Practical Vulnerabilities of the Tor Anonymity Network

Paul Syverson
Center for High Assurance Computer Systems
U.S. Naval Research Laboratory

Abstract

Onion routing is a technology designed at the U.S. Naval Research Laboratory to protect the security and privacy of network communications. In particular, Tor, the current widely-used onion routing system, was originally designed to protect intelligence gathering from open sources and to otherwise protect military communications over insecure or public networks, but it is also used by human rights workers, law enforcement officers, abuse victims, ordinary citizens, corporations, journalists, and others. In this article our focus is less on what Tor currently does for its various users and more on what it does not do. Use of Tor for law enforcement and the national security applications that motivated it faces more significant adversaries than most other uses. We discuss some of the types of threats against which Tor currently offers only limited protection and the impacts of these on all classes of users, but especially on those most likely to confront them.

We have designed and built the Tor anonymity network [3] to secure cyberspace and empower cybercitizens. It is thus squarely in the middle of this volume's concerns. But in law enforcement, the first thought that often comes to mind when one says "anonymity" is of a roadblock against pursuing the source of an attack or other crime. Although this is sometimes the first image that comes to mind, it is not generally the first encounter law enforcers have with anonymity. Typically law enforcers themselves have begun using anonymity long before they observe any criminal activity, and they may even use it to prevent a crime from occurring at all.

As a simple mundane example of anonymity technology used by law enforcers, consider unmarked vehicles. These are used precisely to avoid obvious distinguishability from other cars around them. This might be to avoid alerting a criminal to the presence or location of law enforcement, or to help protect someone being discretely transported, or for various other reasons. Anonymity is an essential part of law enforcement. Note that unmarked cars are effective, not just because unmarked law-enforcement vehicles are not immediately identifiable as law-enforcement vehicles, but also because most vehicles used by others are similarly anonymous. You can't be anonymous by yourself, and the anonymity protection on which law enforcement depends only works if others have it as well. We will return to this point below. Unmarked vehicles are of course just one example. The same applies equally to crime prevention programs, witness protection, anonymous tip lines, and so on. Although there are many important anonymity technologies, our focus herein is anonymous Internet communication.

Tor protects your anonymity by bouncing your Internet traffic over an unpredictable route comprised of volunteer-run traffic relays all over the world. Tor builds a cryptographic circuit over three relays, and the cryptography prevents each relay from knowing about the other parts of the circuit it does not talk to directly. Only the Tor software on your computer knows about all three relays. Early versions of onion routing laid the cryptographic circuit using an onion, a data structure comprised entirely of layers (one for each hop in the circuit) with effectively nothing at the center. This is what gave onion routing—and Tor (Tor's onion routing)—its name; although circuits in Tor are built in a slightly different way for improved security. What is not different is that Tor gets its protection from the difficulty an adversary has observing the whole circuit. If you are browsing the web over Tor, an adversary might see some of your encrypted traffic go by if he can watch it enter the first relay in the circuit. Similarly, he might see some encrypted traffic headed to you if he can watch the server's response enter the last relay in the Tor circuit. But, unless he can watch both places at once, he will not be able to associate you with the website you are visiting. A basic picture of a Tor circuit is shown in Figure 1.

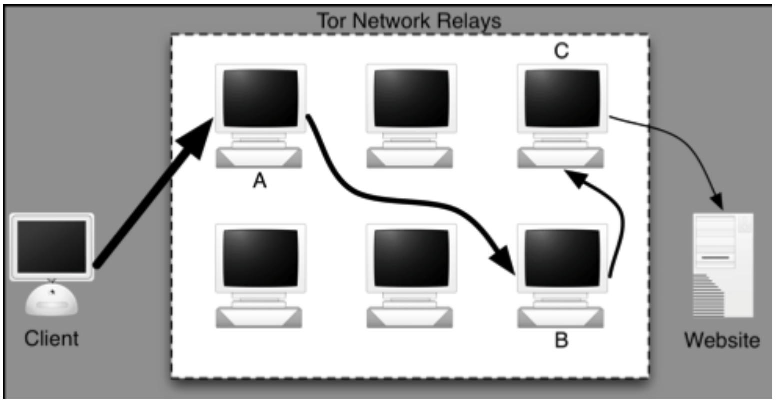

Figure 1: Web browsing through a Tor circuit.

Unfortunately the bad guys have a significant advantage over the good guys when it comes to keeping their communications anonymous. For example, if they want to make phone calls with guaranteed anonymity, they can just steal a cell phone off of a restaurant table and then toss it when done. This avenue is not available to those unwilling to break the law. Similarly for Internet communications, bad guys have easy access to cheap and plentiful compromised computers on botnets. As a bonus, their communication can masquerade as not even attempting to appear anonymous while misdirecting attention to the victims of a botnet compromise.

Networks like Tor do not provide this kind of anonymity, but they can help secure the communications of cybercitizens and those who strive to protect them. Because the public Tor network is comprised of thousands of relays in the network all over the world [5], it would be difficult for an adversary to watch all, or even most, of them. This protects ordinary users from revealing a profile of their online interests to common identity thieves, for example, but it is a computer security mechanism, not magic. And like all computer security mechanisms, Tor is most effective if those who use it understand what protections they have and what protections they do not have.

Elsewhere in this volume, Andrew Lewman describes how Tor empowers law enforcers and many kinds of cybercitizens and how Tor makes the net a safer place by protecting them against traffic analysis. In this article we will talk about some areas where protecting communications against traffic analysis still has a ways to go.

1 The ends of watching the ends

Onion routing cryptographically changes the appearance of the traffic as it passes along the circuit. But some features of the communication are still apparent to an observer. Consequently if an adversary does not see both ends of a Tor circuit, anonymity is typically preserved. If an adversary can see both ends of a Tor circuit, he can trivially correlate who is talking to whom over that circuit. This is thus generally known as an *end-to-end correlation attack*.

There are many ways to correlate the endpoints of Tor communication. Since Tor is used for things like web surfing and chatting, communication must be *low-latency*: anything entering the circuit at one end needs to pop out at the other end pretty quickly. Nobody wants to wait several seconds, let alone several minutes, for a web page to load or to receive the next line in a chat. Besides providing an unacceptably bad user experience, communication protocols for these applications often simply fail with delays of that magnitude.

This means that an adversary who can see both ends of the Tor circuit, for example because he controls both the first and last relay in the circuit, can simply see the same unique pattern of communication popping out of one end of the circuit that he saw entering at the other end. Just passively watching has long been shown to be adequate to guarantee identification with no false positives [18]. In fact, it is not even necessary to wait for, say, a web request and response to flow over the Tor circuit, just watching the Tor circuit setup is enough [1]. There are many ways to do correlation. Instead of timing, the adversary can monitor volume by counting the amount of traffic that has passed over the circuits of each relay he is watching [20].

Why not just pad all the traffic between the user and say the second relay in the Tor circuit? That way all circuits will look the same to someone seeing their first relays. Padding and related proposals may have some effectiveness when the applications permit significant latency, such as email. But they have been extensively explored and have significant problems. First, all circuits that are supposed to be indistinguishable must be padded to the same level. This means predicting how much traffic will go over them or delaying traffic that exceeds the maximum rate of flow. Padding everything in this way would mean large additional load on a public shared network. Also, it is not enough to make the flows all the same. The circuits must all begin and end at exactly the same time.

But aside from the difficulty and the excessive overhead, this will not work anyway if the adversary is able to be at all active. For example, he can corrupt the traffic passing through a relay and then watch for corrupted traffic elsewhere in the circuit. This is sometimes called *bit stomping* or a *tagging attack* [2]. Vulnerability to such attacks have been a recognized limitation of onion routing since its origins in the mid-nineties [9]. If he wants to be more subtle still, the adversary does not have to corrupt the bits, just delay them a little and watch for the delay pattern at the other end. Research has shown that an adversary can add recognizable patterns to low-latency anonymous communication that collaborators will still be able to read even if the traffic is perturbed between them but that will be undetectable by others [22, 23].

There is at least one system design for resisting even active attacks [8]. This work has enough plausibility to merit further exploration. At this time, however, it seems unlikely that the practical overhead and usability issues can ever be adequately resolved. Even if they are, however, there is another problem.

Onion routing does not get its security from everyone being indistinguishable even though the attacker is seeing all of their communications. In this it differs from most anonymous communication models. To achieve that indistinguishability, it is necessary that the attacker cannot recognize the same messages even when he sees them elsewhere in the system and even if they are encrypted. And that requires synchronization of behavior by all who would be indistinguishable. For the above reasons, among others, this means that the set of individuals whose communication is indistinguishable cannot grow to the scale of Tor's hundreds of thousands of users. So even if they are hiding amongst each other better, they are hiding in a much smaller *anonymity set*.

Practical systems for anonymous communication based on trying to achieve indistinguishability do exist [14, 2]. Besides being high-latency and thus much less usable, they have never had more than a few hundred concurrent users hidden by a network of a dozen or so relays. This is fine if the goal is simply plausible deniability or for a limited application setting such as municipal voting. It is also fine if the adversary is only watching locally, for example if he is watching your wireless connection at a coffee shop and nothing else. But in intelligence gathering and law enforcement, this is typically inadequate. If the adversary has the incentives and resources of a nation state or of organized crime, then it is significant if he knows that, for example, someone of a hundred participants is definitely from law enforcement. Also the small anonymity set means that it is is now within the resource constraints of the adversary to closely scrutinize on and offline behavior of everyone identified as participating—at least for the kind of adversary faced in law enforcement and national security settings. So he can learn which participants were from which law enforcement organization even if this was not immediately identified by the communication. In this way, systems designed to resist global observers and active attackers are actually less secure than Tor [21].

End-to-end correlation is not the only way to attack Tor users, but we will not go into others. (See for example [15, 16, 6, 11].) Though some have been shown to be implementable, they are mostly much more obscure and less practical than the kinds of attacks we have already described. More importantly, they generally require a lot more groundwork, other assumptions about the network, or the success of different concurrent attacks, not to mention a technically sophisticated adversary. Compared to these, it requires very little technical sophistication to conduct the correlation attacks we have mentioned. Correlation attacks are also generally easy to scale with the resources of even an unsophisticated adversary: the more resources, the more relays he watches. Or since relays are run by volunteers, he can run some himself.

There is another attack that an adversary can do. It is not nearly as accurate as correlation, and it requires some planning and prediction. But it does not require much more sophistication than correlation. Suppose an adversary goes to a website such as `www.navy.mil` or `www.cnn.com` and records the pattern of bits that flow back and forth as he downloads the homepage or possibly even just the totals. If he later sees a Tor client exchanging the same pattern over a circuit, even though it is encrypted he can have some indication of what website the client is visiting. This is called *website fingerprinting* [10]. It assumes that the website fingerprint is sufficiently unique and has not changed since the adversary visited (or at least that the patterns and sizes have not changed even if the specific content has). It also assumes that the adversary already has the website fingerprint in his database. But if those are all true, then he only has to watch one end of the connection. For example, as mentioned before he could eavesdrop on your WiFi connection at a coffee shop. Tor already does much better against website fingerprinting than other web anonymizers because it sends all communication in uniform size chunks [13].

Still if an adversary is aware of one or more websites of interest that are relatively unique, he can use website fingerprinting as an indicator of those users that he might want to scrutinize further. And, if he is sophisticated, he can also use machine learning techniques to improve his accuracy. Fortunately for the aware user, it is easy to defeat website fingerprinting. By simultaneously visiting multiple sites over the same Tor circuit (which Tor allows automatically) any individual fingerprint becomes hard to recognize [19].

2 Link attackers

Most adversaries are not in a position to be directly sitting on a significant fraction of the Tor relays. But does an adversary even need to do that to watch both ends of a Tor connection? No, as it turns out. An adversary can actually observe from a small number of locations and still see much of the traffic on the Tor network.

Figure 1 is a fairly typical picture of communication over a Tor network—a graph of clients, network relays, and destinations with arrows representing the links between them. What such pictures ignore is the structure of the Internet that routes the traffic along those links.

The Internet is composed of thousands of independent networks of various sizes called *autonomous systems* (ASes). As traffic moves between a Tor client and a relay in the Tor network, it typically traverses multiple ASes. We have known for years [7, 17] that if the same AS appears on the path from the client to the anonymity network and from the anonymity network to the client's destination, such as a website, then an observer located at that AS can perform a correlation attack to identify the client and her destination. So it is possible to do a correlation attack from one location rather than requiring two. Figure 2 represents such an attack.

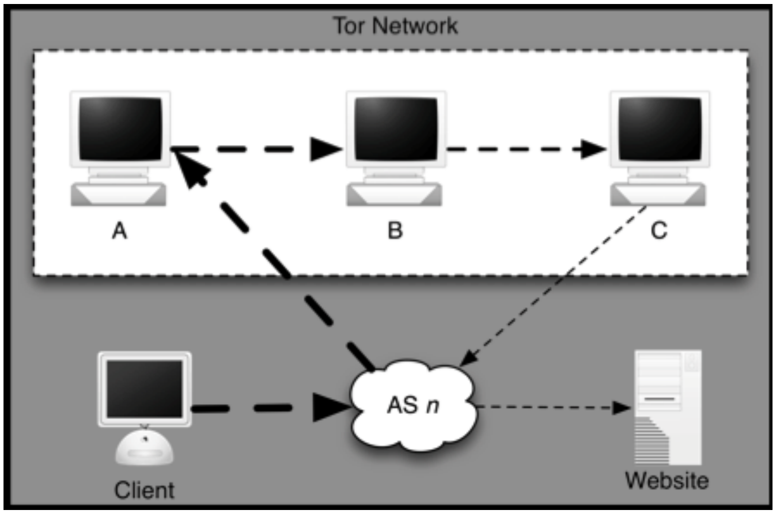

Figure 2: End-to-end correlation by a single AS observer.

In 2004, Feamster and Dingledine showed that the vulnerability to such an attack on the public Tor network was significant [7]. They looked at several client locations in the U.S. and several plausible destinations such as websites of CNN or Google and other destinations where users might want to protect their anonymity. For many of the client-destination pairs the probability that a single AS could observe both sides of the connection was over 50%. The mean overall probability was about 38%. But they mostly looked at client locations in the U.S., and at the time of their study the Tor network only contained thirty three relays. Since that time the network has grown by between one and two thousand relays.

We might hope that there would be a drop in vulnerability commensurate with the tremendous growth of the network. Intuition from the anonymity literature suggests that as the Tor network grows and volunteers operate relays all over the world, it becomes less likely for a single AS to be able to observe both ends of a connection. Intuition from communications networking is more muddy. On the one hand, there have been great increases in both the size and the geographic diversity of the Tor relay network. On the other hand, this might not be reflected in the number of network providers involved, and corporate and service consolidation could even imply a contraction of the distribution of ASes involved in carrying Tor traffic. In experiments we conducted about four years after the original study we did find a drop in the probability of a single AS-level observer seeing both ends of the connection. But, it only dropped from about 38% to about 22%, which is still quite high [4]. Furthermore, this was the average, but the drop was not uniform. For 12.5% of the client-destination pairs the chance of such a vulnerability actually increased.

Another issue is that the original Feamster-Dingledine study was only meant to show that such a vulnerability was realistic for plausible source-destination pairs. They did not try to determine what the distribution of actual traffic on the Tor network was. We also studied this question. We found 2251 client ASes and 4203 destination ASes. For both client and destination sides of circuits, less than two percent of all the ASes we recorded accounted for over half of the connections [4].

Tor already does some things by default to counter the chance of a single AS being able to observe both ends of a circuit. For one thing, Tor requires that the first and last relay cannot have network addresses from the same neighborhood. (It is a bit different than requiring that each relay reside in a different AS, but the effect is roughly comparable.) In simulations we showed that there is about an 18% chance that a single AS will be in the position to observe any Tor connection as of late 2008. There are things that can be done to reduce this to about 6%, with an increase to overall Tor communication and computation that is not trivial but also not prohibitive.

So far, we have only researched vulnerability to a single AS observer. But there is no reason an adversary will be present at only one AS. We should expect a nation state or large criminal adversary to have a much larger reach among both the Tor relays and the links between them.

3 Lessons for law enforcement and others

Lesson one is that Tor guards against traffic analysis not traffic confirmation. If there is reason to suspect that a client is talking to a destination over Tor, it is trivial to confirm this by watching them both. This applies to criminals being investigated, which is welcome news. But it also applies to those trying to protect us from them and to those being protected as well, which is not. As already noted, criminals have stronger mechanisms available to them, but Tor can help level the playing field.

Lesson two is that the current Tor network is not by itself sufficient to protect against all of the significant adversaries that may oppose law enforcement or defenders of national security. Tor is a volunteer run network. Tor picks routes so that relays in a single circuit will generally be from different network and geographic locations. But nothing prevents a large-scale adversary from breaking into or contributing many relays in many locations. An adversary that controls or observes a significant fraction of the network will observe a significant fraction of the network traffic. We have also ignored so far in this article the nonuniformity of network relays. Some carry a much larger fraction of traffic than others, which only exacerbates this vulnerability.

Lesson three is an extension of lesson two. For correlation vulnerability, the communication links between the relays matter as much as the relays themselves, if not more. An adversary that can observe even a small number of strategic Internet links can do end-to-end correlation on a large fraction of the traffic that passes over Tor.

Fortunately Tor is good enough for many of us. An abusive ex-spouse may be determined enough to monitor or break into a mutual friend's communication to try to find you, but is still unlikely to be able to track you if you communicate with that friend over Tor. See Andrew Lewman's article in this volume for several other examples from the mundane to the exotic. Things are a bit more difficult, however, for those who oppose more resourceful and well resourced adversaries.

Thus we need a network like Tor but robust against the vulnerabilities described above. Unfortunately, no such network now exists. Fortunately work on one is in the research stages. By Tor's very nature, there is no way to preclude adversary control of either a large fraction of relays or a significant number of Internet links connecting to them, or both—at least not if it is to continue enjoying and deserving the trust of its users. To resist significant adversaries, we might be tempted run our traffic over a trusted private network of relays under our control. This may obscure some things, but it will still reveal to our adversaries all the connections between the open Internet and this trusted network (hence to or from us). It also provides adversaries a relatively smaller target that will thus be easier to cover entirely. If, however, we make use of trusted and untrusted relays appropriately, we can get the benefits of trusted relays without the problems from their association to us [18, 12]. For example we can start a circuit at highly trusted relays, then move on to progressively less trusted, but also less associated relays. Of course we must also consider trust in Internet link elements such as ASes. Research in this area is still in early stages.

For now, there may be times when stealth operations can be constructed with no overt connection to the organization behind them. But this is not always feasible, and even if it is, defense in depth counsels reducing the risk of this relationship being discovered. It may help to initiate our communication over the Tor network via a collection of relays that we trust, that are part of the larger network, and that we have links to that are not publicly visible.

Especially this may help if we can hide our association with these relays. This may help, but its security implications have not yet been fully analyzed. This approach also will not help when we need to communicate to a trusted but associated destination, such as the home office, starting at an insecure location, such as a hotel room under observation of the adversary. We hope to have clearer advice about reducing your anonymity vulnerabilities soon, as well as better systems for reducing those vulnerabilities. For now, let us hope that a greater understanding of the vulnerabilities and risks of using Tor as it now exists will help empower the cybercitizens that rely on it.

References

[1] Kevin Bauer, Damon McCoy, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. Low-resource routing attacks against Tor. In Ting Yu, editor, *WPES'07: Proceedings of the 2007 ACM Workshop on Privacy in the Electronic Society*, pages 11–20. ACM Press, October 2007.

[2] George Danezis, Roger Dingledine, and Nick Mathewson. Mixminion: Design of a type III anonymous remailer protocol. In *Proceedings, 2003 IEEE Symposium on Security and Privacy*, pages 2–15, Berkeley, CA, May 2003. IEEE Computer Society.

[3] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *Proceedings of the 13th USENIX Security Symposium*, pages 303–319. USENIX Association, August 2004.

[4] Matthew Edman and Paul Syverson. AS-awareness in Tor path selection. In Somesh Jha, Angelos D. Keromytis, and Hao Chen, editors, *CCS'09: Proceedings of the 16th ACM Conference on Computer and Communications Security*, pages 380–389. ACM Press, 2009.

[5] Karsten Loesing et al. Tor metrics portal. `https://metrics.torproject.org/`, December 2010.

[6] Nathan S. Evans, Roger Dingledine, and Christian Grothoff. A practical congestion attack on Tor using long paths. In *Proceedings of the 18th USENIX Security Symposium*, pages 33–50, Montreal, Canada, August 2009. USENIX Association.

[7] Nick Feamster and Roger Dingledine. Location diversity in anonymity networks. In Sabrina De Capitani di Vimercati and Paul Syverson, editors, *WPES'04: Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society*, pages 66–76, Washington, DC, USA, October 2004. ACM Press.

[8] Joan Feigenbaum, Aaron Johnson, and Paul Syverson. Preventing active timing attacks in low-latency anonymous communication [extended

abstract]. In Mikhail J. Attallah and Nicholas J. Hopper, editors, *Privacy Enhancing Technologies: 10th International Symposium, PETS 2010*, pages 166–183. Springer-Verlag, LNCS 2605, July 2010.

[9] David M. Goldschlag, Michael G. Reed, and Paul F. Syverson. Hiding routing information. In Ross Anderson, editor, *Information Hiding: First International Workshop*, pages 137–150. Springer-Verlag, LNCS 1174, 1996.

[10] Andrew Hintz. Fingerprinting websites using traffic analysis. In Roger Dingledine and Paul Syverson, editors, *Privacy Enhancing Technologies: Second International Workshop, PET 2002*, pages 171–178, San Francisco, CA, USA, April 2002. Springer-Verlag, LNCS 2482.

[11] Nicholas Hopper, Eugene Y. Vasserman, and Eric Chan-Tin. How much anonymity does network latency leak? *ACM Transactions on Information and System Security*, 13(2):13–28, 2010. February.

[12] Aaron Johnson and Paul Syverson. More anonymous onion routing through trust. In *22nd IEEE Computer Security Foundations Symposium, CSF 2009*, pages 3–12, Port Jefferson, New York, USA, July 2009. IEEE Computer Society.

[13] Marc Liberatore, , and Brian Neil Levine. Inferring the source of encrypted HTTP connections. In Rebecca N. Wright, Sabrina De Capitani di Vimercati, and Vitaly Shmatikov, editors, *CCS'06: Proceedings of the 13th ACM Conference on Computer and Communications Security*, pages 255–263. ACM Press, 2006.

[14] Ulf Möller, Lance Cottrell, Peter Palfrader, and Len Sassaman. Mixmaster protocol - version 3. IETF Internet Draft, 2003.

[15] Steven J. Murdoch. Hot or not: Revealing hidden services by their clock skew. In Rebecca N. Wright, Sabrina De Capitani di Vimercati, and Vitaly Shmatikov, editors, *CCS'06: Proceedings of the 13th ACM Conference on Computer and Communications Security*, pages 27–36. ACM Press, 2006.

[16] Steven J. Murdoch and George Danezis. Low-cost traffic analysis of Tor. In *2005 IEEE Symposium on Security and Privacy,(IEEE S&P 2005) Proceedings*, pages 183–195. IEEE CS, May 2005.

[17] Steven J. Murdoch and Piotr Zieliński. Sampled traffic analysis by internet-exchange-level adversaries. In Nikita Borisov and Philippe Golle, editors, *Privacy Enhancing Technologies: 7th International Symposium, PET 2007*, pages 167–183. Springer-Verlag, LNCS 4776, 2007.

[18] Lasse Øverlier and Paul Syverson. Locating hidden servers. In *2006 IEEE Symposium on Security and Privacy (S& P 2006), Proceedings*, pages 100–114. IEEE CS, May 2006.

[19] Lexi Pimenidis and Dominik Herrmann. Contemporary profiling of web users. Presentation at the 27th Chaos Communication Congress (27C3), December 2010.

[20] Andrei Serjantov and Peter Sewell. Passive attack analysis for connection-based anonymity systems. In Einar Snekkenes and Dieter Gollmann, editors, *Computer Security – ESORICS 2003, 8th European Symposium on Research in Computer Security*, pages 141–159, Gjøvik, Norway, October 2003. Springer-Verlag, LNCS 2808.

[21] Paul Syverson. Why I'm not an entropist. In *Seventeenth International Workshop on Security Protocols*. Springer-Verlag, LNCS, 2009. Forthcoming.

[22] Xinyuan Wang, Shiping Chen, and Sushil Jajodia. Tracking anonymous peer-to-peer voip calls on the internet. In Catherine Meadows and Paul Syverson, editors, *CCS'05: Proceedings of the 12th ACM Conference on Computer and Communications Security*, pages 81–91. ACM Press, November 2005.

[23] Wei Yu, Xinwen Fu, Steve Graham, Dong Xuan, and Wei Zhao. DSSS-based flow marking technique for invisible traceback. In *2007 IEEE Symposium on Secuity and Privacy (SP'07)*, pages 18–32. IEEE Computer Society, May 2007.