The Tor Project: Pluggable Transports, Network Testing, and Outreach

1st QUARTER OF PROJECT, Q4 OF THE YEAR: JUL 1 2014 – SEP 30 2014

Table of Contents

Project Overview ........................................................................................................................................ 2
Foreign Assistance Framework: Internet Freedom Specific Indicators: ............ 2
Q4 Project Timeline (July 1, 2014 – September 30, 2014) – Activity Summary .5
  Pluggable transport integration .............................................................................................................. 5
  Testing and network simulation improvements ..................................................................................... 8
  Enhanced Outreach ................................................................................................................................. 9
Next Quarter’s Planned Activities: ......................................................................................................... 9
Project Overview

The project contains three main activities:

Part one: Pluggable transport integration
Tor will work to safely deploy transport technology to resist censorship, especially censorship based on deep packet inspection. This will be achieved by working with researchers to improve the usability, portability, security and code maintainability for transports; integrating mature transports into experimental bundles for real users and feeding those results into further improvements; and improving and maintaining the application programming interface between Tor and the pluggable transport layer.

Part two: Testing and network simulation improvements
Tor will improve the correctness and stability of the core Tor software by streamlining and automating the process of launching a complete test; designing and scripting an automated test suite to exercise and stress as much of Tor’s functionality as possible; and extending Tor’s controller interface to allow better monitoring.

Part three: Enhanced outreach
To increase awareness and improve the sustainability of the Tor technology and research, Tor will develop dynamic messaging and targeted campaigns aimed at end users and volunteers. Tor will identify partner organizations for outreach activities, including diaspora populations from countries with low press freedom rankings, open source development groups, and NGOs supporting civil society.

Foreign Assistance Framework: Internet Freedom Specific Indicators:

a. Number of unique users of circumvention technology or secure communication technology (measured monthly):

![Graph showing Directly connecting users from June 2014 to October 2014]
Tor use remained steady at 2.5 million daily users.

Bridges, alternative entry points into Tor, continue to work in the most Internet repressive countries such as Iran and China. Pluggable transport bridges that disguise Tor traffic are particularly useful. We do not have a way to measure every user of pluggable transport bridges. Care must be taken to analyze the impact on anonymity before more precise metrics are implemented, but we have seen an increase in use after we included more pluggable transports in the Tor Browser, including Meek. ¹

b. Number of civil society actors trained on circumvention or digital safety techniques:

Result: We created an inventory of Tor-produced educational content in the last quarter, and this quarter we expanded the inventory to include slide decks. We debated the audience list to help technical writers address people with varying levels of tech savvy. This project will allow us to create material that can be multi-purpose and span across audiences beyond the scope of this project. We anticipate using page views and partner reports to determine the reach of revised materials.

c. Number of USG-supported online tools developed or improved to maintain an open Internet:

Result: 3 including pluggable transports, network testing, and educational modules.

¹ https://trac.torproject.org/projects/tor/ticket/5040
d. Number of individuals or organizations operating in Internet repressive countries that are provided with technical assistance to increase online security:

Result: As part of another DRL-supported project; out of 4,254 help desk requests during this quarter, 783 were rejected as spam and 118 were still open. 97% of legitimate support requests were resolved.

e. Number of USG-assisted campaigns and programs to enhance public understanding, NGO support and media coverage on digital threats and promotion of an open Internet:

Result: We respond to requests to participate in programs for increasing basic Internet literacy, focusing on traffic analysis and encryption as it relates to solutions provided by Tor. We participated in the planning for a trainer summit at the Circumvention Tech Festival in Valencia, Spain, March 1-6 2015.

f. Number of times USG-supported analytic reports is cited by national and international media outlets:

Result: We will evaluate whether network testing meets this requirement.

g. Number of times USG-supported analytic reports is discussed on established social media sites:
Result: We will evaluate whether network testing meets this requirement.
Q4 Project Timeline (July 1, 2014 – September 30, 2014) – Activity Summary

Pluggable transport integration

Motivation and background
In July 2011, Tor introduced its _pluggable transports_ design. The idea is to separate Tor's anonymity and privacy properties from its censorship-resistance properties: the core Tor software focuses on building Tor circuits and getting the multi-layer encryption right, while the transport layer focuses on preventing an attacker from recognizing or blocking the client's connections to the rest of the Tor network. This modular approach lets us "plug in" new transports as needed—and since the transport layer is a separate program, it can be written in whatever rapid prototyping language is most convenient, allowing Tor to adapt much more quickly to a censor's new Deep Packet Inspection (DPI) tactics without needing to touch the core Tor program at all.

We developed a framework called Obfsproxy for handling the interactions between Tor and the transport layer, so transport developers could focus just on the censorship-resistance part of the problem. Our first example transport, obfs2, saw action in February 2012 in Iran: the censors blocked all SSL traffic countrywide by DPI, and within a few days we had 5000 users bypassing the firewall using our experimental bundle. Our second example transport, obfs3, is one of the few robust transports working in China today.

One of the main goals of the pluggable transport design was to make it easy for other researchers and developers to create their own transports for Tor. We succeeded: along with Obfsproxy, we now have Flash Proxy (a Tor/Stanford collaboration to let ordinary web browsers become Tor bridges using a protocol called Websockets), Skypemorph (a University of Waterloo project to make Tor's traffic flows look like Skype video), StegoTorus (a Tor/CMU/SRI collaboration to embed Tor traffic flows in real-world html, javascript, and pdf samples), Format-Transforming Encryption (a Portland State University project to develop regular-expression rules for transforming Tor traffic flows to match what the DPI rules expect to see), ScrambleSuit (a Karlstad University project that extends the obfs3 design to force an attacker to solve computational puzzles before he can confirm that a suspected address is really a Tor bridge), Telex (a Waterloo/Michigan project to redirect traffic flows at the Internet backbone, without the local censor being able to notice), FreeWave (an Illinois/UT Austin to tunnel traffic flows through the real Skype video client), and CloudTransport (a UT Austin project to fetch web pages via the Amazon S3 interface). Developers for circumvention tool projects like Dust and Lantern are exploring ways to reframe their work as a Tor pluggable transport, which if done right should let them inherit Tor's anonymity properties without any extra effort.

Project scope
We have plenty of research prototypes, and now it's time to clean them up and make them operational for users. First, we will work with transport researchers to improve usability,
portability, security, and code maintainability for their transports. (Both Stegotorus and Format-Transforming Encryption would have gotten around the April and May 2013 blocks in Iran, but we weren't comfortable giving either of them to real users because of code security problems.) Second, we will integrate sufficiently mature transports into experimental bundles for real users, and feed those results into further improvements. And third, we will improve and maintain the API between Tor and the pluggable transport layer---many key features remain unfinished, such as coordinating bandwidth rate limits between Tor and the transport layer, and disabling parts of the normal Tor bridge behavior to make it harder for attackers to scan for bridges.

**Milestones and deliverables**
The first question we need to resolve is how to prioritize which pluggable transport projects to work on. A complete answer for how to compare them is a complex research question---it's worth doing (and it's in-scope for our NSF project, so we have synergy with a second funder), but in this project we'll prioritize transports with an emphasis on how close they are to a practical deployment. The plan is to get more real-world experience with a variety of pluggable transport approaches, which will help inform both our future deployment decisions and also the research side.

For the six month milestone, we will decide on a list of evaluation points for assessing pluggable transport designs, and we will assess (i.e. form opinions about and summarize) each pluggable transport design based on these points. In addition, we will select the most promising three designs, and write a roadmap for what work remains before they are ready for integration into the Pluggable Transports Tor Browser Bundle.

For each of the 12, 18, and 24 month milestones, we will integrate one new transport implementation into the Pluggable Transport Tor Browser package. In parallel, we will update our assessment of the pluggable transport designs (based on progress made both by us and by the larger research community), including updating the list of evaluation criteria as needed, and we'll put out a new roadmap for what we then decide are the new three most promising designs.

For the final (24 month) milestone, we will additionally write an assessment of the new transports we've added: how much did they get used in practice? Where (and how) did they get blocked? With the benefit of hindsight, do we still think they were the best transports to have chosen for deployment? Have our evaluation criteria helped us or other projects in prioritizing new design efforts? Have we identified any conspicuous gaps in the design space, i.e. transport approaches that we now wish we could deploy if only they were closer to ready? What unexpected roadblocks did we encounter?

**Initial pluggable transport evaluation criteria**
While deciding on the evaluation questions is part of the milestones, here are some initial categories to give the reviewer a sense of our plans:

Section one, how reviewed / reviewable is it:

1. Is the software published, and is it entirely free / open source software? Some designs call for non-free (and non-distributable) components like Skype, a copy of Windows in a VM image, etc.

2. Is there a published design document, including a threat model? Is there a specification? How testable are its security / unblockability claims? We should
also consider how much peer review the design has received already, and whether the project is getting continued attention by its inventors.

3) What is its deployment history so far? What kind of users did it have (and how many), and how much publicity? Did it get blocked?

Section two, evaluation of design:

4) How difficult or expensive will it be to block the design (by protocol, by endpoints, etc)? For example, what services or protocols does it rely on being usable or reachable? Expense could include actual cost or could include collateral damage. Another way to measure might be the fraction of censoring countries where the technique is expected to work.

5) What anonymity improvements does the design provide, if any? While many pluggable transports focus only on reachability and leave anonymity properties to Tor, some research designs use the pluggable transport interface to experiment with improved traffic analysis resistance, such as by adding padding to defend better against website fingerprinting attacks.

6) What's the bandwidth overhead? Some transports like Obfsproxy don't inflate communication size, while others like Stegotorus wrap their communications in a more innocuous protocol at a cost of sending and receiving more bytes. Designs with higher bandwidth overhead can provide better blocking-resistance, but are also less suited for low-bandwidth environments.

7) Scanning-resistance: how does the design fare against active probing attacks, like China's follow-up connections that test for vanilla Tor traffic? ("How the Great Firewall of China is Blocking Tor", Philipp Winter, FOCI 2012).

Section three, evaluation of implementation:

8) Does the implementation use Tor's Pluggable Transport (PT) Application Programming Interface (API) already? Tor has a standard recommended approach so transport modules can be invoked and managed by the Tor process. The PT API also allows Tor to automatically publish capabilities of the transport, collect user and usage statistics from the transport, and so on.

9) Is the implementation cross-platform (Windows, OS X, Linux at least)? How about support for mobile platforms?

10) How easy is the build process, and how easy is deployment and scaling? For example, what software libraries does it require, how likely are we to get enough bridge-side addresses, etc?

11) How is the code from a security and maintainability perspective? Are there unit tests, integration tests, etc? While the underlying Tor channel provides security properties like encryption and authentication, pluggable transports can still introduce new security risks if designed or built improperly.

Other references:
Testing and network simulation improvements

The (1), (2), and (3) on items below refer to the three major proposal headings.

6 month milestone:

Designs and specifications for controller improvements to allow better monitoring of performance, scalability, and overhead, and to allow better testing of Tor features. (3)

Design and specifications for a unified pluggable integration test framework where developers can add their own network tests. (1,2)

Design and specifications for improved means to decouple Tor modules for better module-level testing. (2)

Identification of highest-priority areas for integration test coverage, and specifications for test modules to validate those areas. (2)

Identification of least-testable areas in current code/design, and plans for replacing/mitigating/testing them. (2)

Alpha version of testing framework, possibly missing major areas of functionality, is implemented, and in use as a regular part of tor development. (1,2)

12 month milestone:

Improve chutney templating to better handle complex multi-version configurations. (1)

Improvements to chutney to launch more kinds of Tor networks and test more Tor features. (1)

Initial implementations of some high-value controller features, so we can start getting experience with how they work. (3)

Alpha version of unified test framework running, including chutney for launching a tor network, stem for scripting Tor nodes, a set of test-clients, and a set of ill-behaved testing clients to check for correct handling of protocol violations. (2)

Alpha version of module-isolation mechanism. (2)

Tor test coverage (integration plus unit tests) at 60% or higher.
18 month milestone:

Test framework is stable enough to recommend to external developers, including a stable API and documentation on how to write extensions. All its components have received stable production-quality releases (2).

Improvements to chutney and test framework based on early reports from external developers. (1,2)

All specified controller features implemented. (3)

24 month milestone:

Tor test coverage (integration plus unit tests) at 80% or higher. (*)

Testing framework includes ill-behaved servers to verify correct behavior of network and clients in their presence.

All highest-priority areas identified by initial research have high (>>80%) test coverage. Most low-testability areas have been replaced, mitigated, or tested. (2)

Further controller improvements TBD based on experiences using initial round of new controller features. (3)

Module isolation and module-level tests integrated into Tor proper (2).

Further improvements TBD based on first 18 months' experience. (*)

**Enhanced Outreach**

- Change workflow of downloads to request donations on download-easy and download pages.
- Internet keyword campaign to target desired communities (native and diaspora).
- PR campaign to get the word out about Tor targeted at desired communities (native and diaspora).
- Reach out to foundations already working in desired communities to rebuild Tor's brand and expertise.

**Next Quarter’s Planned Activities:**

After discussions about moving the start date of this project to October 1, 2014, planning for the bulk of the activities outlined above will take place next quarter.