# On the Feasibility of Real-Time Cyber Attack Attribution on the Internet

Xinyuan Wang, *Member, IEEE*

***Abstract*—The capability to reliably and accurately identify the attacker has long been believed as one of the most effective deterrents to an attack. Ideally, the attribution of cyber attack should be automated from the attack target all the way toward the attack source on the Internet in real-time. Real-time, network-wide attack attribution, however, is every challenging, and many people have doubted whether it is feasible to have practical attack attribution on the Internet.**

**In this paper, we look into the problem, challenges of real-time attack attribution on the Internet, and analyze what it takes to have the real-time attack attribution on the Internet. We show that it is indeed feasible and practical to attribute certain cyber attacks on the Internet in real-time. We build such a real-time attack attribution system upon the malware immunization and packet flow watermarking techniques we have developed. We demonstrate the unprecedented real-time attack attribution capability via live experiments on the Internet and Tor nodes all over the world.**

***Index Terms*—Attack attribution, attack traceback, attack response.**

## I. Introduction

Cyber attacks have become a serious threat to all computers and networks we are relying on daily. In 2015 alone, cyber attacks caused nearly 300 million records leaked and over $1 billion lost [1]. The cyber attack on Anthem insurance company [2] may have compromised sensitive information (e.g., name, SSN, address) of up to 80 million customers and employees.

Besides civilian targets, cyber attacks also aim at military and mission critical systems and networks. In recent cyber attack on U.S. Central Command's Twitter and YouTube accounts [3], [4], the pro-ISIS attackers claimed [4] "we broke into your networks and personal devices and know everything about you. You'll see no mercy infidels. ISIS is already here, we are in your PCs, in each military base." The recent cyber attack on the Office of Personnel Management (OPM) computer systems has compromised not only sensitive personal information (e.g., SSN) of roughly 21.5 million people from both inside and outside the government [5], but also irreplaceable personal biometric (e.g., fingerprint) information of 5.6 million people [6].

Recognizing its serious threat to the national interest of the United States, "the U.S. Director of National Intelligence ranks cyber crime as the No. 1 national security threat, ahead of terrorism, espionage and weapons of mass destruction" [7].

While existing cyber defense mechanisms, such as firewalls, intrusion detection systems (IDS), intrusion prevention systems (IPS), and malware protection systems, are very useful in protecting mission critical cyber infrastructures and global organizations, they fail to automatically address the root cause of all cyber attacks – the attackers or the perpetrators.

One of the greatest fears of all perpetrators is the risk of being caught. As shown in Figure 1, attackers seldom attack directly from their own hosts but rather launder the attack traffic through intermediate stepping stones (or proxies) throughout the world. Sophisticated attackers can even use the publicly available low-latency anonymity systems such as Tor, anonymizer.com to hide their true origin. Such traffic laundering makes the network based attack attribution one of the hardest problems in network security, and many people doubt whether it is technically feasible to track and attribute attacks across the Internet [8], [9], [10]. It is even harder to provide evidence of any attribution [11].

None of existing cyber defense mechanisms has the real-time attack attribution capability to automatically pinpoint the intrusion path and the attack source from where the attack was originated. Consequently, network based attackers have all the potential gains with virtually no risk of being caught. The 2016 Federal Cybersecurity Research and Development Strategic Plan [12] has listed (attack) attribution as one of the long-term (7-15 years) cybersecurity research and develop goal. In order to effectively repel and mitigate increasingly damaging cyber attacks from the network, it is critically important to have an automated, real-time attack attribution capability that helps hold the attackers accountable for their actions. Attacker would be reluctant to attack if the risk of being attributed and caught is high enough. Therefore, even an imperfect attack attribution helps repel the cyber attacks.

## II. Cyber Attack Attribution and Challenges

### A. Cyber Attacks

Successful cyber attacks are often results of out-of-box thinking and exploration. They are extremely versatile and often sophisticated. For example, some cyber attacks could use social engineering techniques such as phishing to harvest victim's credential. Other cyber attacks could use buffer overflow, heap overflow, integer overflow or format string exploits to break-in and control the target system. In term of infection path, some cyber attacks could use the network, files (e.g., copy or download), email, USB to infiltrate, and some other cyber attacks could use trojaned hardware (e.g., keyboard) to compromise the target system.

As more and more mission critical systems are connected to the network, most cyber attacks are coming from the network nowadays. We refer such attacks as network based attacks. The network based attacks could be either unidirectional or

Xinyuan Wang is with the Department of Computer Science, George Mason University, Fairfax, VA 22030, USA. email: xwangc@gmu.edu

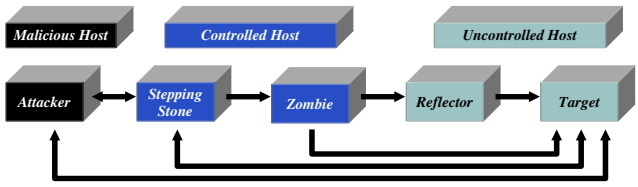


Fig. 1. The Overall Model of the Network Based Attacks

bidirectional. A unidirectional attack only has attack traffic from the attacker to the attack target, and it could freely spoof the source IP address of the attack packets as it does not have any return traffic from the attack target to the attacker. A typical example of unidirectional attacks is denial of service attack such as SYN-flood attack. While a unidirectional attack could disrupt or disable the target, it is not be able to exfiltrate any information from the attack target. A bidirectional attack, on the other hand, involves bidirectional traffic between the attacker and the attack target, and it usually exfiltrates information (e.g., data breach) from the attack target. In bidirectional attacks, the exfiltration and infiltration could happen at either the same time or different time. The bidirectional nature makes it almost impossible for the bidirectional attack to spoof the source IP address of the attack packet.

In this work, we focus on the bidirectional attacks that are originated and conducted from the Internet. Figure 1 illustrates the overall model of the network based attacks. Based on the roles exhibited in the network based attack, the intermediate nodes along the attack path could be classified into three categories: 1) stepping-stone, 2) zombie, and 3) reflector. A stepping stone is a network node controlled (e.g., compromised or rented) by the attacker that functions as a bidirectional conduit for the attack infiltration and exfiltration traffic. A stepping stone supports real-time bidirectional communication and it usually introduces very small delay. A zombie is a network node controlled (e.g., compromised or rented) by the attacker that is used as an attack launching point when triggered by the attacker. The trigger of the attack could be some special packet sent by the attacker to the zombie or a Trojan or logic bomb previously planted by the attacker into the zombie. A reflector is a network node not necessarily controlled by the attacker but somehow has been tricked into being a part of an attack in an innocent manner that is consistent with its normal operation.

In Figure 1, the double arrowed line represents a bidirectional traffic flow and the single arrowed line denotes a unidirectional traffic flow. Generally, zombie and reflector can only be used in unidirectional attack (i.e. denial of service attack). A stepping stone, however, could be used in both bidirectional and unidirectional network based attacks.

### B. Cyber Attack Attribution

Attack attribution generally refers to determining the identity or location of an attacker or an attacker's intermediary. In cyber space, attack attribution seeks to determine

1) from where and via what path the cyber attack has been launched.
2) cyber attack's entry point to the target network and system.
3) all the compromised nodes within the protected network that have involved the cyber attack.
4) who is responsible for the cyber attack.

Ideally, we want to find out the exact source (e.g., IP address, geolocation) from where the cyber attack has been launched. This helps us to determine who is ultimately responsible for the cyber attack. It is also important to figure out the path the attack has taken. Specifically, it is critical to identify the attack's entry point to the target network and system so that we can reinforce the mission critical network and system to prevent future penetration. Given the multiple layers of defense we are using, successful cyber attack usually needs to compromise multiple nodes within the protected network before it could attack the final target. Therefore, it is critically important to identify those compromised nodes within the mission critical network before we can eliminate the foothold of the cyber attack.

### C. Challenges in Cyber Attack Attribution

*1) Reliably Detecting the Cyber Attack First:* Before any cyber attack attribution can be done, we must be able to reliably detect the cyber attack first. Most existing intrusion detection systems (IDS) have detection false positives and false negatives. Therefore, whenever an IDS raises an alarm, there is a chance that the alarm is false. What really matters here is the conditional probability that the alarm is true when there is an alarm. This depends on not only the detection false positive rate (FPR) but also the base rate of the cyber attack or intrusion (the probability that the cyber attack or intrusion happens). In reality, cyber attack is a rare event in that most cyber events are benign.

Let $I$ represent the cyber attack or intrusion, and $T$ represent the detection by the IDS. Assume the cyber attack base rate $\Pr(I) = \frac{1}{100{,}000}$, and we have a very accurate IDS with 99% intrusion detection rate ($\Pr(T|I)$) and only 1% detection false positive rate ($\Pr(T|\neg I)$). Under these conditions, when the IDS raises an alarm, the probability that the alarm really reports a true attack is

$$\begin{aligned} & \Pr(I|T) \\ = \; & \frac{\Pr(I)\Pr(T|I)}{\Pr(I)\Pr(T|I) + \Pr(\neg I)\Pr(T|\neg I)} \\ = \; & \frac{\Pr(I)\Pr(T|I)}{\Pr(I)\Pr(T|I) + (1-\Pr(I))\Pr(T|\neg I)} \\ = \; & \frac{\frac{1}{100{,}000} \times 0.99}{\frac{1}{100{,}000} \times 0.99 + (1 - \frac{1}{100{,}000}) \times 0.01} \\ < \; & 0.1\% \end{aligned} \tag{1}$$

Therefore, when a seemly very effective IDS (with a 99% detection rate and a 1% false positive rate) reports an alarm, the chance that the alarm corresponds to any real attack is actually less than 0.1%! In the context of cyber attack attribution, such a low $\Pr(I|T)$ would simply cause a "false start" of any attack attribution.

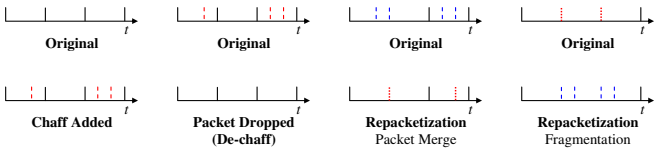


Fig. 2. Intra-flow transformations

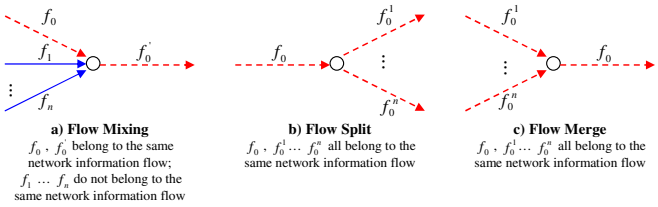


Fig. 3. Inter-flow transformations

The above example demonstrates the "base rate fallacy" [13] in any detection problem. Specifically, a low base rate $\Pr(I)$ would require an extremely low attack detection false positive rate $\Pr(T|\neg I)$ in order to have reasonably high confidence of any alarm raised by the IDS: $\Pr(I|T)$.

Since the cyber attack may deliberately remove the evidence of the attack, certain attribution information can only be collected live while the cyber attack is going on. Therefore, effective cyber attack attribution requires true real-time intrusion detection with exceedingly low detection false positive rate.

*2) Challenges in Attributing Attacks on the Internet:* Now assume we can reliably detect the cyber attack in true real-time and we know exactly when to start attack attribution. Attack attribution needs to reliably trace the detected attack across the network as the attacker seldom attack directly from his/her own host, but use all kinds of techniques to conceal his/her true origin and identity. Specifically, an attacker could

- encrypt the attack traffic and launder it through a number of intermediate nodes such as stepping stones, proxies.
- use low-latency anonymity systems (e.g., Tor [14], Anonymizer [15]) to anonymize the attack traffic.
- add bogus packets to the attack flow or drop random (useless) packets from the attack flow; repacketize the attack packets by combining several smaller packets into one larger packet or fragment one packet into several smaller ones as shown in Figure 2.
- mix the attack traffic with other irrelevant traffic or split the attack flow into multiple sub flows and merge the sub flows later as shown in Figure 3.

As a result, the attack target only sees the attack traffic coming from some stepping stone rather than the true source of the attack. The use of encryption and all kinds of transformations (as shown in Figure 2 and Figure 3) make the attack flow look very different when it across each stepping stone or proxy. All these make real-time attack attribution on the Internet one of the hardest problems in cyber security [16], [11], [9].

## III. Real-Time Attack Attribution based on Dynamically Assigned Tag

### A. The Feasibility of Real-Time Attack Attribution

Based on the analysis in section II, any network-wide, real-time attack attribution requires:

- true real-time attack or intrusion detection with close to zero false positive rate.
- true real-time network-wide attack traceback capability that is robust against various packet flow transformations (e.g., encryption, flow mixing, adding cover traffic).

The key to detecting cyber attacks is the capability of distinguishing the actions by the attacks from all the rest. If we can tag all the legitimate actions of the protected system, and make sure no attacker knows the secret tag, we could reliably detect the attack in real-time by catching its first action without proper tag.

In any bidirectional network based attack, besides the attack traffic, there exists return traffic from the attack target all the way to the attacker no matter how many intermediate nodes it passes. Therefore, if we can somehow transparently tag the return traffic of the attack at the attack target and make sure the tag survives various packet flow transformations such as encryption, flow mixing, adding chaff, we could track where the tagged return traffic of the attack goes and figure out the all the intermediate nodes, and the path the attack traffic has gone through. Given enough monitoring coverage, we can eventually pinpoint the source from where the attack has been launched.

### B. Real-Time Detection of Control Flow Hijacking Attacks based on Dynamically Assigned Sense of Self

Our natural immune system has been shown to be very effective in protecting our body from almost endless variations of pathogens based on the self-nonself discrimination capability that can distinguish our own cells (i.e., "self") from all others (i.e., "non-self"). If we view the uninfected computer system as "self" and malwares as "nonself", then protecting the uninfected computer system from malware attacks is very similar to protecting our body from invading pathogens from the perspective of self-nonself discrimination. If we can effectively and efficiently distinguish the "self" actions of the uninfected computer system from the "nonself" actions of malwares and attacks, we can detect the attack in real-time.

Inspired by the self-nonself discrimination in our natural immune system, we have developed an active method to distinguish "self" actions from "nonself" based on dynamically assigned sense of self [17]. As shown in Figure 4, our approach dynamically assigns a unique and secret tag to all the system calls invoked by the immunized program. Such an dynamically assigned secret tag forms *a dynamically assigned sense of self* of the immunized program. Since the dynamically assigned secret tag is unknown to the adversary, none of the system calls invoked by any malware or attack could have the correct secret tag. Therefore, the dynamically assigned secret tag (i.e., sense of self) enables us to effectively and efficiently distinguish the "self" system calls invoked by the immunized program from the "nonself" system calls invoked by the malware or attack.

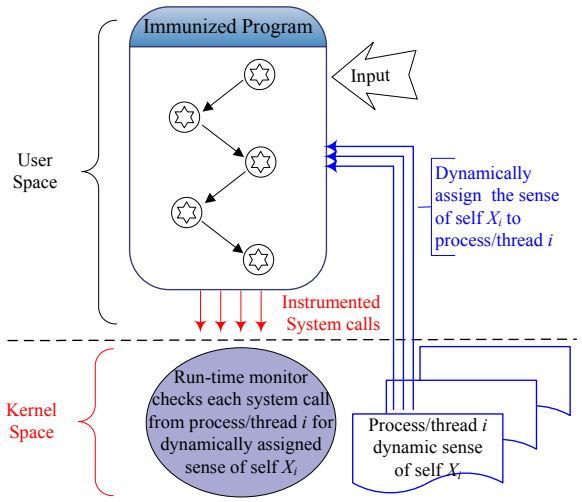


Fig. 4. Real-time attack detection and response based on dynamically assigned sense of self

Unlike all existing passive models of self, our active model of self is independent from the inherent complexity of the protected program or system, it has the following unprecedented capabilities:

- **True real-time attack detection and response:** Our system can detect and block the first (and all the rest) nonself system call invoked by any malware or attack in real-time. This enables true real-time response to the detected attack such as real-time attack attribution, forensics and recovery that were not possible before.
- **Effective without attack signature:** Our attack detection system based on dynamically assigned sense of self is effective against various control flow hijacking attacks (e.g., buffer overflow, return-to-libc, return-oriented exploits, jump-oriented exploits). Since our attack detection does not require any specific knowledge or signature of the attack, it can be effective against new, previously unknown attacks. This is in contrast to almost all existing deployed intrusion detection (e.g., Snort) and anti-virus systems which require periodic update of attack or malware signatures.
- **No false positive in attack detection:** By dynamically assign the unique and secret tag to all the system calls invoked by the to be protected program or system, our real-time attack detection will never falsely accuse any self system call to be nonself.
- **No need for training or re-alignment ever:** As the sense of self is not learned but dynamically assigned.

### *C. Real-Time Network-Wide Attack Tracing based on Dynamically Assigned Watermark*

Once we have detected a bidirectional network based attack at the target in real-time, we want to tag potentially anonymized and encrypted return (or backward) traffic at the attack target. This would allow us to track the tagged return traffic all the way back to the attack source.

With such a goal in mind, we have developed a novel method to embed a unique watermark (i.e., bit string) into the inter-packet timing of the return traffic of the attack by slightly adjusting the timing of selected packets [18]. Since our watermark is encoded in the inter-packet timing of the packet flow, it does not use or change the packet content. This enables us to watermark (i.e., tag) anonymized and encrypted packet flows. If the embedded watermark is unique enough and robust enough, the watermarked return traffic of the attack could be effectively identified and tracked across the Internet.

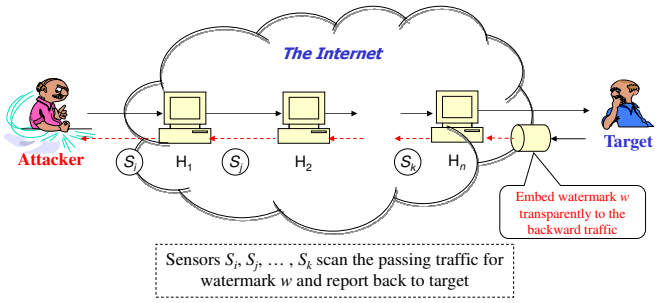


Fig. 5. Network wide attack tracing based on dynamically assigned watermark

Figure 5 illustrate the real-time, network wide attack tracing based on such transparent flow watermarking. The watermark engine is a special router that will embed a specified watermark to a specified packet flow. A sensor is a network device that checks the inter-packet timing of the passing packet flows for specified watermark. Once the real-time attack detection system (described in section III-B and [17]) at the protected host (i.e., attack target) detects a non-self system call, it reports the attack to the central control in real-time. Based on the reported attack information, the central control initiates and coordinates the network wide attack attribution in real-time:

- it instructs the watermark engine to watermark specified return traffic with specified watermark;
- it asks all the sensors across the Internet to scan the passing packet flows for the specified watermark and report back to the central control whenever any packet flow has been found to have the specified watermark.
- it congregate all the reports received from the sensors to construct the attack path toward the attack source.

Note, the network wide attack attribution and traceback require sensors deployed in the network, and the more sensors deployed the better. However, even if the network based attack passes areas that have no sensor deployed, our flow watermark based attack attribution and traceback can still work. For example, an hypothetical attack originated from California may first launder through three stepping stones in Europe (Paris, Berlin and Rome respectively), then another stepping stone at Virgina before it attacks the final target at New York. If we have sensors deployed in USA but not in Europe, we can find these watermarked flows between 1) New York and Virginia; 2) Virginia and Rome; 3) Paris and California. Because we don't have sensors in Europe, we don't know if there is any intermediate node between Paris and Rome. We do know, however, the attack has laundered from California to Paris, Rome, Virginia before attacking the target at New York. So we can still find the attack source in California and all those stepping stones that are within the range of the sensors.

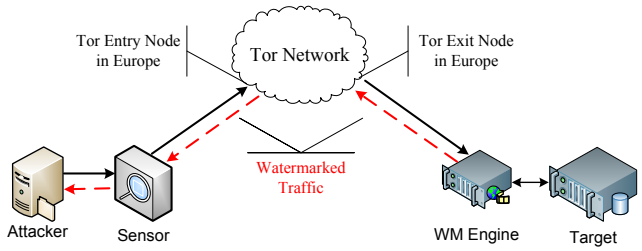


Fig. 6. Experimental setup for real-time, network wide attack attribution across nodes in USA and Tor nodes in Europe

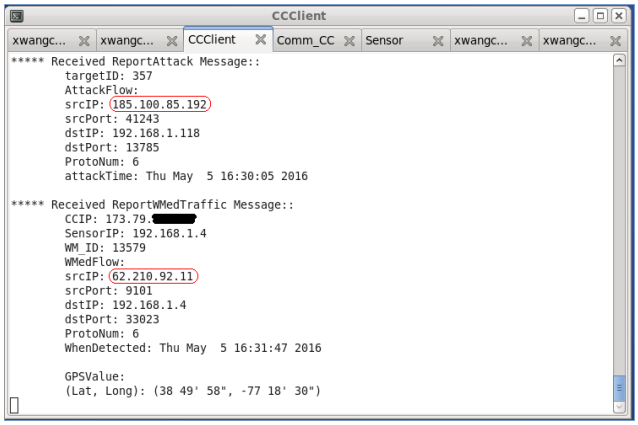


Fig. 7. Real-time, network wide attack attribution experimental result

## IV. Empirical Validation

We have implemented a proof-of-concept prototype of the real-time attack detection system and the network wide attack attribution and tracing system in Linux, and we have conducted real-time, live experiments on the Internet with Tor.

Figure 6 shows the experimental setup. We set up an attack target and a watermark engine in Virgina, and made sure all the traffic to and from the attack target was routed through the watermark engine. We used another machine in Virginia as the attacking machine, and had a sensor deployed close to the attacking machine. To demonstrate the unprecedented real-time, network wide attack attribution capability across the Internet, we chose to use Tor to anonymize the attack traffic we were trying to track. Specifically, We launched simulated attacks from the attacking machine in Virginia and routed the attack traffics through three different Tor nodes before they reached the target.

We have done two sets of experiments with two different Tor circuit. The first set of experiments have used the Tor entry node with IP address 62.210.92.11 in Paris, France, and the Tor exit node with IP address 185.100.85.192 in Bucharest, Romania. The second set of experiments have used the Tor entry node with IP address 195.154.107.23 in Paris, France, and the Tor exit node with IP address 37.123.130.176 in Oslo , Norway. In both set of experiments, we used the watermark engine to watermark the return traffic with a 20-bit watermark, 350ms maximum encoding delay, 500ms time interval and redundancy numbers 1,2,3 and 4. The watermarked return traffic was routed through the Tor exit node, the (unknown) Tor intermediate node and the Tor entry node before it reached back to the attacking machine. The sensor close to the attacking machine decoded the watermarked traffic that had been anonymized by Tor and reported back to the central control the flow that has the most watermark bits matched. In this case, the sensor close to the attacking machine can see the traffic between the attacking machine and the Tor entry node (e.g., 62.210.92.11 in Paris), the attack detection system at the target can see the traffic between the Tor exist node (e.g., 185.100.85.192 in Bucharest) and the target. While our attack attribution system did not see the Tor intermediate node in our experiments, it could automatically figure out the attacking machine in real-time as shown in Figure 7.

| Redundancy | No. of Matched WM Bits | Collision Probability |
|---|---|---|
| 1 | 15.5 | $< 1.479 \times 10^{-2}$ |
| 2 | 19 | $1.907 \times 10^{-5}$ |
| 3 | 19 | $1.907 \times 10^{-5}$ |
| 4 | 20 | $9.537 \times 10^{-7}$ |

TABLE I
Decoding results of 20-bit watermark from watermarked traffic anonymized by Tor entry node 62.210.92.11 in Paris, France and Tor exit node 185.100.85.192 in Bucharest, Romania

| Redundancy | No. of Matched WM Bits | Collision Probability |
|---|---|---|
| 1 | 17 | $1.087 \times 10^{-3}$ |
| 2 | 19 | $1.907 \times 10^{-5}$ |
| 3 | 19.5 | $< 1.907 \times 10^{-5}$ |
| 4 | 19.5 | $< 1.907 \times 10^{-5}$ |

TABLE II
Decoding results of 20-bit watermark from watermarked traffic anonymized by Tor Entry Node 195.154.107.23 in Paris, France and Tor Exit Node 37.123.130.176 in Oslo , Norway

The confidence of the network flow tagging depends on the length of the watermark and the number of bits matched. We experimented with different redundancy numbers, which require different durations of active traffic to be effective. We repeated experiments under each combination twice. Table I shows the average number of matched bits of the decoded watermark under different redundancy number in the first set experiments. We can get 20 out of 20 bits matched when use redundancy 4. This gives us less than $10^{-6}$ collision probability (i.e., false positive) with 80 seconds worth of traffic. With redundancy number 2, we only need 40 seconds worth of traffic, and we can still get 19 out of 20 bits matched, which gives us $1.907 * 10^{-5}$ collision probability. As shown in Table II, the second set of experiments give us similar but slightly different results.

In summary, our real-time, network wide attack attribution system only needs 40 seconds worth of active traffic to reliably attribute attack flows across USA and Tor nodes all over Europe with very low ($< 1.907 * 10^{-5}$) collision probability.

## V. Related Works

There have been substantial research works on how to trace attack packets with spoofed source IP address. Notably, Savage et al first proposed PPM IP traceback approach [19], and Snoeren et al first proposed logging based IP traceback approach [20]. However, such traceback approaches are not

able to track bidirectional attacks that are laundered through stepping stones.

A number of approaches (e.g., [21], [22]) have been proposed to correlate unencrypted packet flows across stepping stones. To correlated encrypted packet flows across stepping stones, researchers proposed using inter-packet timing characteristics to correlate (e.g., [23] ). To resist the active timing perturbation by the adversary, researchers proposed active timing based approaches (e.g., [24], [25], [26], [27], [28], [18]) to deliberately encode watermark into the inter-packet timing of the packet flow. We build our real-time attack attribution upon our interval based flow watermarking scheme [18] which has been proved to be robust against various flow transformations such as encryption, flow mixing, flow splitting, adding chaff and timing perturbation.

There is a large body of work in intrusion detection [13]. To the best of our knowledge, our active malware immunization approach [17] is the only one that can catch the first nonself system call with no false positive in true real-time. Such a capability is a key enabler of true real-time attack attribution.

## VI. Conclusions

In this work, we have examined the problem and technical challenges of real-time, network wide attack attribution, and have analyzed what it takes to have the real-time attack attribution on the Internet. We are the first to show that by combination of novel malware immunization techniques and novel network flow watermarking techniques, it is indeed feasible to track bidirectional network based attacks on the Internet in real-time.

We have implemented a proof-of-concept prototype of such real-time network wide attack attribution system in Linux, and have conducted live experiments on the Internet. Our experimental results have validated the unprecedented real-time attack attribution capability across nodes in USA and various Tor nodes in Europe. Specifically, our attack attribution system needs as little as 40 seconds worth of active traffic to effectively track traffic anonymized by various Tor nodes across Europe with 99.998% ($1 - 1.907 \times 10^{-5}$) confidence.

## References


[1] "9 Worst Cloud Security Threats," http://www.informationweek.com/cloud/infrastructure-as-a-service/9-worst-cloud-security-threats/d/d-id/1114085.

[2] E. Weise, "Massive Breach at Health Care Company Anthem Inc." http://www.usatoday.com/story/tech/2015/02/04/health-care-anthem-hacked/22900925/.

[3] J. Constine, "ISIS "Cyber Caliphate" Hacks U.S. Military Command Accounts," http://techcrunch.com/2015/01/12/cyber-caliphate/.

[4] J. Tanner, "Pro-ISIS Hackers Take Control of US Central Command Twitter Account," http://pix11.com/2015/01/12/hackers-take-control-of-us-military-centcom-twitter-account/.

[5] J. Sciutto, "OPM Government Data Breach Impacted 21.5 Million," http://www.cnn.com/2015/07/09/politics/office-of-personnel-management-data-breach-20-million/.

[6] A. Peterson, "OPM says 5.6 million fingerprints stolen in cyberattack, five times as many as previously thought," https://www.washingtonpost.com/news/the-switch/wp/2015/09/23/opm-now-says-more-than-five-million-fingerprints-compromised-in-breaches/.

[7] I. Bremmer, "These 5 Facts Explain the Threat of Cyber Warfare," http://time.com/3928086/these-5-facts-explain-the-threat-of-cyber-warfare/.

[8] L. Greenemeier, "Seeking Address: Why Cyber Attacks Are So Difficult to Trace Back to Hackers," http://www.scientificamerican.com/article/tracking-cyber-hackers/.

[9] M. J. Ranum, "Attribution is Hard," https://www.tenable.com/blog/attribution-is-hard-part-1.

[10] "Planning for the Future of Cyber Attack Attribution," House Hearing, 111 Congress. Serial No. 111-105.

[11] B. Schneier, "Attack Attribution and Cyber Conflict," https://www.schneier.com/blog/archives/2015/03/attack_attribut_1.html.

[12] "Federal Cybersecurity Research and Development Strategic Plan," https://www.whitehouse.gov/sites/whitehouse.gov/files/documents/2016_Federal_Cybersecurity_Research_and_Development_Stratgeic_Plan.pdf.

[13] S. Axelsson, "The Base-Rate Fallacy and Its Implications for the Difficulty of Intrusion Detection," in *Proceedings of the 6th ACM Conference on Computer and Communications Security (CCS 1999)*. ACM, November 1999, pp. 1–7.

[14] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The Second-Generation Onion Routing," in *In Proceedings of the 13th USENIX Security Symposium*. San Diego, CA, USA: USENIX, August 2004, pp. 303–320.

[15] "The Anonymizer," http://anonymizer.com.

[16] B. Schneier, "Attack Attribution in Cyberspace," https://www.schneier.com/blog/archives/2015/01/attack_attribut.html.

[17] X. Wang and X. Jiang, "Artificial Malware Immunization based on Dynamically Assigned Sense of Self," in *Proceedings of the 13th Information Security Conference (ISC 2010)*, October 2010.

[18] X. Wang, S. Chen, and S. Jajodia, "Network Flow Watermarking Attack on Low-Latency Anonymous Communication Systems," in *Proceedings of the 2007 IEEE Symposium on Security & Privacy (S&P 2007)*, Oakland, CA, May 2007, pp. 116–130.

[19] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Practical Network Support for IP Traceback," in *In Proceedings of the 2000 ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM 2000)*. ACM, September 2000, pp. 295–306.

[20] A. Snoeren and C. Patridge, "Hash-based IP Traceback," in *In Proceedings of the 2001 ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM 2001)*. ACM, September 2001, pp. 3–14.

[21] X. Wang, D. S. Reeves, S. F. Wu, and J. Yuill, "Sleepy Watermark Tracing: An Active Network-Based Intrusion Response Framework," in *Proceedings of the 16th Internatinal Conference on Information Security (IFIP/Sec 2001)*. Kluwer Academic Publishers, June 2001, pp. 369–384.

[22] B. Carrier and C. Shields, "A Recursive Session Token Protocol For Use in Computer Forensics and TCP Traceback," in *Proceedings of Proceedings of the 21th Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom 2002)*, April 2002, pp. 1540 – 1546.

[23] X. Wang, D. S. Reeves, and S. F. Wu, "Inter-Packet Delay based Correlation for Tracing Encrypted Connnections through Stepping Stones," in *Proceedings of the 7th European Symposium on Research in Computer Security (ESORICS 2002)*, ser. LNCS-2502. Springer-Verlag, October 2002, pp. 244–263.

[24] X. Wang and D. S. Reeves, "Robust Correlation of Encrypted Attack Traffic through Stepping Stones by Manipulating of Interpackets Delays," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS 2003)*. ACM, October 2003, pp. 20–29.

[25] P. Peng, P. Ning, D. S. Reeves, and X. Wang, "Active Timing Based Correlation of Perturbed Traffic Flow with Chaff," in *Proceedings of the 2nd International Workshop on Security in Distributed Computing Systems (SDCS-2005)*, June 2005.

[26] X. Wang, S. Chen, and S. Jajodia, "Tracking Anonymous Peer-to-Peer VoIP Calls on the Internet," in *Proceedings of the 12th ACM Conference on Computer and Communications Security (CCS 2005)*. Alexandra, VA: ACM, November 2005, pp. 81–91.

[27] Y. J. Pyun, Y. H. Park, X. Wang, D. S. Reeves, and P. Ning, "Tracing Traffic through Intermediate Hosts that Repacketize Flows," in *Proceedings of the 26th Annual IEEE Conference on Computer Communications (Infocom 2007)*, May 2007.

[28] W. Yu, X. Fu, S. Graham, D. Xuan, and W. Zhao, "DSSS-Based Flow Marking Technique for Invisible Traceback," in *Proceedings of the 2007 IEEE Symposium on Security & Privacy (S&P 2007)*, Oakland, CA, May 2007, pp. 18–32.