doi:10.11959/j.issn.1000-436x.2017196

基于平滑分组的分组间隔时间流水印方法

金华，朱慧，王昌达
（江苏大学计算机科学与通信工程学院，江苏 镇江 212013）

摘 要：针对现有基于时间信道的主动网络流水印技术缺少自纠错和难以抵御熵值隐蔽性嗅探的缺陷，提出一种基于平滑分组的分组间隔时间流水印方法。利用卷积码扩展水印信息，并采用平滑分组的方法将水印信息嵌入数据分组流中，通过交替调制数据分组间隔时间，使水印数据流的分组间隔分布特征无限逼近于正常的网络流，有效降低了数据分组在传输过程中遇到的时延抖动、分组丢失、分组合并、分组分片等因素干扰。理论分析和实验结果均表明，与现有的数据分组流水印技术相比，该方法具有检测准确度高、顽健性和隐蔽性好的特点。
关键词：流水印；平滑分组；间隔时间；卷积码
中图分类号：TP393.08 **文献标识码**：A

Packets flow watermarking method based on the inter-packet delay with smooth crossed grouping

JIN Hua, ZHU Hui, WANG Chang-da
(School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China)

Abstract: To improve the self-correction ability and resist the entropy-based detection, a flow watermarking approach based on the inter-packets delays with smooth crossed grouping was proposed. Such an approach extended the watermarking methods using both the convolutional code and the smooth group methods to embed the watermarks into packet flows. By adjusting the inter-packets delays of the crossed packets, the transmission time distribution of the watermarked packets can indefinitely approach to that of any normal packets transmission times. Furthermore, the approach can mitigate the negative consequences introduced by packets transmission jitters, packets losses, packets aggregations and packets divisions for the watermarks detection. Both theoretical analysis and experimental results show that the proposed approach overweight the known watermarking methods from the aspects of identification accuracy, robustness and hiddenness.
Key words: packets flow watermarking, smooth crossed grouping, inter-packet delay, convolutional code

1 引言

近年来，在经济利益的驱使下，出现了用户恶意使用匿名通信系统逃避合法网络监管或利用匿名通信系统作为保护伞对外发起网络攻击或泄露内网秘密的问题。因此，监管方需要具备发现和追踪用户违规匿名通信的能力，而基于数据分组间隔时间的主动流水印方法[1,2]是目前普遍采用的监控手段。

早期，Wang 等[3]采用流内分组间隔时间（IPD, inter-packet delay）随机分组实现水印的嵌入与追踪。这种技术不能将任意的水印位数据都嵌入特定大小的数据分组流中，也难以抵御时间扰乱和流合并等攻击方式。Pan 等[4]针对上述弱点，提出了改变数据分组发送延迟的分组技术，但该技术通过修改数据分组的时间戳实现，不仅会导致数据分组的结构发生变化，而且实时操作的难度大，水印提取方又没有校正机制，所以，此类方法的水印检出率相对较低。

收稿日期：2017-01-05；**修回日期**：2017-05-12
基金项目：国家自然科学基金资助项目（No.61672269）；江苏省科技成果转化基金资助项目（No.BA2015161）
Foundation Items: The National Natural Science Foundation of China (No.61672269), Jiangsu Province Technology Achievement Transformation Program (No.BA2015161)

Pyun 等[5]提出基于定长时间窗口的水印技术（IBW, interval-based watermarking），其主要思想是将水印嵌入方的时间坐标轴分为若干个定长的窗口，选择 2 个相邻窗口组成窗口对（windows pair），然后通过调节窗口对中含有的数据分组数目来实施水印信息的嵌入。该方法不易实现收发双方的时钟同步，也难以处理哑分组插入的干扰。针对这一问题，Wang 等[6,7]提出基于间隔质心的流水印（ICBW, interval centroid based watermarking）技术，基于均匀分布的原理，在 2 组时隙窗口内通过调整数据分组的时间偏移来嵌入水印信息。此类方法具有较好的顽健性，但由于人为改变时隙窗口的质心，导致水印数据分组传输的时间分布特征发生了变化，所以不能抵御以熵值分析为代表的统计攻击。

通过扩展上述水印技术，出现了基于 IBW 的扩频流水印方法[8]，以及基于 ICBW 的扩频流水印方法[9,10]。此类方法通过伪噪声码对水印信号进行扩频[11]，优点是可有效应对自相关类的水印攻击。但直序扩频技术不适合追踪低速数据流，对于过长的伪噪声码在数据分组流持续的时间内也不能完成追踪，同时，扩频技术还会引起网络吞吐量的变化和信号跨越频带区域等问题[12] 。

近期，Dengle 等 [13]提出采用伪随机函数选取数据分组，根据嵌入水印的比特，增加数据分组的 IPD，使其值成为一个固定步长的奇数倍或偶数倍，水印检测方根据测量的 IPD 提取水印的二进制编码。该方法隐蔽性较好，但由于分组间隔经过网络传输后可能产生较大的畸变，仅依赖固定步长的奇数倍或偶数倍提取水印，顽健性较弱。

本文提出一种基于平滑分组的分组间隔时间流水印技术，主要贡献总结如下。

1) 提出将含有水印的网络流中的分组间隔时间 IPD 划分为 2 种不同的类型：一类用于嵌入水印信息；另一类用于调制数据分组流传输的时间分布特征。通过交叉分组，2 种类型的数据分组在水印数据分组流中交替出现。

2) 在交叉分组的基础上，采用动态关联方式对数据分组的 IPD 进行处理，使含有水印的数据分组流传输时间特征无限逼近网络中正常传输的数据分组流。

3) 通过对相同网络条件中的水印数据分组流和正常数据分组流的修正条件熵（CCE, corrected conditional entropy）的分析比较，证明本文的流水印方法可抵御基于熵值的水印嗅探攻击。

2　水印嵌入及提取方案

基于平滑分组的流水印方案可以分为 4 个部分：平滑分组、水印嵌入、水印提取以及相似度计算。其工作流程如图 1 所示。

2.1　平滑分组

在通信系统中，当通信发送方经过网络链路发送 n 个数据分组 $p_1, p_2, \cdots, p_n$，处于链路上的水印嵌入方首先捕获数据分组的时间戳并加以记录，然后根据待发送水印信息的编码长度对捕获的数据分组进行分组长度为 $8a-1$（a 为正整数，可根据实际情况选取恰当的值，控制分组长度)。其中，IPD 被进一步分成 2 类，一类用于嵌入水印信息，另一类用于调节数据分组流传输的时间分布特征。2 类 IPD 的载体数据分组以交叉分组的方式嵌入，方法如下。

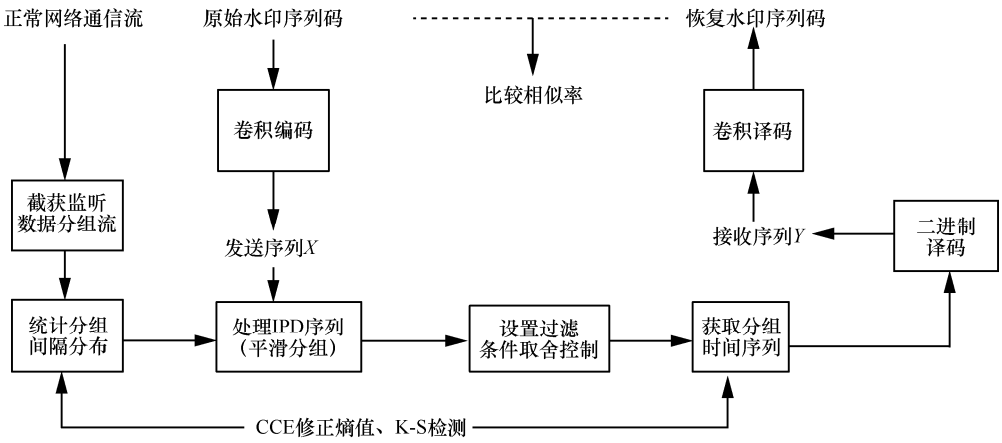

图 1　水印技术实现流程

设 l 为水印二进制序列的长度，q_k 为第 k 个分组中的数据分组，表示为

$$q_k=\{p_{k,1},\ p_{k,2},\ p_{k,3},\cdots,p_{k,8a-1}\},k=1,2,3,\cdots,l \tag{1}$$

若数据分组标号从 1 开始，则将 q_k 中序号为奇数的 IPD 作为水印的载体，而序号为偶数的 IPD 用于调节当前数据分组 IPD 的概率分布特征。

IPD 的载体——数据分组对 PPairs (packages pairs)表示为

$$<p_{k,i},p_{k,i+s}>,s=4a,\ i=1,3,\cdots,4a-1 \tag{2}$$

将 PPairs 交叉分为 A_k 、 B_k 两组，规则如下。

若 $\left\lfloor\frac{i}{2}\right\rfloor\%2=0$ ，则令 $<p_{k,i},p_{k,i+s}>\in A_k$ ；若 $\left\lfloor\frac{i}{2}\right\rfloor\%2!=0$ ，则令 $<p_{k,i},p_{k,i+s}>\in B_k$ 。

按上述方法分组后，A_k 和 B_k 各包含有 a 个 PPairs，即共使用 $2a$ 个 IPD 并将其交叉分为 2 组。若共有 $2n$ 个数据分组，则用于嵌入水印信息的数据分组和用于调节数据分组流传输时间分布特征的数据分组将各占 n 个。

若 p_i 的到达时间为 t_i ($i=1,2,3,\cdots,n$)且满足 $t_1<t_2<t_3<\cdots<t_n$ ，将 $p_{k,i}$ 和 $p_{k,i+s}$ 间的 IPD 记为 $ipd_{k,i}$ (k 表示第 k 个分组，i 表示第 k 组中第 i 个 IPD)，其值的大小可计算为

$$\begin{aligned}&ipd_{k,i}=t_{k,i+s}-t_{k,i},k=1,2,\cdots,L,\\&i=1,3,\cdots,4a-1,s=4a,\ a\in N^+\end{aligned} \tag{3}$$

设 $ipd_{k,1,u}$ 和 $ipd_{k,2,u}$ $(u=1,2,\cdots,a)$ 分别为 A_k 组、B_k 组中第 u 个 IPD 值，则第 k 组中 A_k 、 B_k 的 IPD 均值 $ipd_{k,1,\text{avg}}$ 和 $ipd_{k,2,\text{avg}}$ 可计算为

$$ipd_{k,j,\text{avg}}=\frac{1}{a}\sum_{u=1}^{a}ipd_{k,j,u},\ j=1,2,k=1,2,\cdots,L \tag{4}$$

所以，需要延迟的 IPD 数值 Δ_k 为

$$\Delta_k=\left|ipd_{k,1,\text{avg}}-ipd_{k,2,\text{avg}}\right|,k=1,2,\cdots,L \tag{5}$$

2.2 水印嵌入

将待发送长度为 l 的二进制水印码序列 w 通过编码效率为 $\frac{1}{2}$ 的卷积码进行扩展输出。本文统一采用约束长度 $N=3$ 的卷积码。其连接多项式为

$$\begin{cases}g_1=1+x+x^2\\g_2=1+x^2\end{cases} \tag{6}$$

生成多项式 $g_i(x)$ 对应子码输出序列 $y_{i,j}(x)$ 的通式为

$$y_{i,j}(x)=m(x)\,g_i(x),\ i=1,2,\ j=1,2,\cdots,\infty \tag{7}$$

其中，$m(x)$ 表示输入序列多项式。

若采用解析表示的编码方式，当输入序列为 1011，即 $m(x)=1+x^2+x^3$ ，根据式(6)和式(7)编码得到序列：111000010111，共 12 位；若采用图解表示的编码方式，则将 1011 根据码树图进行编码，可得序列：11100001，共 8 位。与解析表示相比，其输出位数较少，因此，本文采用图解表示方法进行基于卷积码的水印冗余扩展。

若采用 $N=3$ 、编码效率为 $\frac{1}{2}$ 的卷积码对水印信息 w 扩展后得到 X ，根据 2.1 节中的方法，当 X 的长度为 $2l$ 时，共需要使用 $16la-1$ 个 IPD。在使用平滑方法将数据分组后，根据式(4)求出分组数据分组的 IPD 均值，并将其保存至缓存区中。具体规则为：若需要嵌入的信息码 x_i 的信息为“1”时，则调整 A_k 、 B_k 分组中数据分组的发送时间，令 $ipd_{k,1,\text{avg}}$ 与 $ipd_{k,2,\text{avg}}$ 满足不等式 $ipd_{k,1,\text{avg}}\geqslant ipd_{k,2,\text{avg}}$ ；若 x_i 的值为“0”，则令 $ipd_{k,1,\text{avg}}<ipd_{k,2,\text{avg}}$ 成立。具体方法如下。

为满足 $ipd_{k,1,\text{avg}}\geqslant ipd_{k,2,\text{avg}}$ ，可增大 $<p_{k,\varsigma},p_{k,\varsigma+s}>$ 的 IPD 值，减小 $<p_{k,\varsigma+2},p_{k,\varsigma+s+2}>$ 的 IPD 值，其中，ς 为位于分组中前端的数据分组下标值，即在保持 $t_{k,\varsigma}$ 、 $t_{k,\varsigma+s+2}$ 不变的前提下，对 $t_{k,\varsigma+2}$ 、 $t_{k,\varsigma+s}$ 延迟 Δ_k 。同理，对于 $ipd_{k,1,\text{avg}}<ipd_{k,2,\text{avg}}$ 需保持 $t_{k,\varsigma+2}$ 、 $t_{k,\varsigma+s}$ 不变，对 $t_{k,\varsigma}$ 、 $t_{k,\varsigma+s+2}$ 延迟 Δ_k ，同时用于平滑传输时间分布特征的数据分组发送时刻值为

$$t_{2n}=\frac{t_{2n-1}+t_{2n+1}}{2},\ n\in N^+ \tag{8}$$

在计算出所有数据分组的发送时刻后，将缓冲区中的数据分组按对应的时刻传输，具体水印的平滑嵌入算法如算法 1 所示。

算法 1

输入 $a,k,s,x_1,x_2,\cdots,x_k,\ \ t_1,t_2,t_3,\cdots,t_{8ak-1},$
$ipd_{1,1,\text{avg}},ipd_{1,2,\text{avg}},ipd_{2,1,\text{avg}},ipd_{2,2,\text{avg}},\cdots,$
$ipd_{k,1,\text{avg}},ipd_{k,2,\text{avg}}$

输出 $ipd_{1,1,\text{avg}},ipd_{1,2,\text{avg}},ipd_{2,1,\text{avg}},ipd_{2,2,\text{avg}},\cdots,$
$ipd_{k,1,\text{avg}},ipd_{k,2,\text{avg}},t_1,t_2,t_3,\cdots,t_{8ak-1}$

1) for $k=0$ to $n-1$ //n代表水印总长度

2)　$\Delta_k = \left| ipd_{k,1,\text{avg}} - ipd_{k,2,\text{avg}} \right|$

3)　if $\Delta_k = 0$

4)　　$\Delta_k \leftarrow 0.5$

5)　end if

6)　if $x = 0$ and $ipd_{k,1,\text{avg}} > ipd_{k,2,\text{avg}}$ or $ipd_{k,1,\text{avg}} = ipd_{k,2,\text{avg}}$

7)　for $i = 0$ to a

8)　　if $(1+4i+8k)\%(8a) < 4a$

9)　　$t_{1+4i+8k} \leftarrow t_{1+4i+8k} + \Delta_k$

10)　end if

11)　if $(3+4i+8k)\%(8a) > 4a$

12)　　$t_{3+4i+8k} \leftarrow t_{3+4i+8k} + \Delta_k$

13)　end if

14)　end for

15) end if

16) if　$x_k = 1$ and $ipd_{k,1,\text{avg}} < ipd_{k,2,\text{avg}}$

17)　　for $i = 0$　to　a

18)　　if $(1+4i+8k)\%(8a) > 4a$

19)　　$t_{1+4i+8k} \leftarrow t_{1+4i+8k} + \Delta_k$

20)　　end if

21)　　if $(3+4i+8k)\%(8a) < 4a$

22)　　$t_{3+4i+8k} \leftarrow t_{3+4i+8k} + \Delta_k$

23)　　end if

24)　end for

25) end if

26)　for　$j = 1$　to　$s-1$

27)　　$t_{2j+8ak} = \dfrac{t_{2j+8ak-1} + t_{2j+8ak+1}}{2}$

28)　end for

29)　if　$k > 0$

30)　　$t_{2ks} = \dfrac{t_{2ks-1} + t_{2ks+1}}{2}$

31)　end if

32) end for

通过算法 1 可知，在 2 个相邻大组中间总是存在一个用于调节的数据分组，且该数据分组的发送时刻与下一个分组的第一个数据分组的发送时间变化有关，因此，以 $8a-1$ 个数据为一个分组可大大减小组间的耦合关系。

2.3　水印提取

当监听方捕获到监视数据分组流时，就对其 IPD 进行提取、处理和分析。具体过程如下。

首先计算每个分组中的 $ipd_{k,j,u}$，并通过比对 $ipd_{k,1,\text{avg}}$ 和 $ipd_{k,2,\text{avg}}$ 大小提取水印信息，如式(9)所示。

$$Y_i = \begin{cases} 1, & ipd_{k,1,\text{avg}} \geqslant ipd_{k,2,\text{avg}} \\ 0, & ipd_{k,1,\text{avg}} < ipd_{k,2,\text{avg}} \end{cases} \quad k = 1, 2, \cdots, L \tag{9}$$

然后，将提取出的二进制信息表示为卷积码序列 Y_i；最后对 Y_i 进行维特比解码，输出提取的水印序列 w'。具体水印提取算法如算法 2 所示。

算法 2

输入　$ipd_{1,1,\text{avg}}, ipd_{1,2,\text{avg}}, ipd_{2,1,\text{avg}}, ipd_{2,2,\text{avg}}, \cdots, ipd_{k,1,\text{avg}}, ipd_{k,2,\text{avg}}$

输出　$y_1, y_2, \cdots, y_k$

1) for　$k = 1$　to　L

2) if $ipd_{k,1,\text{avg}} < ipd_{k,2,\text{avg}}$

3)　　$y_k = 0$

4) else

5)　　$y_k = 1$

6) end for

7)　$Y \leftarrow y_1 y_2 \ldots y_k$

8)　基于格栅图解码 Y

9)　提取水印 w'

2.4　相似度计算

通常情况下，由于受网络噪声的干扰，按上述方法提取出的 w' 与发送方嵌入的水印 w 并不完全相同。为判断两者之间的关联性，可使用基于汉明距离或余弦相似度的算法。汉明距离通常对分析文本间的相似度比较准确，而余弦相似度则对数字间的相似度判断较为准确。鉴于此，本文采用余弦相似度计算提取水印序列 w' 与原始嵌入水印序列 w 之间的关系，其相似度计算式为

$$sim = \frac{\sum_{i=1}^{m} w_i w_i'}{\sqrt{\sum_{i=1}^{m} w_i^2} \sqrt{\sum_{i=1}^{m} w_i'^2}} \tag{10}$$

3　理论分析

3.1　顽健性

1) 数据分组传输时间抖动的干扰

多数网络信道中的数据分组传输抖动时间符合独立同分布[4,12,13]，设 $\overline{X_{k,1,t}}$ 和 $\overline{X_{k,2,t}}$ 分别表示 $ipd_{k,1,t}$ 和 $ipd_{k,2,t}$ 的数据分组传输抖动时间均值，其差值 $Y_{k,t}$ 为

$$Y_{k,t}=\overline{X_{k,1,t}}-\overline{X_{k,2,t}},\ t=1,2,\cdots,a \tag{11}$$

因此，整个数据分组序列增加的抖动时间为

$$\overline{Y_{k,a}}=\frac{1}{a}\sum_{t=1}^{a}Y_{k,t},\ k=1,2,\cdots,L \tag{12}$$

通常情况下，数据分组传输的抖动越大，对水印检出率的影响就越大。若预设的置信区间为$[-D_k,D_k]$，则能够正确检测出水印需要满足的条件为$\overline{Y_{k,a}}\in[-D_k,D_k]$。

根据中心极限定理，上述条件的满足概率为

$$p_{\text{rob}}[\overline{Y_{k,a}}<D_k]=p_{\text{rob}}\left[\frac{\sqrt{a}\overline{Y_{k,a}}}{\sigma_{Y,k}}<\frac{D_k\sqrt{a}}{\sigma_{Y,k}}\right]=\Phi\left(\frac{D_k\sqrt{a}}{\sigma_{Y,k}}\right)$$
$$\sigma_{Y,k}=\sqrt{Var(Y_{k,t})} \tag{13}$$

可见参数$\sigma_{y,k}$、D_k对水印检出率的影响可通过a的大小调节。增大分组中的 IPD 个数，可提高水印的正确检出率。其中，正确检出概率p_{rob}与a的函数关系，如图 2 所示。

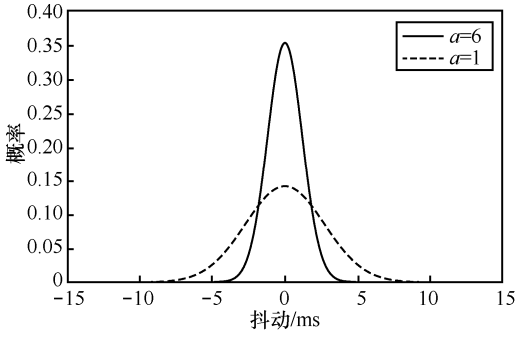

图 2 正确检出概率p_{rob}与a的函数关系

2) 分组丢失、分组合并现象的干扰

根据 2 种类型数据分组，上述现象又可以分为以下 2 种情况。

① 用于调节时序的数据分组丢失。此类数据分组丢失并不影响所嵌入的水印信息，只要将截获到的数据分组按照既定的规则获取分组的数据分组时间戳，并对其进行计算分析，即可判断是否含有水印信息。

② 用于嵌入水印信息的数据分组丢失。即在水印检测方截获的数据分组中找不到译码规则对应匹配的数据分组序号。此时，对不能构成完整$ipd_{k,1,u}$、$ipd_{k,2,u}$分组的数据分组不予分析，仅处理分组中包含完整的分组间隔序号的数据分组，通过对数据分组序号匹配[14]判断该数据分组是否可按要求进行统计，若能则进行水印提取的计算，若不能则予以舍弃。

当出现分组合并现象时，由于分组结束标识和分组头部的起始标识之间没有明显的界限，所以，其序号具有不确定性。对于此类情形，在接收端进行如下处理，若收到连续若干个序号为id_i，id_{i+1}，id_{i+2}的数据分组，其中，id_{i+1}和id_{i+2}差值为 1，而id_i与id_{i+1}差值大于 1，则将id_i序号按照$id_{i+1}-1$进行统计并分组，并将其视为因分组丢失而引起了相邻2个数据分组间隔时间发生改变，然后按照上述情况处理即可。

3) 哑分组、分组分片现象的干扰

哑分组只含有数据分组头部信息，没有数据信息，提取方可通过分析判断直接丢弃。分组分片对流水印的干扰与哑分组干扰的原理相同，均是在原始数据流中增加了冗余数据分组。因此，在处理过程中，需要对数据分组头部的分片标记进行检测，若属分片分组，则在提取水印时将其跳过。

值得指出的是，即使出现上述 3 种干扰无法矫正的情况，由于使用了卷积码的缘故，本文方法仍可以在一定限度内纠正传输过程中出错的水印信息。设维特比译码的纠错能力为c，任意长度的卷积码序列间的最小汉明距离为d_{free}，则

$$c=\left\lfloor\frac{d_{\text{free}}-1}{2}\right\rfloor \tag{14}$$

其中，$\lfloor x\rfloor$表示不超过x的最大整数。在本文中取$d_{\text{free}}=5$，即在每 4 bit 的水印信息中可纠正 1 bit 的误码。

3.2 抵御试探分组攻击

因为本文的水印采用了平滑技术，所以对于试探分组攻击，若不知道水印数据分组的组织方式，则试探的次数会随a的增大呈指数级增长。试探次数与a的函数关系如式(15)所示。

$$C_{8a-1}^{4a}=\frac{(8a-1)!}{(4a)!(4a-1)!} \tag{15}$$

当a的值增大 1 时，组合数的增长幅度为

$$K_{\text{inc}}=\frac{C_{8(a+1)-1}^{4(a+1)}}{C_{8a-1}^{4a}}\approx 2^8 \tag{16}$$

故本文中的水印算法具有较强抵御试探分组攻击的能力。

3.3 分组的盲检测

在水印提取方，采用滑窗方法来定位数据分组

的边界。首先选择一个起始 ipd_i，对后续所有 IPD 按水印嵌入算法设定的规则分组；然后再将 ipd_{i+1} 作为分组初始间隔，将其后续数据分组的 IPD 使用相同规则分组，最后通过译码和相似度计算，将出现概率最大的结果作为本次的检测值。

对于滑窗方法，最佳情况是接收到的所有数据分组完全对应水印嵌入分组，此时需要进行选择分组边界的次数为 0；最差的情况是接收方没有截获第一个数据分组，此时需要进行 $8a-1$ 次向后选择水印数据分组边界。所以正确获得水印数据分组边界的期望次数为 $\left\lfloor 4a-\frac{1}{2}\right\rfloor$。因此，当每 $8a-1$ 个数据分组为一组时，在 a 取值不大于 3 的情况下，至多需要探测 11 次，有效保证了水印提取的效率。

3.4　基于熵值的嗅探

设 N_t 表示嵌入水印信息的数据分组数量，N_s 表示调节数据流传输时间分布的数据分组数量，ipd_w 表示嵌入水印，ipd_s 表示调制间隔时间，两者交替出现。每当嵌入一个调制 ipd_s 时，认为产生一个新的符号 i，设此符号在整个样本空间中出现的概率为

$$P_i=\frac{1}{N_t+N_s} \tag{17}$$

则 ipd_s 的熵值为

$$H_s=-\sum_{i=1}^{N_s}P_i\mathrm{lb}P_i=\frac{N_s}{N_t+N_s}\mathrm{lb}(N_t+N_s) \tag{18}$$

由于插入 N_s，样本空间中水印符号的出现概率会受到影响。定义参数 x 为

$$x=\frac{N_t}{N_t+N_s} \tag{19}$$

则 ipd_w 的熵值为

$$\begin{aligned}H_t'&=-\sum xP_i\mathrm{lb}(xP_i)\\&=-\frac{N_t}{N_t+N_s}\mathrm{lb}\left(\frac{N_t}{N_t+N_s}\right)+\frac{N_t}{N_t+N_s}H_t\end{aligned} \tag{20}$$

其中，H_t 表示未增加 ipd_s 时，ipd_w 的初始熵值。

所以，整体数据分组流 IPD 的熵值结果为

$$\begin{aligned}H_l&=H_s+H_t'\\&=\mathrm{lb}(N_t+N_s)+\frac{N_t}{N_t+N_s}(H_t-\mathrm{lb}(N_t))\end{aligned} \tag{21}$$

本文采用 $\frac{N_t}{N_s}=\frac{1}{2}$ 的估算方法[15]，求得对应的熵值结果为

$$\begin{aligned}H_t&=\mathrm{lb}(2N_t)+\frac{1}{2}(H_t-\mathrm{lb}(N_t))\\&=\mathrm{lb}(2\sqrt{N_t})+\frac{1}{2}H_t\end{aligned} \tag{22}$$

由式(18)可知，一定存在 N_t，满足

$$\mathrm{lb}(2\sqrt{N_t})=\frac{1}{2}H_t \tag{23}$$

可见，嵌入水印后的数据分组流时间分布特征仍然可无限逼近于正常数据分组流的分布特征，所以，本文方法能够有效抵御基于熵值的水印嗅探攻击。

4　实验结果与分析

为验证本文中水印的性能，建立了相关的实验环境。为模拟网络中的不同噪声，设计了干扰器[注1]。此外，为降低耗时，实验选取 $a=1$，即每 7 个数据分组为一组，第 $8i$ $(i=1,2,\cdots,k)$ 个数据分组的发送时间则根据第 $8i-1$ 和第 $8i+1$ 数据分组的发送时间确定。

4.1　抖动干扰对水印检出率的影响

网络中正常传输的数据分组 IPD 值分布所满足的均值和方差各不相同。实验首先分析具有代表性的正态分布，标准差分别取 $\sigma=1$ 和 $\sigma=3$，且将 IPD 的范围分别限制在[15, 25]、[15, 30]、[15,35]、[15,40] ms，即符合同一值域分布的 IPD 最大差值分别为 10 ms、15 ms、20 ms 和 25 ms。

1)　σ=1 时水印的检出率

表 1 所示为是 $\sigma=1$ 时，对于不同均值 μ，提取的卷积码序列 Y 与原始水印序列 W 的相似率。实验表明，数据分组间隔时间越大，传输抖动对水印正确提取的影响越小。在本组实验中，水印检出率均在 98%以上，如图 3 所示。若置信区间取[0.8,1]，则在标准差不大于 1 的情况下，可准确检出所有水印。

表 1　　σ=1 卷积序列 Y 相似率分布

抖动均值 μ /ms	相似率			
	[15,25] ms	[15,30] ms	[15,35] ms	[15,40] ms
2	0.937	0.937 5	0.937 5	0.940
4	0.906	0.912 8	0.937 5	0.940
8	0.812	0.875 0	0.937 5	1.000
16	0.750	0.777 7	0.818 0	0.833

注1：采用本课题组在隐通道相关基金项目资助下开发的模拟网络噪声发生器，可以用其在实验网络中主动注入数据分组传输的抖动时间，以及实施分组丢失、分组合并等操作。

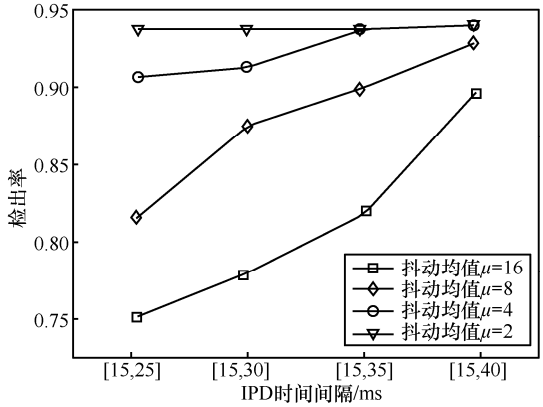

图 3　σ=1 时水印检出率

2) σ=3 时水印的检出率

表 2 和表 3 中的数据是 $\sigma=3$ 时，对于不同均值 μ，提取的卷积码序列 Y 及水印序列 W 相似率分布情况。实验数据表明，即使提取到的卷积码序列与原始卷积码序列的相似率值较低，因为对卷积码序列采用了进一步自纠错解扩措施，后期计算得出的相似率也均在 0.85 以上，所以若置信区间取 [0.8,1]，可准确检出所有水印。IPD 变化与水印检出率的关系如图 4 和图 5 所示。

表 2　　σ=3 卷积序列 Y 相似率分布

抖动均值 μ /ms	相似率			
	[15,25] ms	[15,30] ms	[15,35] ms	[15,40] ms
2	0.812	0.906	0.937	0.937
4	0.750	0.840	0.850	0.875
8	0.730	0.812	0.833	0.854
16	0.700	0.812	0.820	0.836

表 3　　σ=3 水印序列 W 相似率分布

抖动均值 μ /ms	相似率			
	[15,25] ms	[15,30] ms	[15,35] ms	[15,40] ms
2	0.880 0	0.953 0	0.966	1.000
4	0.920 0	0.933 0	0.946	1.000
8	0.912 9	0.925 8	0.933	0.945
16	0.894 4	0.912 9	0.925	0.966

图 5 中的 4 条曲线分别表示当数据分组传输抖动时间的无偏标准差 σ=3，均值为 2 ms、4 ms、8 ms、16 ms 的随机抖动时间对水印检出率的影响。图 5 中的数据表明，水印的检出率随 IPD 取值范围的递增而严格单调递增，且随着 IPD 取值范围的递增，数据分组传输的抖动时间对水印检出率的干扰越来越小。

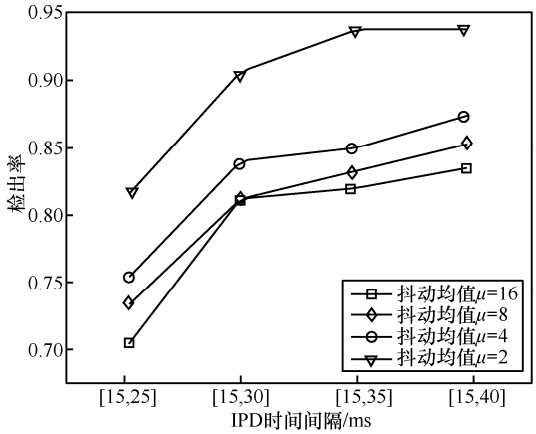

图 4　σ=3 卷积序列 Y 检出率

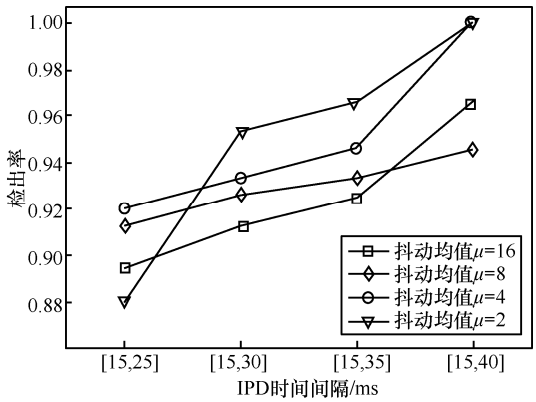

图 5　σ=3 水印序列 W 检出率

图 5 中出现曲线交叉的原因是，在水印的校正算法中，IPD 处于预设范围之外的情况被舍弃。当 IPD 的取值范围在[15,40] ms，随机抖动 $\sigma=3$，均值 $\mu=16$ 时，较高的偏离值则被舍弃，因此，随机抖动的影响迅速减弱，水印的检出率出现较大的增长，致使 $\mu=16$ 的曲线向上穿越 $\mu=8$ 的曲线。$\mu=2$ 的曲线向上穿越 $\mu=4$ 与 $\mu=8$ 的曲线原因相同。由此可以判定，校正算法的使用有效提高了水印的平均正确检出率。

3) 对比实验

为比较本文中方法与基于原始分组分组技术[4]和 IBW[5]水印技术对数据分组传输时间抖动干扰的抵御能力，在相同的网络环境下进行对比实验。实验结果表明：本文提出的水印方法在时延抖动值不超过发送分组平均间隔时间的 40%时，均能保证水印的正确检出率大于 0.9，而基于原始分组技术和 IBW 水印技术在抖动值增大时，其水印的正确检出

率明显下降。其中，数据分组的传输抖动时间与水印正确检出率的关系如图 6 所示。

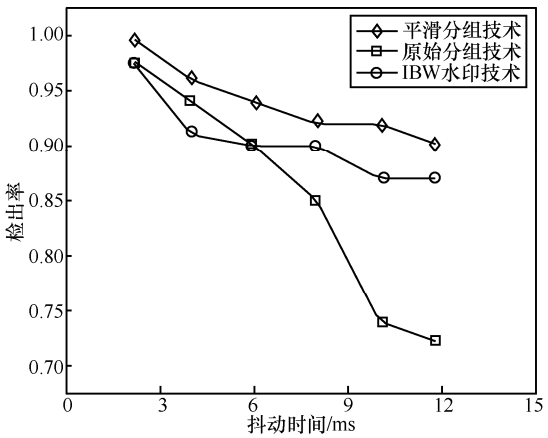

图 6　不同技术检出率对比（时间干扰）

4.2　分组丢失的影响

本组实验分析了本文中的水印方法，以及基于原始分组和 IBW 水印技术的正确检出率与网络分组丢失率的关系，实验结果如图 7 所示。

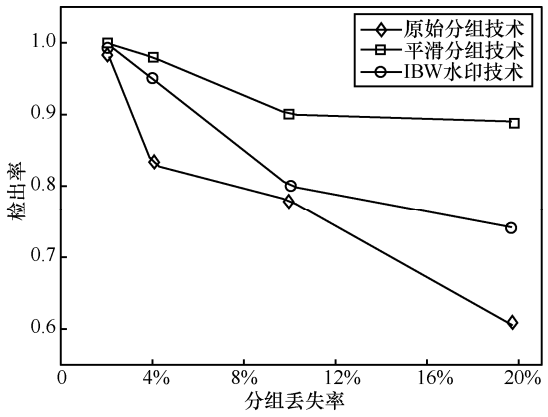

图 7　不同技术的分组丢失率对检出率影响

当网络分组丢失率小于 4%时，基于原始分组技术的水印检出率最低，而当分组丢失率达到 12%~20%时，基于原始分组技术与 IBW 的水印技术的检测率均降至 0.75 以下，而本文中给出的水印方法检出率最高，当设定阈值为 0.8 时，可全部正确检出。

4.3　误判率与准确率

本组实验中，数据分组流 IPD 满足 $\mu = 25, \sigma = 5$ 的正态分布且不含有水印信息。为检测不同网络干扰值对水印误判率的影响，分别设计了有抖动和无抖动 2 组实验。

表 4 中的数据是在数据分组流中额外增加范围为[−2,10] ms 的干扰后，提取出的卷积码序列与原始卷积码序列的相似率（Y 类型序列），以及解扩后的译码水印序列和“伪水印序列”的相似率（W 类型序列）；表 5 中的数据是在未增加额外网络干扰的情况下，Y 类型序列和 W 类型序列的相似率。

表 4　　有抖动时的相似率

序列类型	相似率			
	8 位	16 位	24 位	32 位
Y	0.593 7	0.500	0.479 0	0.437 5
W	0.547 7	0.559	0.527 0	0.500 0

表 5　　无抖动时的相似率

序列类型	相似率			
	8 位	16 位	24 位	32 位
Y	0.625 0	0.531	0.479	0.422
W	0.507 1	0.500	0.472	0.470

表 4 和表 5 中的数据表明，无论是否添加网络传输抖动干扰，对于未嵌入水印的数据分组流，提取方采用本文中的方法计算后，最终得到的水印序列 W 的相似率均低于 0.6；当水印为 24 位时，序列相似率已经低于 0.53，其值都处于置信区间之外。

为比较不同水印方法的误判率，本组实验在数据分组发送时间范围取[15,40] ms 的情况下，分别测试了本文方法与基于原始分组技术的误判率，结果如表 6 所示。本文方法对于未嵌入水印的数据分组流若按照水印嵌入的方式进行译码，则与随机给出的序列相似率总会低于 0.475，而基于原始分组技术则在 0.5±0.05 范围内波动，因此若阈值设置较低时，本文方法误判率较低。

表 6　　不同技术误判率比较

方法名称	误判率				
	μ=2 ms	μ=4 ms	μ=6 ms	μ=8 ms	μ=10 ms
平滑分组技术	0.47	0.45	0.45	0.43	0.42
原始分组技术	0.54	0.55	0.49	0.49	0.48

在本组实验中，还分别设置了 200 条含水印和 200 条不含水印的数据分组流，并统计检测准确率，结果如表 7 所示。当阈值范围以 0.1 为步长变窄时，本文方法的检测准确率仅会出现小幅度下降，而基于原始分组技术的准确率会发生大幅下降。

表 7 不同技术检测准确率比较

阈值范围	有水印嵌入序列的检测准确率		无水印嵌入序列的检测准确率	
	平滑分组技术	原始分组技术	平滑分组技术	原始分组技术
0.6~1.0	1.0	1.0	0.9	0.8
0.7~1.0	1.0	0.9	1.0	0.9
0.8~1.0	1.0	0.84	1.0	1.0
0.9~1.0	0.9	0.8	1.0	1.0

4.4 水印的隐蔽性

一般情况下，网络中正常数据分组流的 IPD 大致满足正态分布[12,16]，因此，可以通过比较水印 IPD 序列与网络中正常数据分组流 IPD 分布的相似度来检测水印的隐蔽性。

F 检测通常被用来判断两类样本分布的相似程度。在本组实验中，每组数据分组在传输中使用相同的随机时延抖动。表 8 中的数据是本文方法与基于原始分组水印技术的隐蔽性对比。

表 8 数据分组传输时间序列分布对比

时间序列名称		平滑分组技术		原始分组技术	
		μ	σ^2	μ	σ^2
嵌入水印时间分布	嵌入方 a	26.70	47.66	27.79	26.1
	检测方 b	28.18	47.03	28.72	40.2
未嵌入水印时间分布	嵌入方 c	26.93	43.75	27.41	41.50
	检测方 d	28.13	43.50	28.82	64.79

对表 8 中的序列 a、b 和 c、d，以及 b、d 进行 F 检测计算，其中样本量为 $n_1=120$，$n_2=120$，显著性水平分别取 $\alpha=0.1$、$\alpha=0.05$、$\alpha=0.01$，其拒绝域临界值的计算结果为

$$F_{0.01}(120,120)=1.53,\quad F_{0.05}(120,120)=1.35$$

$$F_{0.1}(120,120)=1.26,\quad \frac{1}{F_{0.01}(120,120)}=0.65$$

$$\frac{1}{F_{0.05}(120,120)}=0.74,\quad \frac{1}{F_{0.1}(120,120)}=0.79$$

$F_{x/y,i}$ 中的 x/y 表示对比序列，i =1 和 2 分别代表本文水印方法与基于原始分组的水印方法，则当 $\alpha=0.01$ 时，对不同序列有

$$F_{b/a,1}=\frac{\sigma_b^2}{\sigma_a^2}=\frac{47.03}{47.66}=0.986\,7<F_{0.01}(120,120)$$

$$F_{d/b,1}=\frac{\sigma_d^2}{\sigma_b^2}=\frac{43.50}{47.03}=0.92<F_{0.01}(120,120)$$

$$F_{d/c,1}=\frac{\sigma_d^2}{\sigma_c^2}=\frac{43.50}{43.75}=0.994\,2<F_{0.01}(120,120)$$

则 $\frac{1}{F_{0.01}(120,120)}<F_{b/a,1}$，$\frac{1}{F_{0.01}(120,120)}<F_{d/b,1}$，$\frac{1}{F_{0.01}(120,120)}<F_{d/c,1}$。

同理，可分别求得显著水平 α 取值 0.05 和 0.1 时，$F_{b/a,1}$、$F_{d/c,1}$、$F_{d/b,1}$ 的范围，所得结果均处在 $[\frac{1}{F_\alpha(120,120)},F_\alpha(120,120)]$ 内，随着拒绝域的缩小，本文水印方法的隐蔽性仍可得到保持。

进一步对基于原始分组的水印技术也进行相同的 F 检测，可以得到当 α=0.1 时，其值为

$$F_{b/a,2}=\frac{\sigma_b^2}{\sigma_a^2}=\frac{40.2}{26.1}=1.54$$

可见，$F_{b/a,2}>F_{0.01}(120,120)$，$F_{b/a,2}>F_{0.05}(120,120)$ 与 $F_{b/a,2}>F_{0.1}(120,120)$，即当显著水平 α 取 0.01、0.05 或 0.1 时，其检测值均超出拒绝域的范围。

4.5 修正熵值检测

为了使水印数据分组流能够有效抵御 CCE 检测，需要保持样本信息源间的关联关系一致。根据算法 1 及式(8)，本文方法对 IPD 调制量较小，且处理计算均使用动态方式，因此，相邻 IPD 间的关联并未显著破坏。所以，本文方法对基于 CCE 的水印嗅探攻击有较好的抵御能力。

在本组实验中，将误码率检测的阈值设为 0.01，判断水印时间序列的临界值设为网络中正常数据分组流 IPD 样本 CCE 值的 90%～99%；临界值根据测试样本值的范围选取。利用式(4)~式(6)计算嵌入水印的数据分组流样本与正常数据分组流的 CCE 值，若两者比值落在[0.95,1]中且水印相似度的错误率不大于 0.01，则认为可有效抵御基于 CCE 的嗅探攻击。

实验中，IPD 的取值范围设在[15,40] ms, 其抖动干扰满足 $\sigma=3$，μ=4 的正态分布，将时间序列按照步长 5 ms 划分为 6 个区间：$[a,20),[20,25),[25,30),[30,35),[35,40),[40,b]$，其中，2 个边界值 a 和 b 的选取要考虑数据分组传输抖动的影响，因此，在实验中分别取 9 和 55，以保证落在这 6 个区间中。该区间

与集合 M 中符号的对应关系为

$$[a,20)\to 1 \quad [20,25)\to 2 \quad [25,30)\to 3$$
$$[30,35)\to 4 \quad [35,40)\to 5 \quad [40,b]\to 6$$

分别对所有区间中的 IPD 进行统计，本文中仅取其前 40 个 IPD 值：27.0、23.0、30.0、32.0、20.0、28.0、25.0、21.0、36.0、21.0、33.0、23.0、39.0、24.0、28.0、31.0、25.0、22.0、19.0、20.0、32.0、12.0、26.0、43.0、20.0、36.0、27.0、23.0、33.0、22.0、32.0、36.5、23.5、22.5、21.5、37.0、23.5、21.5、28.5、13.5。得到符号 1 至符号 6 的出现概率分别为 $\frac{4}{40}$、$\frac{15}{40}$、$\frac{8}{40}$、$\frac{7}{40}$、$\frac{5}{40}$、$\frac{1}{40}$。若计算长度为 m=3 序列的 CCE，需要先计算其基本熵

$$H=-\sum_{i=1}^{3}p(i)\mathrm{lb}p(i)$$
$$=-\left(\frac{4}{40}\mathrm{lb}\frac{4}{40}+\frac{15}{40}\mathrm{lb}\frac{15}{40}+\frac{8}{40}\mathrm{lb}\frac{8}{40}\right)$$

将 40 个 IPD 按照步长为 3 进行分组，共得到 13 个分组中存在 2 个符合条件的分组，所以有

$$perc(X_3)=\frac{2}{13}$$

$$H(X_3\mid X_2)=H(X_2\mid X_1)=-\sum_{i=1}^{3}\sum_{i=1}^{3}P(ij)\mathrm{lb}P(j\mid i)$$

根据文献[15]，对离散有限集合使用 CCE 估算信源熵值的计算公式为

$$CCE(X_i\mid X_{i-1})=H(X_i|X_{i-1})+perc(X_i)H \quad (24)$$

其中，$perc(X_i)$ 表示长度为 i 的子序列占相同长度全部子序列的百分比。可以分别求得正常数据分组流与嵌入水印数据分组流的 CCE 值，如表 9 所示。

表 9　　CCE 熵值统计对比

检测方法	CCE 值
正常数据分组流	1.95
水印数据分组流	1.89

表 9 中的数据说明，本文中的水印方法可以有效抵御基于 CCE 的水印嗅探攻击。

5　结束语

本文提出了一种基于数据分组平滑分组的网络主动流水印方法。通过将含有水印的网络流数据分组间隔时间划分为用于嵌入水印信息和用于调制数据分组流传输的时间分布特征 2 种类型。基于交叉分组的动态关联方式处理数据分组，使处理后的数据分组流间隔时间特征能无限逼近正常网络分组。详细的理论分析与实验均表明，该方法具有顽健性好、检出率高等特点，而且可以抵抗基于修正熵值的水印嗅探攻击。未来的研究工作将重点提高水印信道的容量，降低用于平滑传输时间分布的数据分组数量。

参考文献：

[1] 何高峰, 杨明, 罗军舟, 等. 洋葱路由追踪技术中时间特征的建模与分析[J]. 计算机学报, 2014, 2(37):357-371.
HE G F, YANG M, LUO J Z, et al. Modeling and analysis of time characteristics used in onion routing traceback techniques[J]. Chinese Journal of Computers, 2014, 2(37): 357-371.

[2] 郭晓军, 程光, 朱琛刚, 等. 主动网络流水印技术研究进展[J]. 通信学报, 2014, 35(7): 178-192.
GUO X J, CHENG G, ZHU C G, et al. Progress in research on active network flow watermark[J]. Journal on Communications, 2014, 35(7): 178-192.

[3] WANG X, REEVES D S. Robust correlation of encrypted attack traffic through stepping stones by manipulation of interpacket delays[C]//ACM Conference on Computer and Communications Security. 2003: 20-29.

[4] PAN Z, PENG H, LONG X, et al. A watermarking-based host correlation detection scheme[C]//International Conference on Management of e-Commerce and e-Government. 2009:493-497.

[5] PYUN Y J, PARK Y H, WANG X, et al. Tracing traffic through intermediate hosts that repacketize flows[C]//IEEE International Conference on Computer Communications. 2007: 634-642.

[6] WANG X, CHEN S, JAJODIA S. Network flow watermarking attack on low-latency anonymous communication systems[C]//IEEE Symposium on Security and Privacy. 2007: 116-130.

[7] KIYAVASH N, HOUMANSADR A. BORISOV N. Multi-flow attacks against network flow watermarking schemes[C]//USENIX Security Symposium. 2008: 307-320.

[8] 张连成, 王振兴, 孙建平. 基于时间间隔的扩频流水印技术[J]. 计算机应用研究, 2011, 28(8): 3049-3053.
ZHANG L C, WANG Z X, SUN J P. Interval-based spectrum watermarking scheme for tracing network flows[J]. Application Research of Computers, 2011, 28(8): 3049-3053.

[9] 张璐, 罗军舟, 杨明, 等. 基于时隙质心流水印的匿名通信追踪技术[J]. 软件学报, 2011, 22(10): 2358-2371.
ZHANG L, LUO J Z, YANG M, et al. Interval centroid based flow watermarking technique for anonymous communication traceback[J].

Journal of Software, 2011, 22(10): 2358-2371.

[10] LUO J, WANG X, YANG M. An interval centroid based spread spectrum watermarking scheme for multi-flow traceback[J]. Journal of Network and Computer Applications, 2012, 35(1): 60-71.

[11] RAMYA K P, REVATHI M K. Protecting streaming data using provenance spread spectrum watermarking[J]. Research in Computer and Communication Engineering, 2014, 3(2):2109-2114.

[12] LUO X, ZHANG J, PERDISCI R, et al. On the secrecy of spread-spectrum flow watermarks[C]//European Symposium on Research in Computer Security. 2010: 232-248.

[13] DENGLE S, LOMTE S. Active watermarking approach in detecting encrypted traffic attack by making correlation scheme robust[J]. International Journal of Science and Research, 2014, 8(3): 691-695.

[14] QURESHI A, MEGÍAS D, RIFÀ-POUS H. Framework for preserving security and privacy in peer-to-peer content distribution systems[J]. Expert Systems with Applications, 2015, 42(3):1391-1408.

[15] GIANVECCHIO S, WANG H. DSSS-based flow marking technique for invisible traceback channels: an entropy-based approach[C]//ACM Conference on Computer and Communications Security. 2007: 307-316.

[16] HOUMANSADR A, KIYAVASH N, BORISOV N. Rainbow: a robust and invisible non-blind watermark for network flows[C]//16th Annual Network& Distributed System Security Symposium(NDSS'09). 2009: 224-236.

作者简介：

金华（1977-），男，江苏海安人，江苏大学博士生、副教授，主要研究方向为隐私保护、网络与信息安全。

朱慧（1989-），女，山西大同人，江苏大学硕士生，主要研究方向为网络安全。

王昌达（1971-），男，江苏南京人，博士，江苏大学教授、博士生导师，主要研究方向为网络安全、无线传感器网络、云计算。