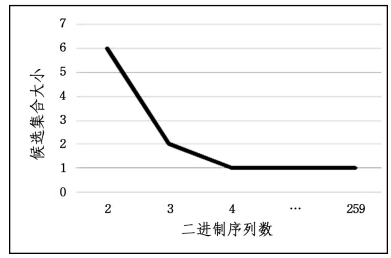



图 5 S2 数据集的候选序列集合大小随对比的二进制序列数变化的特征

Fig. 5 Characteristics of size variation of candidate sequence set for S2 dataset with the number of binary sequences compared

由图 5 可知，S2 数据集的候选序列集合在比对第 4 个二进制序列后就减小为 1，但是实验结果表明此时并未准确获得帧同步序列。获得的是包含帧同步序列的序列。这种现象持续到比对第 175 个二进制序列，当比对完第 176 个二进制序列后，实验获得了准确的序列“AAAAAAAAAAAA-AAAB”。同时，从图 5 中可以得出以下结论：数据之间较大的差异性使得 S2 数据集的候选序列集合变化梯度相较于差异较小的数据集 S1 的下降得更快。

### 4.3 时间、空间复杂度的对比

目前，国内外关于帧切分最前沿的研究包括基于前导码挖掘的未知协议帧切分算法[11]。在使用相同的数据下，将本文算法与该方法进行对比。1)时间复杂度对比，文献[11]需要构建挖掘不同长度频繁序列的字典树即多步骤运行，时间复杂度为 $O(kN)$，其中 $k$ 为需要挖掘的频繁序列长度的种类，且需要人为分析判断频繁序列的频数和长度之间的关系，以判定前导码的长度，存在多余的运算；而本文方法不需要构建字典树，整体流程为重复的两个序列之间的对比，时间复杂度为 $O(L^2+L+\cdots+L)$，即约为 $O(N)$，其中 $L$ 为切割的二进制序列的长度，第一步初始化候选序列集合的时间复杂度为 $O(L^2)$，之后对比的时间复杂度为 $O(L)$，且不需要人为选择判定。2)空间复杂度对比，文献[11]需要构建多个挖掘不同长度的频繁序列的字典树，几乎每次构建字典树都需要对二进制数据中的所有数据进行存储，空间复杂度为 $O(kn)$，其中 $k$ 为需要挖掘的频繁序列长度的种类，$n$ 为二进制序列的长度；而本文方法只需要存储候选的帧同步序列，由上述实验可知，帧同步序列集合逐步减小，空间复杂度为 $O(1)$。3)候选序列集合大小对比，文献[11]在经过两次改进后，候选序列个数控制在 1000 以内，而使用相同数据时，本文方法在第一次求初始候选帧同步序列集合时，就将候选序列个数控制在 20 以内，因此本文方法更具有针对性。综上，相较于文献[11]的基于前导码挖掘的未知链路帧切分算法，本文方法在算法的时间、空间复杂度和初始候选序列集合大小上都具有一定的优势。

本文方法通过逐一比对二进制序列，逐步精确帧同步序列，大大缩小了候选序列集合。与现有的基于 AC 算法等通过挖掘频繁序列提取相关序列的算法相比，本文算法具有以下优点。1)更具有针对性。挖掘频繁序列需要先进行大量统计获得候选序列集合，然后根据相关规则提取候选序列，广撒网，细挑选；而本文的方法在一开始就极大地缩小了候选序列集合，在运行时又逐步精确帧同步序列。2)准确率更高。本文算法不需要设置复杂的参数，减少了人为干涉，候选序列集合小，实验容易控制。3)复杂度低。本文算法不需要大量的空间存储候选序列，且可以全自动运行，不需要大量人为分析。

然而，本文算法仍有一定的局限性，即没有考虑噪声的干

**结束语** 本文针对现有的链路层协议帧切割方法针对性不强、实现复杂、时间和空间复杂度高等问题，提出了高效、准确且容易实现的基于最长公共子串挖掘的未知链路层协议帧切割算法，并通过实验验证了该算法能高效、准确地提取帧同步序列，实现链路层协议帧切割。下一步的研究方向是降低求两个二进制序列的最长公共子串集合的复杂度，进一步完善算法。

## 参 考 文 献


[1] WALTZ E. Information Warfare: Principle and Operations[M]. Beijing: Publishing House of Electronics Industry, 2003: 1-20.

[2] FEN L, TONG L, CHUN-RUI Z, et al. Length Identification of Unknown Data Frame[C]// Computational Intelligence and Security (CIS), 2012 Eighth International Conference on. IEEE, 2012.

[3] WU Y C, XU T H, WANG L M. Design of frame synchronization circuit[J]. Modern Electronic Technique, 2003, 26(4): 69-71, 73.

[4] YU P D, PENG H, GONG K X, et al. Fast Blind Recognition of Convolutional Interleavers Based on Existence of Frame Sync Codes[J]. Acta Electronica Sinica, 2018, 46(6): 1530-1536.

[5] LIU H Y, CHEN J, CHEN G Q. Review of Classification Algorithms for Data Mining[J]. Journal of Tsinghua University (Science and Technology), 2002, 42(6): 727-730.

[6] SUN L, ZHU Y Q. Study of Key Techniques in Mining Frequent Sequential Patterns[J]. Computer Engineering, 2006, 32(11): 95-96.

[7] VURAL S, WANG N, BUCKNELL P, et al. Dynamic preamble subset allocation for RAN slicing in 5G networks[J]. IEEE Access, 2018, 6: 13015-13032.

[8] JIN L. Study on bit stream oriented unknown frame head identification[D]. Shanghai: Shanghai Jiaotong University, 2011: 29-39.

[9] BAI Y, YANG X J, ZHANG Y. A Recognition Method of m-sequence Synchronization Codes Using Higher-order Statistical Processing[J]. Journal of Electronics and Information Technology, 2012, 34(1): 33-37.

[10] WANG H Z H, XUE K P, HONG P L, et al. An unknown link protocol bit stream segmentation algorithm based on frequent statistics and association rules[J]. Journal of University of Science and Technology of China, 2013, 43(7): 554-560.

[11] XUE K P, LIU B, WANG J S, et al. Data Link Bit Stream Oriented Association Analysis on Unknown Frame[J]. Journal of Electronics & Information Technology, 2017, 39(2): 374-380.

[12] LEI D, WANG T, WANG X H, et al. Unknown protocol frame segmentation algorithm based on preamble mining[J]. Journal of Computer Applications, 2017, 37(2): 440-444, 449.

[13] WU Y M. The Frame Location and Protocol Feature Analysis From The Bit-Stream in The Wireless Network[D]. Chengdu: University of Electronic Science and Technology of China, 2014.

[14] YU T, WANG S, YU X. A Preamble Mining Algorithm Oriented to Binary Protocol Using Random Probes[C]// International Conference on Intelligent Information Hiding and Multimedia Signal Processing. Springer, Cham, 2017: 318-326.

[15] MOHAMMED Y, MALIK K, NUR A, et al. Mobile forensic images and videos signature pattern matching using M-aho-corasick [J]. International Journal of Advanced Computer Science and Applications, 2016, 7(7): 261-264.

[16] QIAO Z, GOTO K, OHSHIMA T, et al. Dictionary matching: review of the aho-corasick algorithm and vision for large dictionaries[C]// Proceedings of the 8th International Conference on Information Systems and Technologies. ACM, 2018: 4.

[17] HOOSHMAND S, TAVAKOLI N, ABEDIN P, et al. On Computing Average Common Substring Over Run Length Encoded Sequences[J]. Fundamenta Informaticae, 2018, 163(3): 267-273.

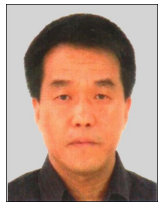
[18] THANKACHAN S V, APOSTOLICO A, ALURU S. A provably efficient algorithm for the k-mismatch average common substring problem[J]. Journal of Computational Biology, 2016, 23(6): 472-482.

[19] BARROS V, LIAO L, ROUSSEAU J. On the shortest distance between orbits and the longest common substring problem[J]. Advances in Mathematics, 2019, 344: 311-339.

[20] KOCIUMAKA T, RADOSZEWSKI J, STARIKOVSKAYA T. Correction to: Longest Common Substring with Approximately k Mismatches[J]. Algorithmica, 2019, 81(7): 3074.

[21] TANENBAUM A S, WETHERALL D J. Computer Networks: International Edition[M]. Pearson Schweiz Ag, 2010: 153-154.


**CHEN Qing-chao**, born in 1996, postgraduate. His main research interests include cyber security and so on.



**WANG Tao**, born in 1964, Ph.D, professor. His main research interests include cyber security and cryptography.