Blocking-resistant Transport Evaluation Framework

Short Description

Tor's long-term viability as a censorship circumvention tools will depend on the ability of the protocol to bypass increasingly sophisticated blocking techniques such as Deep Packet Inspection (DPI). There is a proposal for adding blocking resistance to Tor through pluggable transports. However, how do we know which transport is best? I propose an evaluation framework for testing the effectiveness of transports against various blocking techniques using sample Tor traffic.

Proposal

a. What project would you like to work on? Use our ideas lists as a starting point or make up your own idea. Your proposal should include high-level descriptions of what you're going to do, with more details about the parts you expect to be tricky. Your proposal should also try to break down the project into tasks of a fairly fine granularity, and convince us you have a plan for finishing it. A timeline for what you will be doing throughout the summer is highly recommended.

1. Define the scope of the project. At a high level it will require tools for the collection of code and traffic samples and tools for running transport encoders and decoders as well as blocking algorithms on traffic samples and a method for evaluating the results of tests. There are many options for technologies on which to base these tools, so a key element of the project will be finding the options that are a best fit for the Tor community.

2. Development of sample collection tools. Samples of code and traffic could be collected through a web interface or checked into a source code repository. The specific artifacts that will be collected are samples of Tor traffic, samples of non-Tor traffic, blocking-resistant encoding/decoding algorithms, and blocking algorithms that attempt to classify encoded traffic as either blocked or not. The most important element of design is openness. The current state of the collection should be visible to everyone in the Tor community and it should be easy for outside researchers in the area of blocking-resistant protocol design to add their own algorithms to the collection.

b. Point us to a code sample: something good and clean to demonstrate that you know what you're doing, ideally from an existing project.

My own blocking-resistant protocol design, Dust, is available here: https://github.com/blanu/Dust

c. Why do you want to work with The Tor Project / EFF in particular?

New DPI-based blocking algorithms are the greatest threat to free speech online today and Tor is the best and most used tool for maintaining free speech in the world. It is therefore likely that Tor will become a particular target for blocking, as indeed it already has in the case of the blockade of the Internet in Iran. Blocking-resistant algorithms are a natural fit for Tor, but without a clear winner among the transport options it is unclear how best to proceed. I deeply believe in the Tor project and the cause of Internet freedom. I think that Tor is our best hope for a free Internet and that finding the most effective blocking-resistant transport protocol is the best hope for the long-term viability of Tor as a censorship circumvention tool.

d. Tell us about your experiences in free software development environments. We especially want to hear examples of how you have collaborated with others rather than just working on a project by yourself.

I was the co-founder of the Freenet project and the project manager for the first three years of its existence. I maintained the mailing list and the SourceForge project, was a member of the board of directors of the parent non-profit organization, and I even printed the first round of t-shirts. This was the largest project I have ever worked on and it was also the first open source project I had collaborated on. As a result, I think of my time administering the Freenet project as a series of hard lessons in how not to run an open source project. However, all turned out well in the end as I've carried those lessons with me to this today and Freenet, though it bears little resemblence to the project I worked on, continues on and is a Summer of Code project this year as well. The primary challenge that I struggled with at the Freenet project was the balance between openness and staying on track. Free speech advocates are very local about their opinions and contributions, but the project still needs to move forward.

e. Will you be working full-time on the project for the summer, or will you have other commitments too (a second job, classes, etc)? If you won't be available full-time, please explain, and list timing if you know them for other major deadlines (e.g. exams). Having other activities isn't a deal-breaker, but we don't want to be surprised.

I am dedicating my full summer to Summer of Code. I have applied for a second Summer of Code project and if it's accepted I would like to do both projects as I think they will be complimentary. The other project is implementing a blocking-resistant transport for the StatusNet protocol. I have excellent time management skills and will have no difficulty completing both projects on time.

f. Will your project need more work and/or maintenance after the summer ends? What are the chances you will stick around and help out with that and other related projects?

While the code for the framework itself will be self-contained, the project of collecting algorithms and sample traffic will be ongoing as new use cases arise. As blocking-resistant protocol design is an areas of important research for me, I would love to continue to be involved in the search for the most effective blocking-resistant transport for Tor. My long-term goals for this project are 1) to start a competition or conference with Tor and the EFF to solicit the best blocking-resistant

protocol designs and 2) to see the best blocking-resistant transport incorporated into Tor and adopted for widespread use.

g. What is your ideal approach to keeping everybody informed of your progress, problems, and questions over the course of the project? Said another way, how much of a "manager" will you need your mentor to be?

I am a self-motivated worker that needs little management in terms of meeting deadlines. This is a necessity in both graduate school and telecommuting, two activities in which I have a great deal of experience. My strategy for maintaining progress is to check everything in to version control every day. That way it is always transparent as to the current state of the project. Additionally, I will be on IRC 24/7 to answer any spur of the moment questions and will send out brief but informative weekly high-level project updates.

h. What school are you attending? What year are you, and what's your major/degree/ focus? If you're part of a research group, which one?

The University of Texas at Austin, finishing my 2nd year in a PhD program on Information Studies, focusing on issues of information preservation such as censorship. I'm a member of the Cultivating Digital Librarianship Faculty (CDLF) program, which focuses on issues in digital libraries.

i. How can we contact you to ask you further questions? Google doesn't share your contact details with us automatically, so you should include that in your application. In addition, what's your IRC nickname? Interacting with us on IRC will help us get to know you, and help you get to know our community.

I'm currently on #tor as blanu. My email is brandon@blanu.net. Twitter is @blanu.

j. Is there anything else we should know that will make us like your project more?

My overall goal is to help Tor be an effective tool by figuring out the best way that it can continue to circumvent censorship as new techniques to censor are invented. This particular project proposal was my best guess as to what the next step should be. I'd be open to working on anything which furthers this same high-level goal. However, whenever I think about the future of Tor I always come back to the same issue, which is that we don't have any way of really knowing which encoding is best. It's expensive to switch to a new encoding. I think the plan the Tor team has right now is to just implement pluggable encodings and let people implement whatever encodings they want. This is a step in the right direction as the current encodings (straight SSL) has already been shown to have some vulnerabilities. However, I think that this approach isn't a long-term strategy if Tor starts to be blocked more as eventually a better default encoding will be needed. I think it's the right strategy right now, but that's because there's not enough information about which encodings are good. Demonstration of a clear winner would allow the Tor community to have the confidence to switch from SSL to something more effective when this becomes necessary.

Q&A

A) Can you get into some more specifics about actual technical things you're going to build? We don't need the whole set, but having some more details on some of them would help us to envision the project better. Or said another way, how should we plan to judge whether you've "finished" or "not finished" your summer project? A collection of tools and traces would be quite valuable, but I'd also like to have a roadmap for what tools and traces exactly we'll need, where we might get them, what tools will need to be built or adapted, how, etc.

A) The overall framework will be structured similar to unit tests. There will be a number of scenarios and you can select which scenarios you want to run. Each scenario will select a filtering algorithm and compare performance of the detector against different encoders and unencoded traffic. An example scenario is blocking of Tor traffic based on characteristics of the SSL handshake. Another scenario is blocking of all SSL traffic.

Example traffic: unencrypted web browsing, SSL encrypted web browsing, Tor encrypted web browsing, and a mix of other traffic such as Skype, DNS, IMAP, SSH, XMPP, and FTP. The best way to refine this list is probably to look at the proportion of Internet traffic of different protocols and try to provide a similar mix.
Example encoders: Tor with an extra SOCKS proxy, ssh tunneling, Dust, ObfuscatedSSH. We're short on encoders, but hopefully once this project is going it will make it easy for people to try out new encoder ideas and we'll have a growing list of options.
Example detectors: static string matching, packet length inspection, and timing are the major DPI techniques. So these general detectors will be implemented with parameters that can be tuned to detect some different patterns. I plan for these detectors to be pretty simple, none of the sophisticated machine learning techniques used by Hjelmvik to detect BitTorrent.
Tools: The simplest tools will be a way to upload and download items from the repository and a way to run the scenarios. These will probably be scripts on top of existing tools in the same category as git and make. Then there are the scenario scripts which will most likely be written using an existing unit test framework. They will run separate programs to do the encoding and detecting in order to allow for encoders and detectors to be written in any language as existing code will most likely be used for these. However, I will also make some reusable tools that make it easier to write detectors.
Detector tools: a tool to concatenate all of the content from captured packets in order to make string matching easier, a tool that convert captured packets to a sequence of packet lengths to make packet length inspection easier, and a tool that converts

captured packets into relative timings to make timing analysis easier. I'll probably also need to write a more general packet processing tool that puts captured packets into an easy to work with format, although if the available pcap libraries are excellent, this might not be necessary.
I'll probably write these tools in python if that's seems acceptable.
So in terms of project complection, I think the basic framework with uploading, downloading, running tests, and with some Tor and non-Tor traffic, two encoders, and three detectors (strings, lengths, and timings) is the minimum necessary to consider it a working project. Any extra time can be spent on adding more scenarios based on the priority of how relevant they seem to the blocking issues that Tor faces now or may plausibily face in the future.

B) Can you point us at the StatusNet people? We should talk to them more -- one of our main limiting factors this year is mentors, so we should coordinate with them to see what their situation is.

B) They are easily found on Freenode #statusnet. My main contact at StatusNet is Evan Prodromou, evan@prodromou.name. I'm sure he'd love to hear from you guys. He's quite the advocate of free speech online, open source, etc.. He was involved in Freenet in the early days.
Regarding my IRC connectivity, Freenode kicks me off every time I leave my computer for some reason, and my other IRC servers don't. I will get it sorted out before summer. :-)