Traffic correlation using netflows

by [arma](https://blog.torproject.org/users/arma) | November 14, 2014

People are starting to ask us about a recent [tech report](https://mice.cs.columbia.edu/getTechreport.php?techreportID=1545&format=pdf) from Sambuddho's group about how an attacker with access to many routers around the Internet could gather the netflow logs from these routers and match up Tor flows. It's great to see more research on traffic correlation attacks, especially on attacks that don't need to see the whole flow on each side. But it's also important to realize that traffic correlation attacks are not a new area.

This blog post aims to give you some background to get you up to speed on the topic.

First, you should read the first few paragraphs of the [One cell is enough to break Tor's anonymity](https://blog.torproject.org/blog/one-cell-enough) analysis:

First, remember the basics of how Tor provides anonymity. Tor clients [route their traffic](https://www.torproject.org/images/htw2.png) over several ([usually three](https://wiki.torproject.org/noreply/TheOnionRouter/TorFAQ#VariablePathLength)) relays, with the goal that no single relay gets to learn both where the user is (call her Alice) and what site she's reaching (call it Bob).

The Tor design [doesn't try to protect](https://www.torproject.org/svn/trunk/doc/design-paper/tor-design.html#subsec:threat-model) against an attacker who can see or measure both traffic going into the Tor network and also traffic coming out of the Tor network. That's because if you can see both flows, some [simple statistics](http://freehaven.net/anonbib/#danezis:pet2004) let you decide whether they match up.

Because we aim to let people browse the web, we can't afford the extra overhead and hours of additional delay that are used in high-latency mix networks like [Mixmaster](http://freehaven.net/anonbib/#mixmaster-spec) or [Mixminion](http://freehaven.net/anonbib/#minion-design) to slow this attack. That's why Tor's security is all about trying to decrease the chances that an adversary will end up in the right positions to see the traffic flows.

The way we generally explain it is that Tor tries to protect against **traffic analysis**, where an attacker tries to learn whom to investigate, but Tor can't protect against **traffic confirmation** (also known as **end-to-end correlation**), where an attacker tries to confirm a hypothesis by monitoring the right locations in the network and then doing the math.

And the math is really effective. There are simple packet counting attacks ([Passive Attack Analysis for Connection-Based Anonymity Systems](http://freehaven.net/anonbib/#SS03)) and moving window averages ([Timing Attacks in Low-Latency Mix-Based Systems](http://freehaven.net/anonbib/#timing-fc2004)), but the more recent stuff is [downright scary](http://www.lightbluetouchpaper.org/2007/05/28/sampled-traffic-analysis-by-internet-exchange-level-adversaries/), like Steven Murdoch's PET 2007 paper about achieving high confidence in a correlation attack despite seeing only 1 in 2000 packets on each side ([Sampled Traffic Analysis by Internet-Exchange-Level Adversaries](http://freehaven.net/anonbib/#murdoch-pet2007)).

Second, there's some further discussion about the efficacy of traffic correlation attacks at scale in the [Improving Tor's anonymity by changing guard parameters](https://blog.torproject.org/blog/improving-tors-anonymity-changing-guard-parameters) analysis:

Tariq's paper makes two simplifying assumptions when calling an attack successful [...] 2) He assumes that the end-to-end correlation attack (matching up the incoming flow to the outgoing flow) is instantaneous and perfect. [...] The second one ("how successful is the correlation attack at scale?" or maybe better, "how do the false positives in the correlation attack compare to the false negatives?") remains an open research question.

Researchers generally agree that given a handful of traffic flows, it's easy to match them up. But what about the millions of traffic flows we have now? What levels of false positives (algorithm says "match!" when it's wrong) are acceptable to this attacker? Are there some simple, not too burdensome, tricks we can do to drive up the false positives rates, even if we all agree that those tricks wouldn't work in the "just looking at a handful of flows" case?

More precisely, it's possible that correlation attacks don't scale well because as the number of Tor clients grows, the chance that the exit stream actually came from a different Tor client (not the one you're watching) grows. So the confidence in your match needs to grow along with that or your false positive rate will explode. The people who say that correlation attacks don't scale use phrases like "say your correlation attack is 99.9% accurate" when arguing it. The folks who think it does scale use phrases like "I can easily make my correlation attack arbitrarily accurate." My hope is that the reality is somewhere in between — correlation attacks in the current Tor network can probably be made plenty accurate, but perhaps with some simple design changes we can improve the situation.

The discussion of false positives is key to this new paper too: Sambuddho's paper mentions a false positive rate of 6%. That sounds like it means if you see a traffic flow at one side of the Tor network, and you have a set of 100000 flows on the other side and you're trying to find the match, then 6000 of those flows will look like a match. It's easy to see how at scale, this "base rate fallacy" problem could make the attack effectively useless.

And that high false positive rate is not at all surprising, since he is trying to capture only a summary of the flows at each side and then do the correlation using only those summaries. It would be neat (in a theoretical sense) to learn that it works, but it seems to me that there's a lot of work left here in showing that it would work in practice. It also seems likely that his definition of false positive rate and my use of it above don't line up completely: it would be great if somebody here could work on reconciling them.

For a possibly related case where a series of academic research papers misunderstood the base rate fallacy and came to bad conclusions, see [Mike's critique of website fingerprinting attacks](https://blog.torproject.org/blog/critique-website-traffic-fingerprinting-attacks) plus the [follow-up paper from CCS this year confirming that he's right](https://www.eecs.berkeley.edu/~sa499/papers/ccs-webfp-final.pdf).

I should also emphasize that whether this attack can be performed at all has to do with how much of the Internet the adversary is able to measure or control. This diversity question is a large and important one, with lots of attention already. See [more discussion here](https://blog.torproject.org/blog/improving-tors-anonymity-changing-guard-parameters#fix4).

In summary, it's great to see more research on traffic confirmation attacks, but a) traffic confirmation attacks are not a new area so don't freak out without actually reading the papers, and b) this particular one, while kind of neat, doesn't supercede all the previous papers.

(I should put in an addendum here for the people who are wondering if everything they read on the Internet in a given week is surely all tied together: we don't have any reason to think that this attack, or one like it, is related to the recent arrests of [a few dozen people](https://blog.torproject.org/blog/thoughts-and-concerns-about-operation-onymous) around the world. So far, all indications are that those arrests are best explained by bad opsec for a few of them, and then those few pointed to the others when they were questioned.)

[Edit: be sure to read Sambuddho's comment below, too. -RD]