Chinese Journal of Electronics
vol. 33, no. 3, pp. 694–707, May 2024
https://doi.org/10.23919/cje.2022.00.366

RESEARCH ARTICLE

IP-Pealing: A Robust Network Flow Watermarking Method Based on IP Packet Sequence

Wangxin FENG[1,2], Xiangyang LUO[1,2], Tengyao LI[1,2], and Chunfang YANG[1,2]

1. *State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China*
2. *Key Laboratory of Cyberspace Situation Awareness of Henan Province, Zhengzhou 450001, China*

Corresponding author: Xiangyang LUO, Email: luoxy_ieu@sina.com
Manuscript Received October 31, 2022; Accepted February 14, 2023

Abstract — Network flow watermarking (NFW) is usually used for flow correlation. By actively modulating some features of the carrier traffic, NFW can establish the correspondence between different network nodes. In the face of strict demands of network traffic tracing, current watermarking methods cannot work efficiently due to the dependence on specific protocols, demand for large quantities of packets, weakness on resisting network channel interferences and so on. To this end, we propose a robust network flow watermarking method based on IP packet sequence, called as IP-Pealing. It is designed to utilize the packet sequence as watermark carrier with IP identification field which is insensitive to time jitter and suitable for all IP based traffic. To enhance the robustness against packet loss and packet reordering, the detection sequence set is constructed in terms of the variation range of packet sequence, correcting the possible errors caused by the network transmission. To improve the detection accuracy, the long watermark information is divided into several short sequences to embed in turn and assembled during detection. By a large number of experiments on the Internet, the overall detection rate and accuracy of IP-Pealing reach 99.91% and 99.42% respectively. In comparison with the classical network flow watermarking methods, such as PROFW, IBW, ICBW, WBIPD and SBTT, the accuracy of IP-Pealing is increased by 13.70% to 54.00%.
Keywords — Network flow watermarking, IP packet sequence, Detection sequence set, Fast detection, Robustness.
Citation — Wangxin FENG, Xiangyang LUO, Tengyao LI, *et al.*, "IP-Pealing: A Robust Network Flow Watermarking Method Based on IP Packet Sequence," *Chinese Journal of Electronics*, vol. 33, no. 3, pp. 694–707, 2024. doi: 10.23919/cje.2022.00.366.

I. Introduction

With the substantial increase on the scale of network traffic and the continuous improvement of attacker's capabilities, network attacks have shown new characteristics such as scale, automation, intelligence, and diversification. These new characteristics make network systems face with severe security threats. With anonymous networks or stepping stones, the attackers try to hide their traces, injecting attack instructions or steal sensitive files. In this circumstance, it is difficult for defenders to take defensive measures and trace the attack traffic effectively. As a typical means of active traffic analysis, network flow watermarking (NFW) is usually used for flow correlation, so as to help defenders establish the relationship between traffic located at different network nodes [1]. By consciously changing the flow characteristics at the source through the watermark embedder, NFW can embed watermarks into the traffic. The watermark can remain observable and effective despite the network noise during transmission [2]. The watermark detector deployed in a specific network node analysis the flow characteristics in order to extract watermarks from an observed flow and obtain the hidden information.

An application scenario of NFW in anonymous networks is shown in Figure 1. In order to find servers that provide illegal anonymous hidden services, the tracker can embed watermarks into the traffic sent by a hidden service client and detect watermarks through several watermark detectors deployed in different network nodes. If the similarity between the extracted watermark and the original watermark exceeds the threshold, it can be determined that the server interacting with the current client is an illegal hidden server. In comparison with pas-

Associate Editor: Prof. Jiwei HUANG, China University of Petroleum.

sive traffic analysis means, NFW has advantages of smaller demand of packet quantities, higher accuracy, lower detection time and so on [3]. It is of great significance to carry out the research on NFW for improving the defense capability of network system.

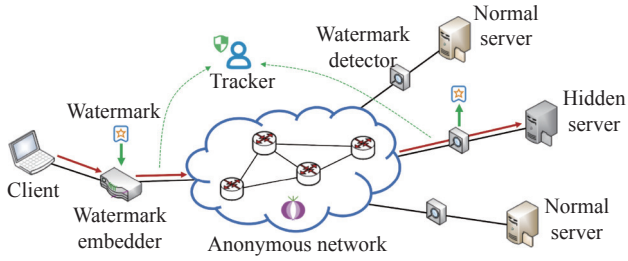


Figure 1 Application scenario of network flow watermarking.

According to the difference of watermark carriers, the existing NFW methods can be categorized into four types, including packet payload based, packet timing based, packet rate based and packet sequence based. Packet payload based NFW methods embed watermarks by modifying the packet payload. Wang *et al.* [4] proposed sleepy watermark tracing (SWT) which embeds watermarks by adding special characters to the payload of Telnet or Rlogin protocol. SWT has strong robustness, but it can only be applied to specific application layer protocols, which results in its limited scope of application.

Packet timing based NFW methods embed watermarks by modulating the feature of packet timing. Wang *et al.* [5] proposed a method called watermark based on inter-packet delay (WBIPD), which embeds watermarks by manipulating the inter-packet delays (IPDs) of several packets in the carrier traffic. Pyun *et al.* [6] proposed interval based watermarking (IBW) to complete watermark embedding by controlling the packet count in specific time intervals. The interval centroid based watermarking (ICBW) proposed by Wang *et al.* [7] embeds watermarks by modulating the IPDs in the time interval to make its interval centroid shift relatively. Using the operating mechanism of TCP and simulating the natural packet loss phenomenon of the network, Iacovazzi *et al.* proposed DROPWAT [8] and INFLOW [9], which embed watermarks by actively dropping some packets to make the carrier traffic generate some specific silent intervals. Based on the work of INFLOW, the ON/OFF flow watermarking technique proposed by Yang *et al.* [10] drops all packets in some specific time intervals to set up an IPD sequence corresponding to watermark bits. Yao *et al.* [11] proposed hidden Markov state based flow watermarking (HMSFW) which divides IPDs into Markov states and modulates the mean value of state transition probability in a specific time interval by analyzing the features of historical traffic. Tao *et al.* [12] studied the watermarking scheme of IPv6 environment and proposed mixed interval based watermarking (MIBW), which combines time interval centroid and time interval as the watermark carrier. Packet timing based NFW methods can achieve a good balance between invisibility and robustness. However, the existing methods often require large quantities of packets to complete watermark detection, which is not suitable for short traffic. The time jitter interference widely existing in the network channel also has a great impact on the effect of such methods.

Packet rate based NFW methods embed watermarks by controlling the packet rate. Yu *et al.* [13] embedded watermarks based on direct sequence spread spectrum (DSSS) by modulating the packet rate with a certain change rule in a specific time window. The SND based traceback technique (SBTT) proposed by Ling *et al.* [14] controls the packet rate of the sender by modifying the TCP notification window field returned by the receiver and then modulates the secret signal into the carrier traffic. Packet rate based NFW methods are easy to use. Unfortunately, the various interferences in the network channel can easily change the packet rate, which makes it difficult for the existing methods to recover the watermark correctly from the disturbed traffic. The existing methods also rely on the traffic with a long duration to complete watermark detection.

In order to resist the impact of time jitter, packet sequence based NFW methods have emerged, which embed watermarks by adjusting the sending sequence of packet. The typical method is packet reordering based flow watermarking (PROFW) proposed by Zhang *et al.* [15]. PROFW uses different permutations of several packets to represent the watermark and introduces error correction coding [16] to correct the packet reordering error of the watermark in network transmission. Since most of the packet sequence can be well maintained in network transmission [17], the robustness of packet sequence as a watermark carrier will be relatively better than packet timing and packet rate. Packet sequence based methods can complete watermark detection with only a few packets, which is suitable for short traffic. However, PROFW does not consider the impact of packet loss when designing the error correction mechanism. In face of packet loss interference, it is often unable to effectively recover the watermark information, which lacks of robustness. Moreover, PROFW identifies the packet sequence based on the sequence field in TCP header, which can only be applied to TCP traffic.

In this paper we propose IP-Pealing, a robust network flow watermarking method based on IP packet sequence. It is expected to improve the applicable protocol range of NFW method and can further improve the robustness of the watermark while using fewer packets to complete the watermark detection.

Our major contributions are as follows:

- We propose using IP packet sequence as watermark carrier and propose an algorithm of judging packet sequence based on IP identification field (IP ID). The packet sequence is identified by comparing IP ID values be-

tween packets, which is applicable to all IP based traffic.

- We design a watermark coding algorithm based on detection sequence set (DSS). The DSS is constructed based on the variation range of the packet sequence to design the watermark coding, which can correct the network channel transmission error and improve the robustness of watermark against packet loss and packet reordering.
- We propose embedding and detecting watermarks based on short sequences. The long watermark information is split into several short sequences and then embedded in turn. The watermark information is reconstructed during detection so as to avoid lots of packets out of order in a short time, which achieves higher detection accuracy for a long watermark information while enhancing the invisibility of watermarks.

The rest of this paper is organized as follows. Section II presents related works. Section III presents the details of IP-Pealing. Section IV performs theoretical analysis on the robustness of IP-Pealing. Section V provides experimental results of the performance of IP-Pealing. Section VI concludes this paper.

II. Related Works

In this section, we describe the basic principle of PROFW and analyze its possible problems. PROFW encodes the watermark information into different permutations of several packets. To correct the transmission error caused by packet reordering interference, PROFW measures the distance between the codewords by a Hamming distance, which corresponding to the displacement of two adjacent packets. The watermark coding schemes with different error correction capabilities are obtained by setting the valid codeword and the invalid codeword, as shown in Figure 2 (this codeword has the ability to correct the reordering error of two adjacent packets). To keep the packet reordering within a certain range, PROFW introduces probability modulation [18], [19] and embeds the watermark into the carrier traffic at a certain embedding ratio.

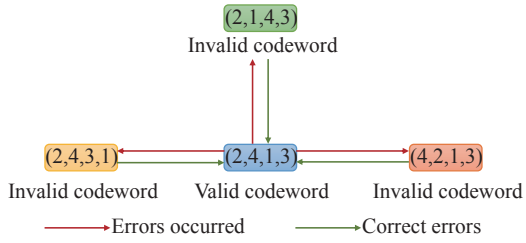


Figure 2 Error correction coding of PROFW.

The basic principle of PROFW is shown in Figure 3. In watermark embedding, the original watermark is first encoded into different permutations of several packets. Then, several packets are selected from the carrier traffic according to the embedding ratio. After that, the sequence of these packets is adjusted according to the watermark coding scheme. In watermark detection, the arrival sequence of packets is first analyzed in the observed traffic. Then, the packet sequence representing the watermark information is decoded into recovered watermark. Finally, if the recovered watermark and the original watermark pass the similarity judgment, it can be inferred that the current observed traffic is related to the traffic marked by watermark embedder.

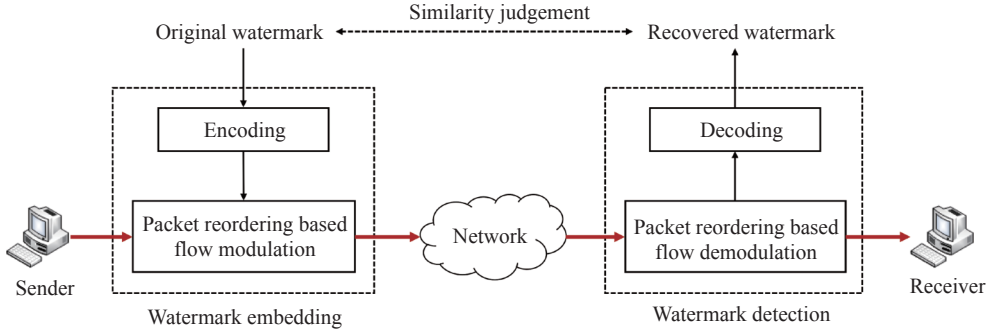


Figure 3 Basic principle of PROFW.

PROFW uses the sequence field in TCP header to identify the packet sequence. For example, given a group of four TCP packets $\langle P_1, P_2, P_3, P_4\rangle$ which follows a normal sequence $(1, 2, 3, 4)$, we have

$$\begin{cases} \mathrm{Seq}(P_1) = \mathrm{Seq}(P_1) \\ \mathrm{Seq}(P_2) = \mathrm{Seq}(P_1) + \mathrm{Len}(P_1) \\ \mathrm{Seq}(P_3) = \mathrm{Seq}(P_2) + \mathrm{Len}(P_2) \\ \mathrm{Seq}(P_4) = \mathrm{Seq}(P_3) + \mathrm{Len}(P_3) \end{cases} \tag{1}$$

where $\mathrm{Seq}(\cdot)$ indicates the value of sequence field of a TCP packet. $\mathrm{Len}(\cdot)$ indicates the payload length of a TCP packet. Given another group of four TCP packets $\langle P'_1, P'_2, P'_3, P'_4\rangle$ which follows a reordering sequence $(2, 4, 1, 3)$, we have

$$\begin{cases} \mathrm{Seq}(P'_1) = \mathrm{Seq}(P'_3) + \mathrm{Len}(P'_3) \\ \mathrm{Seq}(P'_2) = \mathrm{Seq}(P'_4) + \mathrm{Len}(P'_4) \\ \mathrm{Seq}(P'_3) = \mathrm{Seq}(P'_3) \\ \mathrm{Seq}(P'_4) = \mathrm{Seq}(P'_1) + \mathrm{Len}(P'_1) \end{cases} \tag{2}$$

By analyzing the relationship between the values of sequence field of several TCP packets, PROFW can detect whether there is a packet sequence representing a

watermark in TCP traffic.

PROFW shows strong robustness against delay jitter and packet reordering in the experimental environment set by the author. However, PROFW still has the following two shortcomings:

1) Dependent on specific protocols. Since PROFW uses the sequence field of TCP to identify the packet sequence, it is only valid for TCP traffic and cannot be applied to traffic based on other protocols, such as UDP traffic, because UDP does not have the field which can identify the packet sequence.

2) Insufficient robustness against packet loss interference. Packet loss may result in the disappearance of packet sequence information. When designing the error correction mechanism, PROFW only considers the error caused by packet reordering and does not consider the error caused by packet loss.

III. The Proposed Watermarking Method

In view of the shortcomings of PROFW, we propose IP-Pealing. The framework of IP-Pealing is illustrated in Figure 4. In the process of watermark embedding, firstly, the watermark information with length M is split into several short sequences as the information bit with length m (m is determined by watermark coding scheme). To distinguish the different watermark information, a flag bit is added at the front. Then, according to the watermark coding scheme, the flag bit and information bit are encoded into the watermark represented by the corresponding packet sequence one by one. Finally, the watermark is embedded into the carrier traffic at the embedding ratio p in the order from flag bit to information bit. When detecting the watermark, the packet sequence in the traffic is first identified according to IP ID and the detection sequence (DS) (referring to Section III.2) will be constructed. Then, the packet sequence representing the watermark will be decoded into the corresponding flag bit or information bit if it is detected in the current traffic according to the relation between DS and DSS. The existence of watermark can be confirmed after detecting any flag bit or information bit. The watermark information is reconstructed when flag bit and all information bits are detected.

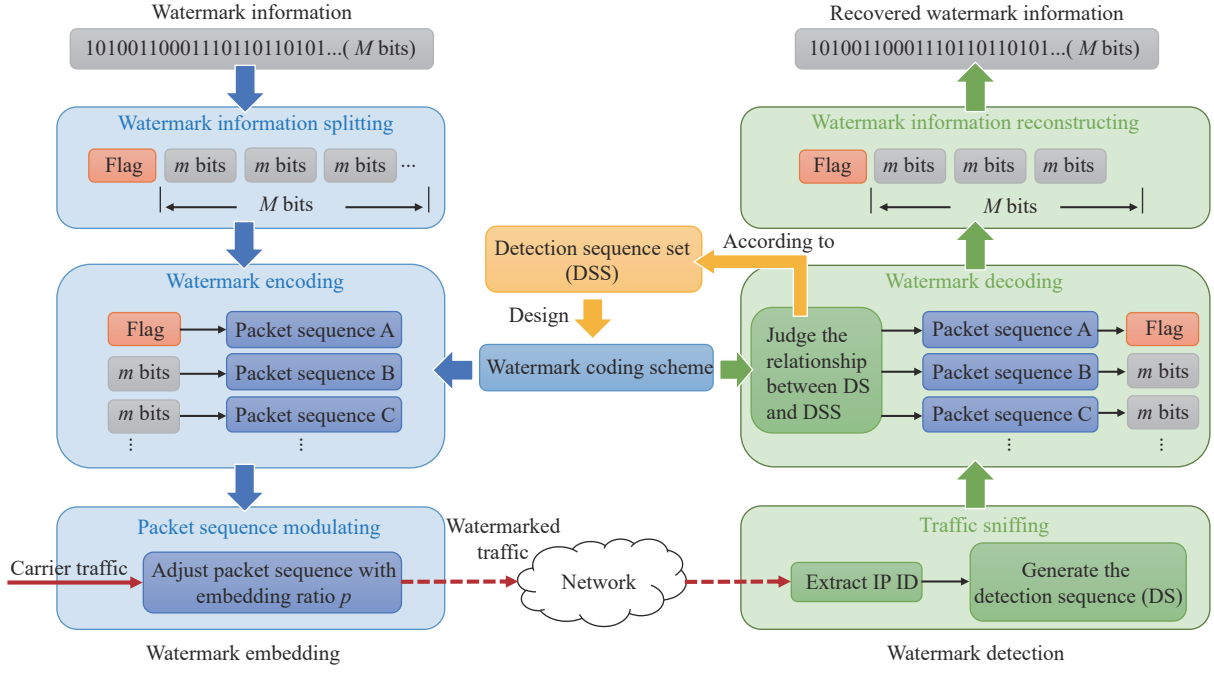


Figure 4 Framework of IP-Pealing.

In IP-Pealing, packet sequence identification, detection sequence set construction and watermark coding scheme design are the key parts. Next, we first elaborate the above three key parts and then introduce the specific process of watermark embedding and watermark detection.

1. Packet sequence identification

The IP header contains an identification field with a length of 16 bits. To avoid packet confusion, the sending host usually maintains a counter. Each time a packet is generated, the counter will add 1 and assign this value to the identification field. From the unidirectional traffic of a session, if there is no packet reordering occurs in the transmission, the IP IDs of the entire traffic in this direction will show an increasing order (when IP IDs are equal, it indicates that IP fragmentation has occurred, which is not within the scope of this paper). But if packet reordering occurs during transmission, the IP IDs of some packets in this direction will show a non-increasing order. Based on this phenomenon, the packet sequence

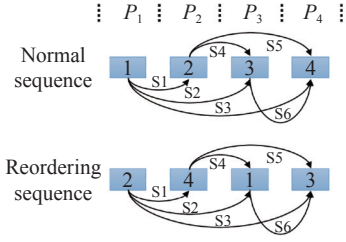
can be identified by comparing the IP ID value relation between packets. Given a queue of n packets $\langle P_1, P_2, \ldots, P_n\rangle$, the packet sequence identification function $f_{ps}(\langle P_1, P_2, \ldots, P_n\rangle)$ is

$$\mathrm{DEC}(b_{1,2}b_{1,3}\ldots b_{1,n}b_{2,3}b_{2,4}\ldots b_{2,n}\ldots b_{n-1,n}) \tag{3}$$

where $b_{1,2}b_{1,3}\ldots b_{1,n}b_{2,3}b_{2,4}\ldots b_{2,n}\ldots b_{n-1,n}$ is a binary sequence. The value of each bit $b_{i,j}$ is computed as

$$b_{i,j} = \begin{cases} 0, & \mathrm{IPID}(P_i) < \mathrm{IPID}(P_j) \\ 1, & \mathrm{IPID}(P_i) > \mathrm{IPID}(P_j) \end{cases} \tag{4}$$

where $1 \le i \le n-1$, $i+1 \le j \le n$. $\mathrm{IPID}(P_i)$ represents the IP ID value of P_i. $b_{i,j}$ is obtained by comparing the IP ID value of P_i and the IP ID values of its subsequent packets. DEC$(\cdot)$ is the function to transfer binary into decimal values. Let seq_n be the sequence formed by $\langle P_1, P_2, \ldots, P_n\rangle$. Since seq_n is a permutation with length n, each $f_{ps}(\langle P_1, P_2, \ldots, P_n\rangle)$ actually corresponds to an unique seq_n. Figure 5 shows an example of a packet sequence with $n = 4$. When $seq_4 = (1, 2, 3, 4)$, which indicates the normal packet sequence, we can see that each $b_{i,j}$ obtained from step 1 to step 6 is 0 and we have $f_{ps}(\langle P_1, P_2, \ldots, P_n\rangle) = 0$. But when $seq_4 = (2, 4, 1, 3)$, which indicates the reordering packet sequence, we can see that $b_{1,3}$, $b_{2,3}$ and $b_{2,4}$ obtained in step 2, step 4 and step 5 is 1 respectively and we have $f_{ps}(\langle P_1, P_2, \ldots, P_n\rangle) = 22$. Therefore, we can identify that seq_4 is $(1, 2, 3, 4)$ and $(2, 4, 1, 3)$ according to $f_{ps}(\langle P_1, P_2, \ldots, P_n\rangle)$ is 0 and 22, respectively.



Packet sequence	Step 1	2	3	4	5	6	f_{ps} result
	$b_{1,2}$	$b_{1,3}$	$b_{1,4}$	$b_{2,3}$	$b_{2,4}$	$b_{3,4}$	
(1,2,3,4)	0	0	0	0	0	0	0
(2,4,1,3)	0	1	0	1	1	0	22

Figure 5 Using IP ID to identify the packet sequence.

2. Detection sequence set construction

In order to correct the transmission errors, it is necessary to analyze the variation range of packet sequence caused by channel interferences. We propose to construct a DSS for watermark coding scheme design. As shown in Figure 6, the DS is composed of a set of f_{ps} results of a packet queue. The change of packet sequence will cause the corresponding change of DS. The variation range of DS will be expressed as DSS. Therefore, DSS can reflect the variation range of packet sequence under certain errors. Next, we introduce the construction methods of DS and DSS respectively.

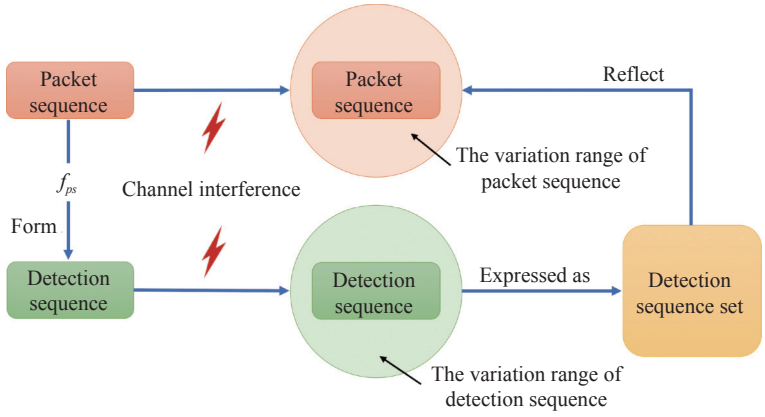


Figure 6 Relation between packet sequence, DS and DSS.

Given the length of a packet sequence n, a total of $(3n-2)$ packets are required to construct DS, including n packets with the sequence seq_n and $(2n-2)$ packets with normal sequence before and after the above n packets. DS is constructed through detection sequence generation (DSG) algorithm which is shown in Algorithm 1. When the sliding window W_s moving from L_1 to L_{2n-1}, the result of DSG algorithm is obtained by calculating the f_{ps} result of n packets in W_s one by one and forming these results into an ordered array. According to DSG algorithm, in the case of no packet reordering in $\langle P_1, P_2, \ldots, P_{3n-2}\rangle$, all elements in DS will be set as zero. But when these packets $\langle P_n, P_{n+1}, \ldots, P_{2n-1}\rangle$ are out of order, that is, the seq_n of these n packets is not $(1, 2, \ldots, n)$, then some non-zero elements will be obtained in DS. It can be inferred that there is a unique

correspondence between seq_n and DS.

Algorithm 1 Detection sequence generation

Input: n: packet sequence length; $\langle P_1, P_2, \ldots, P_{3n-2}\rangle$: packet queue.

Output: detection sequence DS.

1: set a sliding window W_s with length n and its initial position coincides with $\langle P_1, P_2, \ldots, P_n\rangle$;

2: let the position of the first packet inside W_s be the position of W_s, then the initial position of W_s can be recorded as L_1;

3: let $\langle P_1^w, P_2^w, \ldots, P_n^w\rangle_{L_i}$ represents the packet queue in W_s at L_i;

4: $DS \leftarrow [\ ]$ //DS is an ordered array;

5: for $1 \le i \le 2n-1$ do

6: DS.append($f_{ps}(\langle P_1^w, P_2^w, \ldots, P_n^w\rangle_{L_i})$);

7: end for

8: return DS.

Figure 7 illustrates an example of generating the DS of $(2,4,1,3)$. The sequence of $\langle P_4, P_5, P_6, P_7\rangle$ is $(2,4,1,3)$. When W_s moving from L_1 to L_7, we can obtained that the DS of $(2,4,1,3)$ is $[0,0,3,22,48,0,0]$ according to DSG algorithm. Conversely, we can identify that there is a packet sequence $(2,4,1,3)$ in $\langle P_1, P_2, \ldots, P_{10}\rangle$ according to the DS of $\langle P_1, P_2, \ldots, P_{10}\rangle$ is $[0,0,3,22,48,0,0]$.

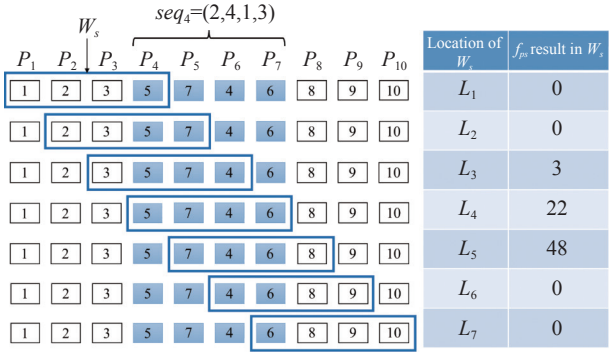


Figure 7 Construction method of DS.

To design the error correction coding scheme, it is necessary to analyze the possible changes of the packet sequence after the network transmission. The DSS is constructed based on the variation range of packet sequence, which reflects the possible DS changes of seq_n under certain errors. Each seq_n can correspond to a DSS. In this paper, we consider watermark deviation when errors result from a single packet loss or a single packet reordering occurs. Since the number of packets used by a single watermark is very small (usually just a few packets) comparing to the whole traffic, the probability of more than one packet loss and more complex packet reordering appeared within the watermark is relatively small. We make a statistic in the data set of April 2022 provided by WIDE MAWI [20] archive. For each flow in the data set, we randomly select n consecutive packets as a single simulated watermark which is already embedded into the flow. The results show that when n is taken as 4, 5 and 6, the single packet loss error accounts for 72.10%, 76.45% and 78.56% respectively among all packet loss errors occurred inside watermark, while the single packet reordering error accounts for 62.66%, 65.56% and 67.83% respectively among all packet reordering errors occurred inside watermark. Therefore, the single packet loss and single packet reordering we considered are sufficient to deal with most scenarios.

Firstly, we consider the case of packet loss error of the watermark. For the watermark representing by seq_n, when it loses the 1st to nth packet inside the watermark, respectively, n kinds of DSs will be generated, which represents all possible DSs of a watermark with a single packet loss error. As illustrated in Figure 8, $seq_4 = (2, 4, 1, 3)$ is put forward as an example. Secondly, we consider the case of packet reordering error in two adjacent packets. When the 1st and 2nd to the $(n-1)$th and nth packets inside the watermark representing by seq_n are reordering, respectively, $(n-1)$ kinds of DSs will be generated, which represents all possible DSs of the watermark under a single packet reordering error. Figure 9 illustrate an example of $seq_4 = (2, 4, 1, 3)$. And finally, the DSS is composed of all the above DSs as elements, which represents the possible variation range of DS resulting from a single packet loss or a single packet reordering.

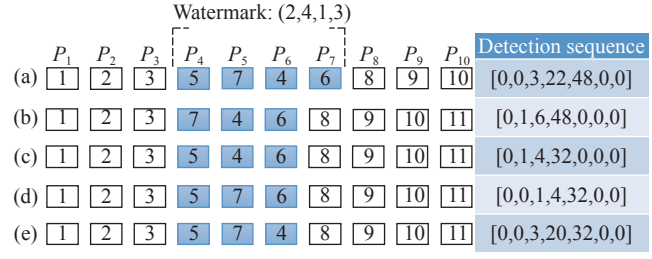


Figure 8 Variation range of DS in case of packet loss. (a) No packet loss; (b) Lost P_4; (c) Lost P_5; (d) Lost P_6; (e) Lost P_7.

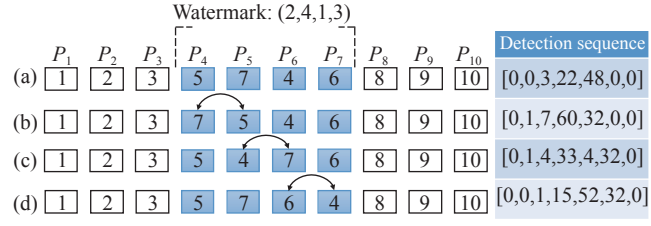


Figure 9 Variation range of DS in case of packet reordering. (a) No packet reordering; (b) P_4 and P_5 are reordered; (c) P_5 and P_6 are reordered; (d) P_6 and P_7 are reordered.

3. Watermark coding scheme design

Watermark coding is the core part of IP-Pealing. The original watermark information will be mapped into the watermark represented by the corresponding packet sequence after watermark coding. For the convenience of description, we regard the packet sequence as codeword. There are $n!$ types of codewords with length n, but to achieve error correction, only some of them can be used

for watermark coding. We know that each codeword can correspond to a DSS, which reflects the error range of the codeword. In a group of codewords, if the error range between any two codewords does not overlap, then this group of codewords has error correction ability according to error correction coding theory.

Table 1 lists the relevant concepts used in watermark coding. cw_{ul}, cw_{fp} and cw_{lr} will not be used for watermark coding for the following three reasons respectively: i) cw_{ul} indicates the normal packet sequence. ii) Since the distance between cw_{fp} and cw_{ul} is only 1 (the displacement of two adjacent packets corresponds to a Hamming distance [15]), it is easy to cause false alarm when cw_{fp} is used for watermark coding. iii) When losing a certain bit, cw_{lr} may lose reordering characteristic, resulting in the invalidation of watermarks.

Table 1 Relevant concepts used in watermark coding

Concept	Symbol	Description
Codeword	cw	Packet sequence
Hamming distance	$\mathrm{HD}(cw_1, cw_2)$	The distance between cw_1 and cw_2
Detection sequence set	DSS_{cw}	DSS corresponding to cw
Useless codeword	cw_{ul}	Codeword representing the normal packet sequence
False positive codeword	cw_{fp}	$\mathrm{HD}(cw_{fp}, cw_{ul}) = 1$
Low robustness codeword	cw_{lr}	Codeword that may lose reordering characteristic when losing a certain bit

Given the codeword length n, we propose the valid codeword groups generation (VCGG) algorithm to select all valid codewords to form codeword groups from $n!$ types of codewords, as shown in Algorithm 2. In VCGG algorithm, the initial set CW contains $n!$ types of codewords. The selection process of valid codeword groups is as follows:

1) Remove cw_{ul}, cw_{fp} and cw_{lr} in CW.

2) Remove cw in CW that conforms to the following constraints: $\mathrm{HD}(cw, cw_{fp}) < 2$. Since IP-Pealing considers codewords with a distance of 1 from the original codeword when correcting the packet reordering error, it is easy to cause false alarm if the distance between the original codeword and cw_{fp} is less than 2, so these codewords need to be removed.

3) The valid codewords are selected from the remaining codewords of CW according to the following principles: the currently selected codeword cw_i to be added to the valid codeword group and any codeword cw_j in current valid codeword group must conform to $DSS_{cw_i} \cap DSS_{cw_j} = \varnothing$. In the valid codeword group selected according to this principle, the DSS between any two codewords has no intersection, which means the error range of any two codewords does not overlap. Therefore, the transmission errors of these codewords can be accurately corrected when detecting the watermark, as shown in Figure 10.

Algorithm 2 Valid codeword groups generation

Input: n: codeword length.

Output: valid codeword group set $VCGS$.

```
the set CW is composed of n! types of codewords;
remove cw_ul, cw_fp and cw_lr in CW;
for cw in CW do
    if HD(cw, cw_fp) < 2 then
        remove cw;
    end if
end for
VCGS ← { };
for cw_i in CW do
    VCG_temp ← { };
    VCG_temp.append(cw_i);
    for cw_j, cw_j ≠ cw_i in CW do
        available ← True;   //Boolean variable
        for cw_t in VCG_temp do
            if DSS_cwj ∩ DSS_cwt ≠ ∅ then
                available ← False;
                break;
            end if
        end for
        if available then
            VCG_temp.append(cw_j);
        end if
    end for
    VCGS.append(cw_j);
end for
return VCGS.
```

The VCGG algorithm gives all valid codeword groups that meet the requirements of error correction coding under the condition of current watermark length. Let $VCG_{\max}$ be the valid codeword group with the maximum number of codewords in all valid codeword groups, calculate the maximum integer m satisfying with $2^m + 1 \leq \mathrm{card}(VCG_{\max})$, then m is the maximum number of bits that can be encoded with a single watermark under the condition of current watermark length. The watermark coding scheme can be selected as the valid codeword group containing $(2^m + 1)$ to $\mathrm{card}(VCG_{\max})$ codewords. According to VCGG algorithm, the valid codeword groups can be obtained only when $n \geq 5$.

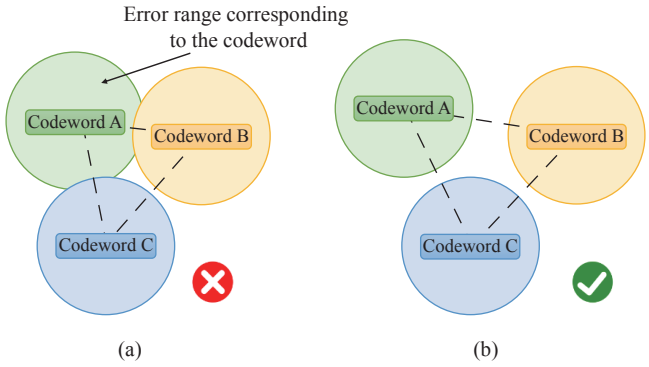


Figure 10 Intersection between the error range of codeword. (a) Since the error range between any two codewords overlaps, the error cannot be corrected accurately. (b) Since the error range between any two codewords does not overlap, the error can be corrected accurately.

When n is taken as 5, 6, and 7, m can be calculated as 2, 4 and 6 bits, respectively. Table 2 shows an example of a watermark coding scheme when $n = 6$.

Table 2 Watermark coding scheme ($n = 6$)

Original information	Codeword	Original information	Codeword
flag	(1,2,6,5,4,3)	1000	(3,4,5,1,6,2)
0000	(1,3,6,4,5,2)	1001	(3,4,6,2,1,5)
0001	(1,3,5,2,6,4)	1010	(4,2,5,3,6,1)
0010	(1,5,4,6,2,3)	1011	(4,3,6,5,2,1)
0011	(2,1,6,3,5,4)	1100	(4,5,1,2,3,6)
0100	(2,5,4,1,3,6)	1101	(5,3,2,4,1,6)
0101	(2,6,3,1,4,5)	1110	(5,6,3,4,1,2)
0110	(3,1,5,6,4,2)	1111	(6,1,4,2,3,5)
0111	(3,2,1,5,4,6)		

4. Watermark embedding

IP-Pealing's watermark embedding aims at adjusting the packet sequence in the carrier traffic according to the codeword, which can be summarized as the following three main steps:

1) Watermark information splitting. To enhance the invisibility of watermark and avoid lots of packets out of order in a short time, IP-Pealing splits the long watermark information into several short information bits and embeds each information bit as a single watermark. A flag bit will be added in front of each watermark information to distinguish them. The watermark information with length M will be split into M/m information bits and a flag bit.

2) Watermark encoding. The flag bit and all information bits will be mapped into the corresponding codewords respectively according to the watermark coding scheme.

3) Packet sequence modulating. IP-Pealing modulates the packet sequence of the carrier traffic through a packet buffer, as shown in Figure 11. When embedding a single watermark, the watermark embedder selects n consecutive packets from the carrier traffic according to the embedding ratio p and puts them into the buffer, then adjusts the sequence of packets in the buffer according to the codeword cw and sends them back into the traffic immediately. In this process, except the above n packets, the rest of packets in the traffic maintain the original sequence.

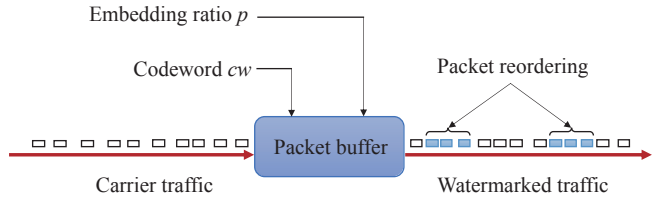


Figure 11 Packet sequence modulating.

5. Watermark detection

To determine the existence of watermarks and extract the watermark information, the watermark detector analyzes the arrival order of packets in the carrier traffic according to the watermark coding scheme. The watermark coding scheme and the watermark information length M need to be shared as a secret key between the watermark embedder and the watermark detector. The watermark detection can be summarized as the following three main steps:

1) Traffic sniffing. The traffic is divided into different sessions at the watermark detector. For each session, the watermark detector identifies the packet sequence based on IP ID from the unidirectional packet queue Q_r, as shown in Figure 12. An extraction window W_e with a length of $(3n-2)$ packets is used to identify the packet sequence. The initial position of W_e coincides with the first $(3n-2)$ packets in Q_r and moves backwards from the head of the packet queue. At each position, the DS is obtained in W_e through DSG algorithm.

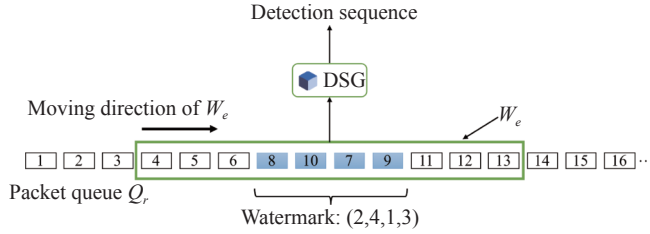


Figure 12 Traffic sniffing.

2) Watermark decoding. For each DS obtained in the previous step, in order to determine whether the current DS represents a certain original information, the watermark detector judges whether the current DS belongs to a DSS corresponding to a certain codeword according to watermark coding scheme and the correspondence between codeword and DSS, as shown in Figure 13. If so, decode it into the original information (flag bits or information bits) corresponding to DSS. Otherwise, the watermark is considered not detected in current DS.

3) Watermark information reconstructing. As illus-

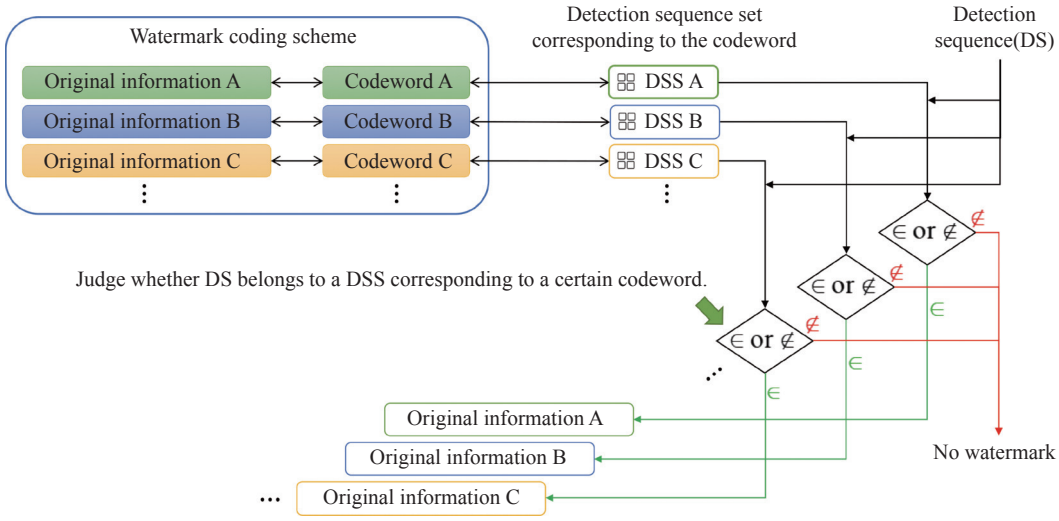


Figure 13 Watermark decoding.

trated in Figure 14, when the flag bit is found, the M/m information bits behind it are combined and the watermark information is recovered. When multiple watermark information are embedded into the carrier traffic, the flag bit can distinguish the different watermark information.

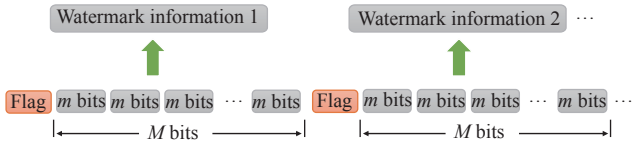


Figure 14 Watermark information reconstructing.

IV. Robustness Analysis

IP-Pealing improves the robustness of watermark by using DSS to correct the transmission errors. In order to verify the effectiveness of DSS in improving the robustness of watermark, we analyze the impact of packet loss and packer reordering on the detection rate of a single watermark under the condition of using DSS and without DSS, respectively.

Since the embedding of each watermark is independent, let us consider a watermarked traffic f_w embedded with a single watermark according to the embedding ratio p. Suppose that there are n consecutive packets representing the watermark and k packets with normal sequence in f_w (k increases with the decrease of p), then the total number of packets in f_w can be expressed as $(k+n)$.

1. Robustness against packet loss

First, we consider the influence of packet loss on the detection rate. To facilitate analysis, we assume that the packet loss follows the independent identically distributed (iid). Suppose that f_w loses l packets after network transmission, then the packet loss rate p_l can be expressed as $l/(k+n)$. Under the condition of using DSS, the watermark can be correctly detected when packet loss does not occur inside the watermark or 1 packet is lost inside the watermark, then the detection rate P_1 is given by

$$P_1 = \frac{\binom{k}{l} + \binom{k}{l-1}\binom{n}{1}}{\binom{k+n}{l}} \tag{5}$$

When DSS is not used for detection, the watermark can be detected only when packet loss does not occur inside the watermark, then the detection rate P_1' is given by

$$P_1' = \frac{\binom{k}{l}}{\binom{k+n}{l}} \tag{6}$$

Figure 15 shows the trend of P_1 and P_1' according to formulas (5) and (6) where we consider $n=7$ and three values of k (100, 50 and 25). As shown in Figure 15, P_1' decreases rapidly with the increase of p_l, but the decrease of P_1 is relatively slower due to the use of DSS.

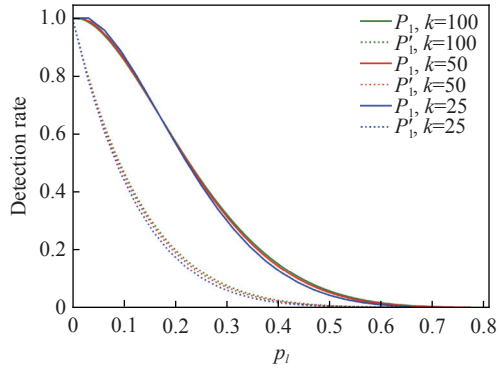


Figure 15 Watermark detection rate under varying packet loss rate.

2. Robustness against packet reordering

Then, we consider the influence of packet reordering on the detection rate. To facilitate analysis, we also assume that the packet reordering follows iid. Suppose r

packets in f_w are out of order after network transmission (according to RD [21] standard, the reordering degree of two adjacent packets is 1), then the packet reordering rate p_r can be expressed as $r/(k+n)$.

Under the condition of using DSS, the watermark can be correctly detected when packet reordering does not occur inside W_e or occurs inside the watermark, then the detection rate P_2 is given by

$$P_2 = \frac{\binom{k-2n+1}{r} + \binom{k-2n+1}{r-1}\binom{n-1}{1}}{\binom{k+n-1}{r}} \tag{7}$$

When DSS is not used for detection, the watermark can be detected only when packet reordering does not occur inside W_e, then the detection rate P_2' is given by

$$P_2' = \frac{\binom{k-2n+1}{r}}{\binom{k+n-1}{r}} \tag{8}$$

In Figure 16 we show the trend of P_2 and P_2' according to formulas (7) and (8) where we consider $n = 7$ and three values of k (100, 50 and 25). As depicted in Figure 16, the use of DSS significantly improves the detection rate of watermarks as well. The results show that the watermark coding scheme based on DSS can significantly improve the robustness of watermarks.

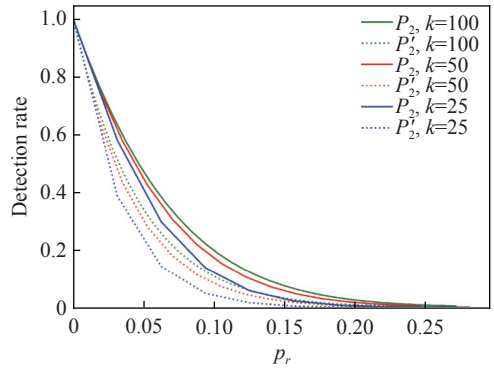


Figure 16 Watermark detection rate under varying packet reordering rate.

V. Experimental Results and Analysis

In order to verify the performance of IP-Pealing, an experimental environment is built on the Internet, as shown in Figure 17. The six nodes are deployed on virtual instances on elastic cloud service (ECS) and all traffic going from one instance to another is pass through the Internet. All instances are equipped with Ubuntu 18.04.5 LTS. The four instances are distributed in four countries. The sender and watermark embedder are deployed on an instance running in the first country. The sender uses the secure copy (SCP) command or Socket to generate traffic for testing. The watermark embedder uses the iptables of Linux Netfilter to intercept the traffic and modulate the packet sequence. The two stepping stones are deployed on two instances running in the second and third countries, which are used to forward traffic without storage. The watermark detector and receiver are deployed on an instance running in the fourth country, which are used to detect watermarks and receive the traffic respectively. The watermark parameters used for experiment are listed in Table 3.

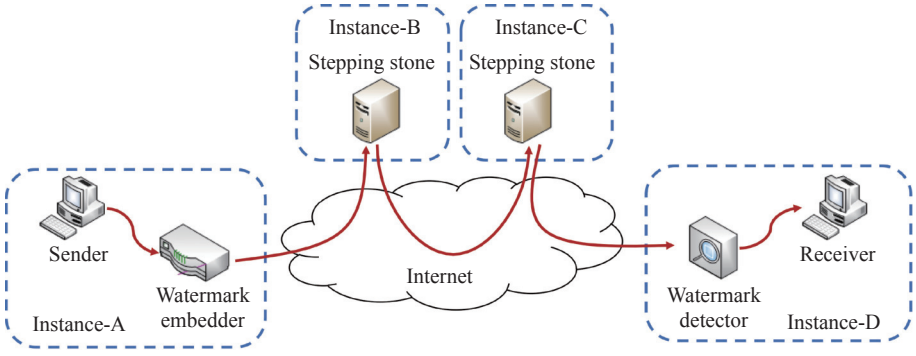


Figure 17 Experimental environment.

Table 3 Watermark parameters used for experiment

Parameter	Description	Value
M	Length of watermark information	12, 24, 36 bits
n	Length of codeword	5, 6, 7
p	Embedding ratio	0.01–0.05

The experiment is divided into the following four parts:

1) The detection rate (R_D) and accuracy (R_A) of the watermark are evaluated through 2250 tests. R_D refers to the probability that the watermarked traffic can be correctly identified. R_A refers to the ratio of the correct bits to all bits on the basis of correctly identifying the watermarked traffic.

2) The number of packets required in watermark detection is evaluated through 900 tests. We use the following two indicators: the number of packets required to detect the watermark (N_D) and the number of packets required to extract all watermark information completely (N_E). N_D refers to the number of packets required from the first packet of the traffic to the completion of water-

mark existence judgement. N_E refers to the number of packets required from the first packet of the traffic to the completion of all watermark information extraction.

3) The robustness of watermark under varying channel interferences is evaluated through 240 tests. The detection rate (R_D^s) and accuracy (R_A^s) of a single watermark are tested under the interferences of packet loss and packet reordering.

4) We compare IP-Pealing with IBW [6], ICBW [7], WBIPD [5], SBTT [12] and PROFW [13] through 1200 tests. N_E and R_A are selected as experimental indicators. The experiments are tested 600 times in the domestic Internet environment and the transnational Internet environment, respectively.

1. Detection rare and accuracy

In this part of the experiment, we use Socket to generate three rates of traffic ([25,50], [50,100] and [100,200] pkts/s) to test R_D and R_A while $M = 36$ bit. Table 4 lists the test results of R_D and R_A. Each result in the table is taken as the average of 50 tests. The results show that both R_D and R_A of IP-Pealing can maintain a high level in the traffic with different rates. The overall R_D and R_A is 99.91% and 99.42% respectively, which indicates the strong robustness of IP-Pealing on the Internet.

Table 4 Detection rate and accuracy (pkts/s: packets/s)

Rate (pkts/s)	Indicator	n	p				
			0.01	0.02	0.03	0.04	0.05
[25,50]	R_D	5	100%	100%	100%	100%	100%
		6	100%	100%	100%	100%	98%
		7	100%	100%	100%	100%	100%
	R_A	5	100%	100%	100%	100%	100%
		6	98%	100%	96%	100%	100%
		7	100%	100%	100%	100%	100%
[50,100]	R_D	5	100%	100%	100%	100%	100%
		6	100%	98%	100%	100%	100%
		7	100%	100%	100%	100%	100%
	R_A	5	100%	100%	98%	100%	100%
		6	100%	100%	100%	100%	100%
		7	100%	100%	98%	98%	100%
[100,200]	R_D	5	100%	100%	100%	100%	100%
		6	100%	100%	100%	100%	100%
		7	100%	100%	100%	100%	100%
	R_A	5	100%	98%	100%	100%	98%
		6	100%	100%	100%	98%	100%
		7	98%	98%	100%	98%	98%

2. Number of packets

In this part, we use SCP to generate SSH flows to test the variation of N_D and N_E under different parameters. Figure 18 shows N_D and N_E obtained in our experiment. Each point in the figure corresponds to the average of 20 tests. As shown in Figure 18(a), IP-Pealing only needs an average of 45 packets to complete the watermark existence detection and an average of 134 packets to completely extract all watermark information while $M = 12$ bits, $n = 7$ and $p = 0.05$. Even when $M = 36$ bits, as shown in Figure 18(c), IP-Pealing only needs 306 packets on average to extract all watermark information. The above results show that IP-Pealing only needs fewer packets to complete watermark detection, which can be applied to short traffic.

3. Robustness

On the basis of Internet channel interferences, we test the robustness of watermark under strong channel interferences by adding different intensity of packet loss and packet reordering at the stepping stone. We use Socket to generate traffic with a rate of [25, 50] pkts/s to test R_D^s and R_A^s under different packet loss and packet reordering rate. The experiment is carried out under the condition of using DSS and without DSS while $M = 6$ bits, $n = 7$, $p = 0.02$ are selected. Figure 19 shows the test result of R_D^s and R_A^s. Each point in the figure is taken as the average of 10 tests. Each test uses a traffic embedded with 50 kinds of watermarks. The results show that the use of DSS can significantly improve R_D^s, which conforms to the analysis results of formulas (5)–(8). As illustrated in Figure 19(a), we can see that packet loss affects both R_D^s and R_A^s, especially R_D^s without using DSS, but the use of DSS reduces this impact. Figure 19(b) illustrates that packet reordering mainly affects R_D^s, but the use of DSS also reduces this impact. The experi-

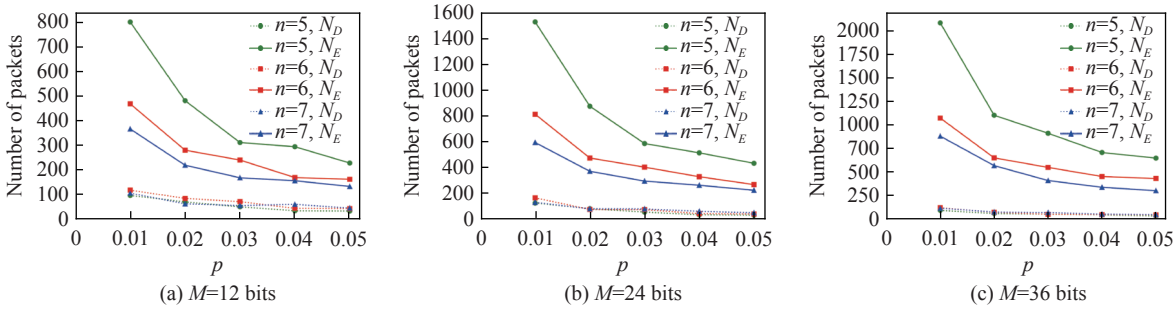


Figure 18 Number of packets required in watermark detection with different watermark parameter.

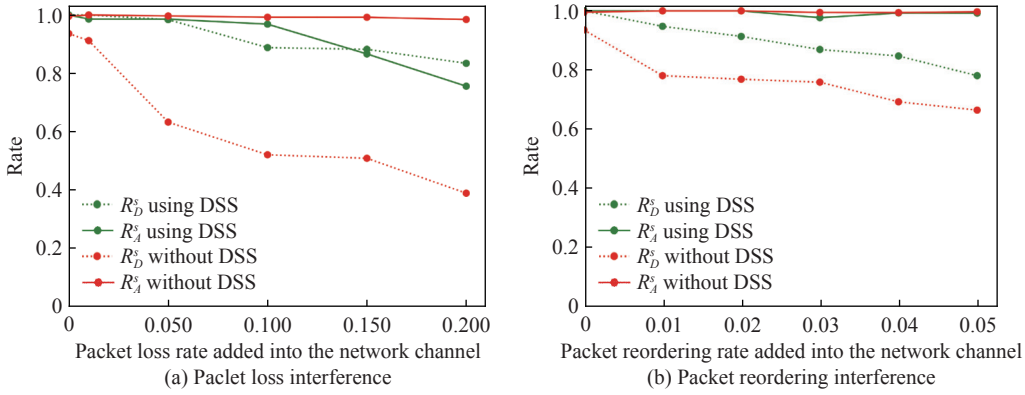


Figure 19 Detection rate and accuracy under varying channel interference rate.

ment results of this part verify the effectiveness of DSS in improving the robustness of watermark.

4. Comparative experiment

To validate the improvements of IP-Pealing, we compare it with some existing NFW methods. Our purpose is to select methods based on different watermark carriers as many as possible. Packet payload based methods are not selected due to its limited scope of application. Since we will carry out experiments in network channel with relatively strong packet loss interference, DROPWAT and INFLOW that construct watermarks by actively dropping some packets cannot work properly in that scenario. HMSFW that needs to use historical traffic characteristics is not consistent with our experimental scenario. Thus, we finally select the three most typical packet timing based methods: WBIPD, IBW and ICBW. For packet rate based methods, we select SBTT, which is the state-of-the-art method. Since IP-Pealing is improved on the basis of PROFW, we naturally select PROFW to compare with IP-Pealing, both of which are packet sequence based methods. The key parameters of each selected method are shown in Table 5. The traffic is generated by Socket. The length of watermark information is uniformly selected as 12 bits. We choose R_A and N_E as indicators to compare the performance of IP-Pealing and other five methods. The experiments are carried out in domestic and transnational Internet environment respectively. The domestic Internet environment is built using four instances located in the same country, while the four instances used in transnational Internet environment are distributed in four countries. The results of each group are taken as the average of 10 tests.

Table 5 Key parameters of each method

NFW method	Watermark carrier	Key parameters	Supported protocol
IP-Pealing	Packet sequence	Codeword length: $n = 7$; Embedding ratio: $p = 0.02$	TCP, UDP
PROFW [15]	Packet sequence	Codeword length: $n = 7$; Embedding ratio: $p = 0.02$	TCP
IBW [6]	Packet timing	Interval length: $T = 0.5$ s; Maximum delay: $d = 0.5$ s; Redundancy: $r = 3$; Offset: $o = 2$ s	TCP, UDP
ICBW [7]	Packet timing	Interval length: $T = 0.5$ s; Timing adjustment: $a = 0.35$ s; Redundancy: $r = 3$; Offset: $o = 2$ s	TCP, UDP
WBIPD [5]	Packet timing	Quantization step size: $s = 0.4$ s	TCP, UDP
SBTT [14]	Packet rate	Time interval: $T = 1$ s; TCP window size ratio: 3/4; Redundancy: $r = 3$; Offset: $o = 2$ s; Sliding window: $W_i = 0.01$ s; Step of W_i: $q = 1$	TCP

As shown in Figure 20(a) and Figure 21(a), the R_A of IP-Pealing is higher than that of other methods. It can be seen that the R_A of PROFW is also relatively higher than other methods based on packet timing and packet rate. But in comparation with the domestic Internet environment, the R_A of PROFW decrease significantly in transnational Internet environment with stronger channel interferences, while the R_A of IP-Pealing still remains at a high level. This is due to the fact that PROFW cannot handle packet loss errors, while IP-Pealing can correct packet loss errors by using DSS. Compared with IBW, ICBW, WBIPD, SBTT and PROFW, the overall R_A of IP-Pealing has increased by 46.54%, 52.00%, 54.00%, 48.46% and 13.70% respectively.

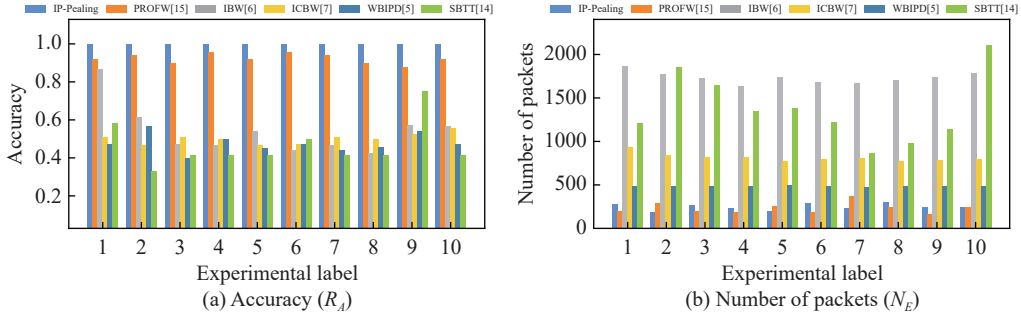


Figure 20 In comparison with other methods (domestic Internet environment).

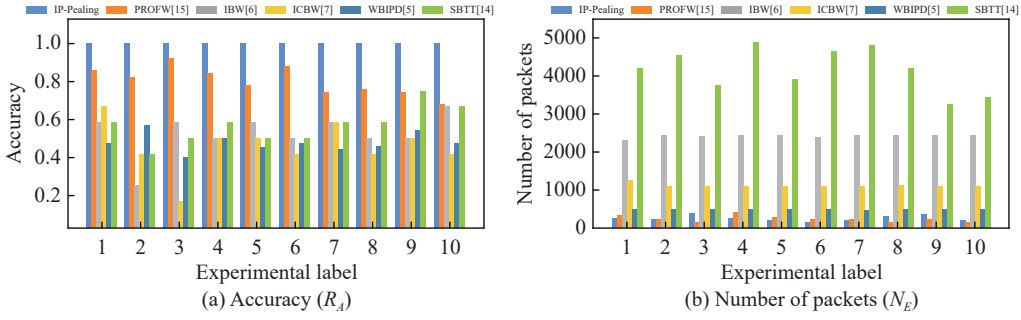


Figure 21 In comparison with other methods (transnational Internet environment).

Figure 20(b) and Figure 21(b) show that under the condition of embedding watermark information of the same length, the N_E of IP-Pealing and PROFW are less than that of other methods. In comparation with IBW, ICBW, WBIPD and SBTT, the overall N_E of IP-Pealing has decreased by 87.66%, 74.02%, 47.86% and 90.98% respectively. The above results indicate that IP-Pealing can achieve higher accuracy with fewer packets.

VI. Conclusion

To overcome the shortcomings of dependence with specific protocols, demand for large quantities of packets and weakness on resisting network channel interferences in existing NFW methods, we proposed IP-Pealing. First, IP-Pealing uses IP ID to identify the packet sequence, which is suitable for all IP traffic. Second, IP-Pealing constructs DSS based on the variation range of the packet sequence. The watermark coding scheme based on DSS achieves the error correction of watermark against network channel interferences, such as packet loss and packet reordering. Third, by splitting the long watermark information into several short sequences and reconstructing it during detection, IP-Pealing achieves high detection accuracy while enhancing the invisibility of watermarks. The experimental results based on the Internet show that the detection rate and accuracy of IP-Pealing have reached a high level. Even in face of strong channel interferences, IP-Pealing can still maintain strong robustness.

However, the performance of IP-Pealing may be affected in some special scenarios. For instance, if stepping stones use a store and forward method rather than forward without storage, which means the packet reordering is recovered at stepping stones, IP-Pealing cannot work properly in this case. In addition, some NAT (network address translation) mechanisms rewrite the IP ID or replace the entire IP header, the packets are not guaranteed to be forwarded in the original order at network layer, IP-Pealing may fall into invalidation due to the disappearance of packet reordering in this situation. In the future, the performance of IP-Pealing in anonymous networks will be put into the next researches.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (Grant No. 2022YFB3102904), the National Natural Science Foundation of China (Grant Nos. 62172435, U1804263, and

61872448), the Zhongyuan Science and Technology Innovation Leading Talent Project of China (Grant No. 214200510019), the Henan Province Key Research, Development and Promotion Project on Tackling Key Scientific Problems (Grant No. 222102210018), and the Key Research and Development Project of Henan Province (Grant No. 221111321200).

References

[1] H. Jin and C. Wang, "Progress in research on active network flow watermark," *Application Research of Computers*, vol. 37, no. 7, pp. 1925–1930, 2020.

[2] A. Iacovazzi and Y. Elovici, "Network flow watermarking: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 512–530, 2017.

[3] L. C. Zhang, Y. Z. Kong, Y. Guo, *et al.*, "Survey on network flow watermarking: Model, interferences, applications, technologies and security," *IET Communications*, vol. 12, no. 14, pp. 1639–1648, 2018.

[4] X. Y. Wang, D. S. Reeves, S. F. Wu, *et al.*, "Sleepy watermark tracing: An active network-based intrusion response framework," in *Proceedings of the IFIP TC11 16th Annual Working Conference on Information Security*, Paris, France, pp.369–384, 2001.

[5] X. Y. Wang, D. S. Reeves, and S. F. Wu, "Inter-packet delay based correlation for tracing encrypted connections through stepping stones," in *Proceedings of the 7th European Symposium on Research in Computer Security*, Zurich, Switzerland, pp.244–263, 2002.

[6] Y. J. Pyun, Y. H. Park, X. Wang, *et al.*, "Tracing traffic through intermediate hosts that repacketize flows," in *Proceedings of the 26th IEEE International Conference on Computer Communications*, Anchorage, AK, USA, pp.634–642, 2007.

[7] X. Y. Wang, S. P. Chen, and S. Jajodia, "Network flow watermarking attack on low-latency anonymous communication systems," in *Proceedings of the 28th IEEE Symposium on Security and Privacy*, Berkeley, CA, USA, pp.116–130, 2007.

[8] A. Iacovazzi, S. Sarda, D. Frassinelli, *et al.*, "DropWat: An invisible network flow watermark for data exfiltration traceback," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1139–1154, 2018.

[9] A. Iacovazzi, S. Sarda, and Y. Elovici, "Inflow: Inverse network flow watermarking for detecting hidden servers," in *Proceedings of the IEEE International Conference on Computer Communications*, Honolulu, HI, USA, pp.747–755, 2018.

[10] K. Yang, Z. H. Liu, Y. Zeng, *et al.*, "Sliding window based ON/OFF flow watermarking on Tor," *Computer Communications*, vol. 196, pp. 66–75, 2022.

[11] Z. J. Yao, L. Zhang, J. G. Ge, *et al.*, "An invisible flow watermarking for traffic tracking: A hidden Markov model approach," in *Proceedings of the IEEE International Conference on Communications*, Shanghai, China, pp.1–6, 2019.

[12] J. Tao, Z. C. Zhu, Z. Y. Wang, *et al.*, "A feature watermarking generation and embedding scheme for IPv6 network," *Journal of Computer Research and Development*, vol. 58, no. 11, pp. 2400–2415, 2021. (in Chinese)

[13] W. Yu, X. W. Fu, S. Graham, *et al.*, "DSSS-based flow marking technique for invisible traceback," in *Proceedings of the 2007 IEEE Symposium on Security and Privacyl*, Oakland, CA, USA, pp.18–32, 2007.

[14] Z. Ling, J. Z. Luo, D. N. Xu, *et al.*, "Novel and practical SDN-based traceback technique for malicious traffic over anonymous networks," in *Proceedings of the 38th IEEE Conference on Computer Communications*, Paris, France, pp.1180–1188, 2019.

[15] L. C. Zhang, Z. X. Wang, and J. Xu, "Flow watermarking scheme based on packet reordering," *Journal of Software*, vol. 22, no. S2, pp. 17–26, 2011.

[16] M. B. O. Medeni and E. M. Souidi, "A novel steganographic protocol from error-correcting codes," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, no. 4, pp. 337–343, 2010.

[17] J. C. R. Bennett, C. Partridge, and N. Shectman, "Packet reordering is not pathological network behavior," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 789–798, 1999.

[18] J. Fridrich and M. Goljan, "Digital image steganography using stochastic modulation," in *Proceedings of the SPIE 5020, Security and Watermarking of Multimedia Contents V*, Santa Clara, CA, USA, pp.191–201, 2003.

[19] J. H. He and J. W. Huang, "Steganalysis of stochastic modulation steganography," *Science in China Series F*, vol. 49, no. 3, pp. 273–285, 2006.

[20] WIDE MAWI Working Group, "MAWI Working Group Traffic Archive", Available at: https://mawi.wide.ad.jp/mawi/samplepoint-F/2022, 2022.

[21] T. Banka, A. A. Bare, and A. P. Jayasumana, "Metrics for degree of reordering in packet sequences," in *Proceedings of the 27th Annual IEEE Conference on Local Computer Networks*, Tampa, FL, USA, pp.333–342, 2002.

Wangxin FENG was born in 1994. He received the B.E. degree in information and communication engineering from National University of Defense Technology in 2016. He is now a M.S. candidate in State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, China. His current research interests focus on cyberspace security.
(Email: fwxws@163.com)

Xiangyang LUO was born in 1978. He received the B.E., M.S. and Ph.D. degrees from State Key Laboratory of Mathematical Engineering and Advanced Computing in 2001, 2004 and 2010 respectively. He is the author or co-author of more than 150 refereed international journal and conference papers. He is now a Professor in State Key Laboratory of Mathematical Engineering and Advanced Computing. His research interests include network and information security.
(Email: luoxy_ieu@sina.com)

Tengyao LI was born in 1991. He received the B.E. and M.S. degrees in computer science and technology in 2014 and 2016 respectively, the Ph.D. degree in cyberspace security in 2020 from Air Force Engineering University. He is currently a Lecturer in State Key Laboratory of Mathematical Engineering and Advanced Computing. His research interests include network active defense and traceability analysis.

Chunfang YANG was born in 1983. He received the B.E., M.S. and Ph.D. degrees from State Key Laboratory of Mathematical Engineering and Advanced Computing in 2005, 2008 and 2012 respectively. He is currently an Associate Professor in State Key Laboratory of Mathematical Engineering and Advanced Computing. His research interests include information hiding and analysis technology.