MULTI-FLOW ATTACK RESISTANT WATERMARKS FOR NETWORK FLOWS

*Amir Houmansadr †, Negar Kiyavash ‡, Nikita Borisov †**

† Dept. of Electrical and Computer Engineering, UIUC
‡ Dept. of Computer Science, UIUC

ABSTRACT

In this work we present a Multi-flow Attack Resistant Interval Centroid Based Watermarking (MAR-ICBW) scheme for network flows. Our proposed scheme can withstand the newly introduced multi-flow watermarking attack that defeats the state-of-the-art interval-based network flow watermarking schemes. Multi-flow attack uses the dependent correlations among the flows marked with the same watermark to recover the secret parameters, and remove the watermark from a flow. The attack can be effective even if different flows are marked with different values of a watermark. MAR-ICBW survives the attack by virtue of randomizing the location of the embedded watermark across multiple flows and therefore, effectively removing the correlations between the flows. While we represent our counter measure to multi-flow attack in terms of an improved version of ICBW, the same methodology can be used to strengthen other interval-based flow watermarking schemes.

Index Terms— Flow Linking, Flow Watermarks, Multi-flow Attack

1. INTRODUCTION

As cyberspace privacy and security become more of a concern to users, traffic analysis which is the practice of inferring sensitive information from communication patterns receives more attention. Traffic analysis has been particularly studied in the context of anonymous communication systems, where features such as packet timings, sizes, and counts can be used to link two flows and break anonymity guarantees [1]. Another application of traffic analysis is in intrusion detection where, for example, it is applied to detection of stepping stones within an enterprise [2].

Recently, *network flow watermarks* have been used to aid traffic analysis [3, 4, 5, 6, 7]. In this case, traffic patterns of one flow (usually packet timings) are actively modified to convey a message (aka watermark). If the same watermark is later found on another flow, the two are considered linked. Watermarking significantly reduces the computation and communication costs of traffic analysis, and may also lead to more precise detection with fewer false positives. Watermarking has been applied to both the problems of attacking anonymity systems [4, 6, 7] and detecting stepping stones [3, 5]. Both applications require that many flows must be watermarked before linked flows can be discovered.

Recently, Kiyavash et al. [8] showed that an attacker can learn enough information to defeat the watermark by observing multiple watermarked flows [1]. The multi-flow threat attack of [8] defeats the latest generation of *interval-based watermarks* [5, 6, 7]. These watermarks subdivide the flow into discrete time intervals and perform transformative operations on an entire interval of packets. This renders the watermark more robust to packet losses, insertions, and repacketization than previous approaches that focused on individual packets [3, 4]. However, the same interval-based embeding approach can be used by attackers to "line up" multiple watermarked flows and observing the transformations that were inserted.

In this work we present a counter measure that modifies the Interval Centroid Based Watermarking scheme (ICBW) of [6]. However, our countermeasure applies to other *interval-based watermarking* schemes such as IBW and DSSS presented in [5, 7]. Our modified watermarks can withstand the multi-flow attack of [8]. We show that by using multiple "seeds" (interval assignments) to watermark different flows, it is possible to survive the multi-flow attack. This countermeasure comes at a cost of higher computation overhead at the detector and a higher rate of false positives. This increased cost is only linear, whereas the increased cost for the attacker is superexponential, thus providing an effective defense.

The rest of the paper is organized as follows. Section 2 presents briefly the application of network flow watermarking. Section 3 describes out proposed scheme MAR-ICBW after first discussing the original ICBW scheme and why it is vulnerable to multi-flow attack of [8]. In Subsection 3.2.1, we discuss the performance of MAR-ICBW. Finally, we discuss why MAR-ICBW is superior to other potential countermeasure of choosing multiple watermark values in Subsection 3.2.2.

*This research was supported in part by NSF grants CCF 07-29061 and CNS 08-31488.

[1] The term "attacker" here refers to someone attacking the watermarking scheme regardless of the purpose of the watermark insertion (e.g. the watermarks in anonymous communication systems are often inserted for malicious purposes.)

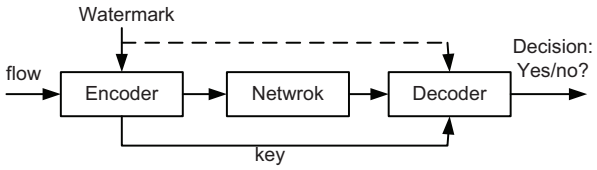


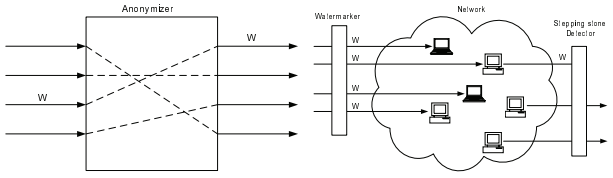
Fig. 1. Network Flow Watermarking

2. NETWORK FLOW WATERMARKING

The setting for network flow watermarking is similar to that of other digital media watermarks. The general model, as shown in Figure 1, involves a network flow (i.e., a collection of packet interarrival times, modeled as a point process) passing through a watermarking point (typically a router of some sort) that transforms, or *distorts*, the flow in some way (typically by modifying packet timings by selectively delaying some packets). In the general setting, the watermarker has a secret *key* and uses it to encode a *message* in the traffic characteristics.

After watermarking, the flow undergoes some natural or intentional distortion. Natural distortion can take the form of delays at intermediate routers (or rather, variability of delays, i.e., *jitter*), but may also include dropped or retransmitted packets, repacketization, and other changes. In addition, an attacker may intentionally distort traffic characteristics in order to prevent the watermark from being recovered.

The distorted flow finally arrives at a detection point. The detector shares the secret key and uses it to extract the message encoded in the watermark. A good watermark will allow reliable recovery of the message from the watermarked flow despite the intermediate distortion.



(a) Defeating Anonymous Systems (b) Detection of Stepping Stones

Fig. 2. Two Applications of Network Flow Watermarking

In network flow watermarks, the *message* component of the watermark may be used in two ways. First, all watermarked flows may be marked with a single message. In this case, the detector's main goal is to decide whether the watermark is present or not by checking whether the decoded message is the correct one. Alternately, different flows may have a different message embedded, so that when a watermarked flow is detected, it can be linked with a particular marked flow. This comes at a cost of less reliable detection, since the single-message context creates more opportunities to detect errors. The multi-flow attack of [8] is designed to work in both single-message and multiple-message contexts. Our proposed multi-flow attack resistant watermarking scheme also works in both of the aforementioned scenarios.

Two main outlets for use of watermarks in network flows are anonymous communication systems and in detection of stepping stones. At a very high level, an anonymous system maps a number of input flows to a number of output flows while hiding the relationship between them, as shown in Figure 2 (a). The goal of an attacker, then, is to link an incoming flow to an outgoing flow (or vice versa). A watermark can be used to defeat anonymity protection by marking certain input flows and watching for marks on the output flows. The second application of watermarking in networks is detection of stepping stone, a host that is used to relay traffic through an enterprise network to another remote destination, in order to hide the true origin of the flow. To detect such hosts, an enterprise must be able to link an incoming flow to the relayed outgoing flow. The situation is therefore very similar to an anonymous communication system, where a border router for an enterprise will insert watermarks on all incoming flows, and check for the presence of the mark on all outgoing flows, as shown in Figure 2 (b). Once a watermark is detected the outgoing flow is terminated and therefore, the stepping stone attack is prevented.

3. MAR-ICBW: A MULTI-FLOW ATTACK RESISTANT WATERMARK BASED ON ICBW

We propose our counter measure to multi-flow attack of [**?**] by improving the scheme proposed by Wang et al. [6]. However, our counter measure also applies to the other interval-based watermarking schemes [5, 7] that previously were defeated by multi-flow attack of [8].

Next we give a brief description of ICBW and why it is vulnerable to the multi-flow attack of [8]; for more details of the scheme as well as some analysis we refer the reader to [6]. Likewise more details on the multi-flow attack can be found in [8].

3.1. Interval Centroid-Based Watermarking (ICBW)

At high level, the scheme is based on dividing the stream into intervals of equal lengths, using two parameters: o, the offset of the first interval, and T, the length of each interval. A subset of $2n = 2rl$ of these intervals are chosen at random, and then randomly divided into two further subsets A and B each consisting of $n = rl$ intervals. Each of the sets A and B are randomly divided to l subsets denoted by $\{A_i\}_{i=1}^{l}$ and $\{B_i\}_{i=1}^{l}$, each consisting of r intervals. The i-th watermark bit is encoded using the sets $\{A_i, B_i\}$. Therefore, a watermark of length l can be embedded in the flow.

The watermarker and detector agree on the parameters o, T and use a random number generator (RNG) and a seed s to

randomly select and assign intervals for watermark insertion. To keep the watermark transparent, all of these parameters are kept secret. Depending on whether the i-th watermark bit is 1 or 0, the watermarker delays the arrival times of the packets at the interval positions in sets A_i or B_i respectively, by a maximum of a. Figure 3 illustrates the effect of this delaying strategy over the distribution of packet arrival times in an interval of size T (this operation is called "squeezing" by Wang et al.).

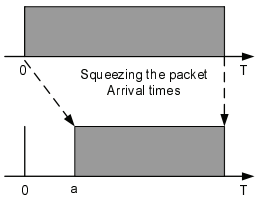
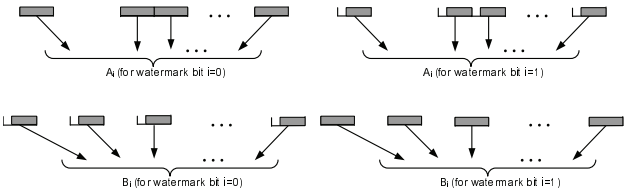


Fig. 3. Distribution of packet arrival times in an interval of size T before and after being delayed.

The overall watermark embedding is illustrated in Figures 4 (a) and (b). As the result of this embedding scheme, the expected value of aggregate centroid, i.e., the average offset of the packet arrival time from the beginning of the current length T interval, in either the intervals A_i (when watermark bit is 1) or B_i (when watermark bit is 0) corresponding to bit i is increased by $\frac{a}{2}$. The difference between the aggregate centroid of A_i and B_i now will be $\frac{a}{2}$ when watermark bit is 1 or $-\frac{a}{2}$ when watermark bit is 0.

The detector checks for the existence of the watermark bits. The check on watermark bit i is performed by testing whether the average difference of the aggregate centroid of packet arrival times in the intervals A_i and B_i is closer to $\frac{a}{2}$ or $-\frac{a}{2}$. If it is closer to $\frac{a}{2}$, then the watermark bit is decoded as 1 and if it is closer to $-\frac{a}{2}$, the bit is declared a 0. By focusing on the arrival times of many intervals (r of them for each bit of the watermark) rather than individual packet timings, the ICBW approach is robust to repacketization, insertion of chaff, and mixing of data flows. Network jitter can shift packets from one interval into another, but the suggested parameters for a and T (350ms and 500ms respectively) are large enough that few packets will be affected.

The secrecy of the interval positions A_i and B_i make the mark difficult to detect or remove, as it is hard to distinguish the patterns generated by the mark from natural variation in traffic rates. However, Kiyavash et al. show that a multi-flow attack technique allows an observer to effectively recover the watermark positions and values [8]. This technique is applicable to any watermarking scheme that creates periods of clear or low traffic at *specific* parts of the flows across many flows such as [5, 7]. More precisely, while it is highly unlikely to observe same periods of clear interval across independent flows, interval-based watermarking schemes such as ICBW



(a) Insertion of watermark bit 0 (b) Insertion of watermark bit 1

Fig. 4. ICBW bit insertion

clear same parts of network flows as the seed s for selecting the random intervals remains the same across all flows. Therefore, the average copy of the watermarked flows always exhibits patterns of no arrivals that exceeds the normal silent periods in unwatermarked traffic that give away the location of the watermark as well as the parameters of the watermarking scheme.

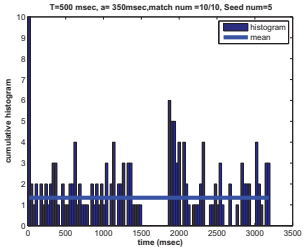
3.2. Multi-flow Attack Resistant Watermark

The main vulnerability of interval-based watermarking schemes of [5, 6, 7] is that they embed the watermark in the same positions in the flows. Therefore, an attacker that observes multiple watermarked flows can align them to render the watermarks visible. However, if the watermark was embedded using different positions, [2] the alignment approach of [8] would fail. Therefore, we suggest an improved scheme where the encoder uses multiple seed values, $s_1, \ldots, s_n$, and pick one of them at random for each flow. To deal with this, the detector would need to try to recover the watermark with each possible s_i and pick the best match. Once again, the probability of error grows with the number of the possible positions of the watermark n, but increased redundancy can again be used to make up for it. Note that the probability of error falls exponentially with increased redundancy, but grows only roughly linearly with n.

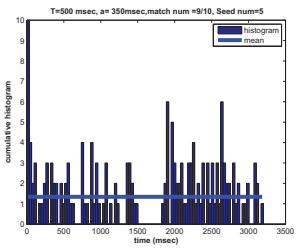
3.2.1. Analysis

When flows are marked using multiple seed values, the attackers can still execute the attack of [8]; however, the complexity grows quickly out of control. The probability of a given set of k flows using the same seed is $\left(\frac{1}{n}\right)^{k-1}$, which falls quite quickly even when $k = 10$ (the number of flows recommended for successful execution of attack in [8]). By the pigeon hole principle, within $n(k-1)+1$ flows we can always find a subset of k flows with the same seed, but the search space of all $\binom{n(k-1)+1}{k}$ subsets grows superexponentially in n. For example, with $n = 5$ and $k = 10$, $\binom{46}{10} > 10^9$, resulting in an infeasible number of subsets to enumerate.

[2] This is also true in case of IBW scheme [5]. However for DSSS scheme [7], this counter measure means that different PN codes have to be chosen across different flows.

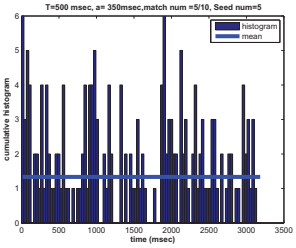


(a) Case of 10 out of 10 match

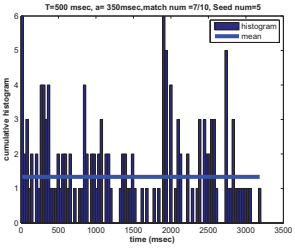


(b) Case of 9 out of 10 match

Fig. 5. Multi-flow Attack against MAR-ICBW with 5 watermark seeds



(a) Case of 5 out of 10 match



(b) Case of 7 out of 10 match

Fig. 6. Multi-flow Attack against MAR-ICBW with 5 watermark seeds

However, even when not all the flows are marked using the same seed, the attackers can notice that certain intervals have fewer than usual packets. Figures 5 and 6 show the results of our implementation of multi-flow attack for our proposed watermark scheme, MAR-ICBW. In the simulations we choose $n = 5$ seeds for location of watermark. The parameters of the watermark are $T = 500$ msec and $a = 350$ msec, respectively.

Figure 5 (a) shows that when all 10 flows (as recommended in [8]) are watermarked using the same seed, the multi-flow attack reveals the cleared interval 1500-1850 msec. As depicted in Figure 5 (b), when 9 out of 10 flows are marked with the same seed, the watermarked interval of 1500-1850 msec is still visible. However, still $\binom{41}{9} > 10^8$ subsets of flows are needed to be tested which is infeasible. Figures 6 (a) and (b) show that by looking at fewer matches (here, 5 and 7), the clear watermark is no more detectable. Note that finding a 5 flow or 7 flow match still requires $\binom{21}{5} = 20349$ and $\binom{31}{7} > 10^6$ subsets that are not even potentially useful.

Note that this increased attack resistance comes at the cost of higher computation overhead at the decoder. However, this increased cost is only linear, whereas the increased cost for the attacker is superexponential. The Probability of detection of our scheme remains the same as that of ICBW [6], where $P_D \sim 1$ for $r = 20$ repetitions of the watermark in the flow. The false positive rates of our scheme though slightly increased remain at the order of 10^{-3} with $r = 20$ repetitions of the watermark. Therefore with slight increase of computation cost at the decoder, we provide an effective defense.

3.2.2. Discussion

Another possible counter measure that comes to mind is the use of multi-message watermarks. In other words, if different watermarks are embeded in different flows, the aggregation performed by multi-flow attack will no longer work, since by switching between 1 and 0 bits, ICBW applies different transforms to different intervals. More specifically, in ICBW a given interval may be squeezed when a certain bit is 0, and not squeezed when that bit is 1. By aggregating flows where that bit changes, no empty periods will be detected.

However, by observing a few more flows, multi-flow attack still detects the presence of a watermark. Given a bit b and a set of $2k-1$ flows, by the pigeon hole principle, there exists a subset of k flows where the bit has the same value. Thus, to detect the watermark, it is enough to examine $\binom{2k-1}{k}$ subsets of k flows out of a collection of $2k-1$. The number of such subsets is, of course, superexponential in k. However, the multi-flow attack works with as few as k around 10, which makes such a search feasible, as $\binom{19}{10} = 92378$. Hence, MAR-ICBW is a far superior counter measure to multi-flow attack.

4. REFERENCES

[1] J.F. Raymond, "Traffic Analysis: Protocols, Attacks, Design Issues, and Open Problems," *LECTURE NOTES IN COMPUTER SCIENCE*, pp. 10–29, 2001.

[2] Y. Zhang and V. Paxson, "Detecting stepping stones," in *USENIX Security Symposium*, Steven Bellovin and Greg Rose, Eds., Berkeley, CA, USA, Aug. 2000, pp. 171–184, USENIX Association.

[3] X. Wang and D. S. Reeves, "Robust correlation of encrypted attack traffic through stepping stones by manipulation of interpacket delays," in *ACM Conference on Computer and Communications Security*, Vijay Atluri, Ed., New York, NY, USA, 2003, pp. 20–29, ACM.

[4] Xinyuan Wang, Shiping Chen, and Sushil Jajodia, "Tracking anonymous peer-to-peer VoIP calls on the Internet," in *ACM Conference on Computer and Communications Security*, Catherine Meadows, Ed., New York, NY, USA, Nov. 2005, pp. 81–91, ACM.

[5] Y. Pyun, Y. Park, X. Wang, D. S. Reeves, and P. Ning, "Tracing traffic through intermediate hosts that repacketize flows," in *IEEE Conference on Computer Communications (INFOCOM)*, George Kesidis, Eytan Modiano, and R. Srikant, Eds., May 2007, pp. 634–642.

[6] X. Wang, S. Chen, and S. Jajodia, "Network flow watermarking attack on low-latency anonymous communication systems," In Pfitzmann and McDaniel [9], pp. 116–130.

[7] W. Yu, X. Fu, S. Graham, D.Xuan, and W. Zhao, "DSSS-based flow marking technique for invisible traceback," In Pfitzmann and McDaniel [9], pp. 18–32.

[8] N. Kiyavash, A. Houmansadr, and N. Borisov, "Multi-flow Attacks Against Network Flow Watermarking Schemes," in *Proceedings of the Usenix Security Symposium*, San Jose, CA, 2008.

[9] Birgit Pfitzmann and Patrick McDaniel, Eds., *IEEE Symposium on Security and Privacy*, May 2007.