Award identification number: S-LMAQM-17-GR-1121
Recipient Organization: Tor Project Inc.
DUNS Number: 809211100
EIN: 1208096820A1
Period covered by report: FYQ3 2018

Tor Project

Progress Report Q2 2018 - S-LMAQM-17-GR-1121

Isabela Bagueros
Project Manager
The Tor Project
isabela@torproject.org

Table of Contents

Project Information

Grantee:	The Tor Project, Inc.
Project Title:	Tor Browser for Android
Award Number	S-LMAQM-17-GR-1121
Period of performance:	Q2 of the year -April 1st 2018 - June 31st 2018
Reporting date:	August 17, 2018
Reporting frequency:	Quarterly
Email contact:	isabela@torproject.org

Summary

Work related to Tor Browser for Android
We continued to get Tor Browser for Android ready for our first alpha launch. We are happy to say that some of the new user experience we built for Tor Browser for desktop alpha 8 series will be part of our first alpha for Android as well. This quarter we worked on a new user onboarding experience. The Android version of the onboarding only has steps for the features we have already built for this alpha, the rest will come along as we build the remaining functionality. We are sharing screenshots of it in this report. We landed our first patches upstream with Mozilla and our new team member (Android developer) start to work on our build system and release process so we are ready for the launch next quarter.

Work related to Tor network optimization for mobile users
As part of our effort to make core Tor more optimize for mobile use, we are reducing the binary size of core Tor build by doing the initial work to make modules conditionally compiled. This way a mobile developer can compile only the modules they need resulting on a smaller binary size to be added to their app. We also started to investigate how Tor is working for ARM64

mobile systems to see if we can have an impact on CPU load there as well. As for bandwidth consumption, we worked on the controller Tor has less wake ups that consumes bandwidth. And we wrote proposal 293, which will let us do some follow-up work so that we can later deploy proposal 275, which will reduce download size even further -- making consensus diffs roughly 50% smaller. We also fixed issues found at our controller API and continued to work on 'Report intermediate PT bootstrapping status'.

Objective 1: Build a Tor Browser for Android to empower mobile users to safely, anonymously, and securely interact with Internet resources and services.

Activity O1.1: Build a Tor Browser for Android with functionality and build processes in parity with desktop Tor Browser.

Progress Report:

We were working on patch rebasing for Tor Browser for Android, initially we thought to base Tor Browser for Android on Firefox 61 and later versions following the "normal" release cycle, not the ESR one, due to the concern about security updates we might miss. We rebased both our desktop Tor Browser patches + the Android ones for Firefox 61[1][2].

However, after we attended Mozilla All Hands and met with their Firefox mobile team, we learned that Mozilla Firefox for Android (Fennec) is currently in maintenance-mode. So we revised our plan and rebased the mobile patches for ESR60 instead[3].

There are a lot of other benefits on basing TBA on ESR60, we will be able to use the same git branch for desktop and mobile. The diff between releases will be small too, helping our developers with the release process (merging code etc). Our users will also benefit from this decision because the 60 series are the one that includes all the Quantum features that made Firefox a much faster browser.

Our team is finally completed, our Android developer started and is already working on our build system[4][5], we want to ensure that TBA has reproducible builds to increase its security. The plan for this project is documented on the ticket, here is the list of tasks our new developer created:

- Reproducible Tor Browser for Android builds
- List components required to build Tor Browser for Android
- Add definition for new platform in rbm.conf
- Add Android toolchain

[1] https://trac.torproject.org/projects/tor/ticket/25741
[2] https://trac.torproject.org/projects/tor/ticket/26233
[3] https://trac.torproject.org/projects/tor/ticket/26401
[4] https://trac.torproject.org/projects/tor/ticket/5709
[5] https://trac.torproject.org/projects/tor/ticket/26693

- Add Mobile Branding for Tor Browser
- Hardening Wrapper Removed in Debian Stretch
- Decide which ABIs to support for Android

We did a big investigation to ensure proxy safety on Android[6], while auditing the code for network connections we saw that Mozilla has already plugged many proxy-bypass calls. Most of the remaining instances are within the Firefox Accounts code. The telemetry code and mozstumbler have bypass bugs, too. We are continuing to audit the code for more proxy bypass bugs, we are also investigating the best way to control this connections, so we can tell which should go via proxy which can go directly.
We have updated our 'HACKING' document to include Tor Browser for Android build information[7]. This will help new developers to learn where is what and how our development team does their work for mobile.

During this quarter we also worked on 2 proposals:

- Future Plan for Tor Browser for Android[8]
    - At the Mozilla All-Hands in SF we had some discussions about Mozilla's future plans on Android. Based on those discussions we wrote this plant to align Tor Browser for Android's future with Mozilla's plan.
- Sandboxing Tor Browser - A New Beginning (?)[9]
    - This is the beginning of a conversation about creating a plan for moving towards sandboxing Tor Browser on every platform. This proposal will help us bring sandboxing to Android but also align our implementation with Mozilla's as well.

Risks Assessments:

We are happy to say that with our Non-Cost Extension and our team complete, we feel very confident that we will be on time with our timeline. Of course we are pay attention to our progress and the tasks ahead of us, but our team is definitely on a good pace.

Next Report [Q3 2018]:

We hope that during Q3 we have finished our alpha release, which all these efforts are for. After this first major release, which will be where The Tor project will launch officially our Tor Browser for Android with its own PlayStore and FDroid pages. So far all releases we have been doing are Orfox releases[10] because we were not yet in a good level of parity to start Tor Browser for Android alpha releases. We are now in a much better shape to do that and are looking forward for our first official release.

---

[6] https://trac.torproject.org/projects/tor/ticket/21863
[7] https://trac.torproject.org/projects/tor/ticket/25562
[8] https://lists.torproject.org/pipermail/tbb-dev/2018-June/000852.html
[9] https://lists.torproject.org/pipermail/tbb-dev/2018-July/000874.html
[10]
- based on Firefox 52.7.3 ESR (live 4/6)
- based on Firefox 52.8.1 ESR/52.8.0 ESR
- based on Firefox 52.9.0 ESR (live 7/10)

Activity 01.2: Research and develop Android specific fingerpriting defenses for Tor Browser.

Progress Report:

We continued our work to build fingerprinting defenses on Android[11]. We closed the tickets that we started to work on last quarter:

- We closed the ticket: Firefox Mobile(and Orfox) allows the user to copy the url to another app[12].
- We closed the ticket: Orfox lists external apps when the user clicks and holds an Android URI in a WebPage[13].
- We closed the ticket: Disable Firefox Mobile accessibility services[14].

And worked on these new tickets:

- We closed the ticket: intl.accept_languages changes when the user changes their OS language[15].
    - User language options was leaking even if the user didn't touched the settings. We fixed that so bad agents can't identify users by using this fingerprinting (language option).
- And we start to work on the ticket: audit or disable Apple HLS implementation on Android[16].
    - We start auditing Apple HLS on Android, looking for:
        - proxy bypasses;
        - disk avoidance;
        - fingerprinting;

Risks Assessments:

We are progressing in a good pace with this work and are feeling good about concluding it on time with our extended deadline.

Next Report [Q3 2018]:

For the next quarter we plan on continue investigating more fingerprinting possibilities on Android and fix them.

[11] https://trac.torproject.org/projects/tor/ticket/25703
[12] https://trac.torproject.org/projects/tor/ticket/25635
[13] https://trac.torproject.org/projects/tor/ticket/25790
[14] https://trac.torproject.org/projects/tor/ticket/25902
[15] https://trac.torproject.org/projects/tor/ticket/26018
[16] https://trac.torproject.org/projects/tor/ticket/26613

Activity 01.3: Work with Mozilla to merge and built defenses back into Firefox Mobile

Progress Report:

We finally started this work, we went to Mozilla All Hands in June and connected with their mobile team.

These are the first patches that we will be upstreaming to Firefox Mobile:

- HLS Player doesn't use the centralized Proxy Selector[17].
    - DefaultHttpDataSource, that it is used by the GeckoHlsPlayer, uses the default URL::openConnection (without Proxy) method provided by the Android SDK instead of the ProxySelector. This patch fixes that so all http connections in Fennec goes through a centralized proxy.
- Even when resistFingerprinting is enabled, FF leaks the OS locale in the accept headers[18].
    - This patch fixes locale leaking by the OS. When the user would go to about:config and changed the privacy.resistFingerprinting preference to true and then changed their language option on Android Input & Language settings.The HTTP Accept-Language header adds the OS Language. Now with our patch, the HTTP Accept-Language header does not change based on OS values.
- Firefox lists external apps even if the network.protocol-handler.external-default is false[19].
    - For this patch we fixed an issue where even when the user has set on about:config the network.protocol-handler.external-default to false, whenever they installed an external app the context menu would still allow the user to open that app. With our patch the app is not shown at the context menu.
- built_in_addons.json not created during mobile/android packaging[20]
    - This patch already existed for desktop but not yet for Android, our patch makes sure that during the packaging process, all system add-ons are listed in the built_in_addons.json file so th list can be parsed at the startup process without creating alerts that could leak information.

Risks Assessments:

Now that we have established a relationship with Mozilla's Firefox Mobile team we

Next Report [Q3 2018]:

We will continue to submit our patches upstream to Mozilla as we work on fixing Android specific fingerprinting and other privacy protection features.

[17] https://bugzilla.mozilla.org/show_bug.cgi?id=1459420
[18] https://bugzilla.mozilla.org/show_bug.cgi?id=1459089
[19] https://bugzilla.mozilla.org/show_bug.cgi?id=1455165
[20] https://bugzilla.mozilla.org/show_bug.cgi?id=1440789

Objective 2: Specifically Build a Tor Browser for Android to empower mobile users to safely, anonymously, and securely interact with Internet resources and services.

Activity O2.1: Enable standalone use of Tor Browser for Android without separate Orbot installation

Progress Report:

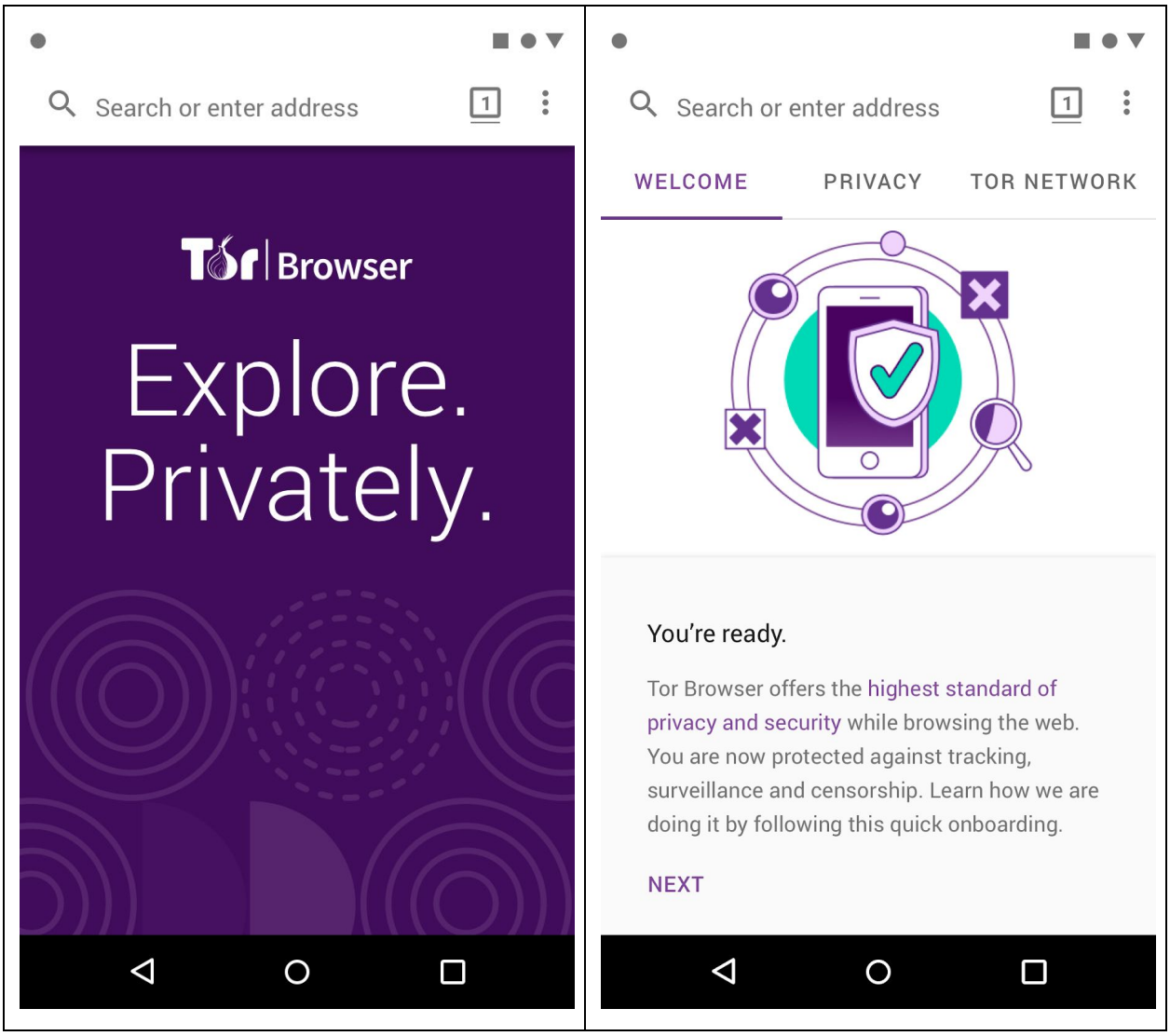
Because we wanted to be ready to launch our alpha we decided to prioritize the build process over this task during Q2. We will continue to work here on Q3 and are still aiming to have our stable release completely independent of Orbot.

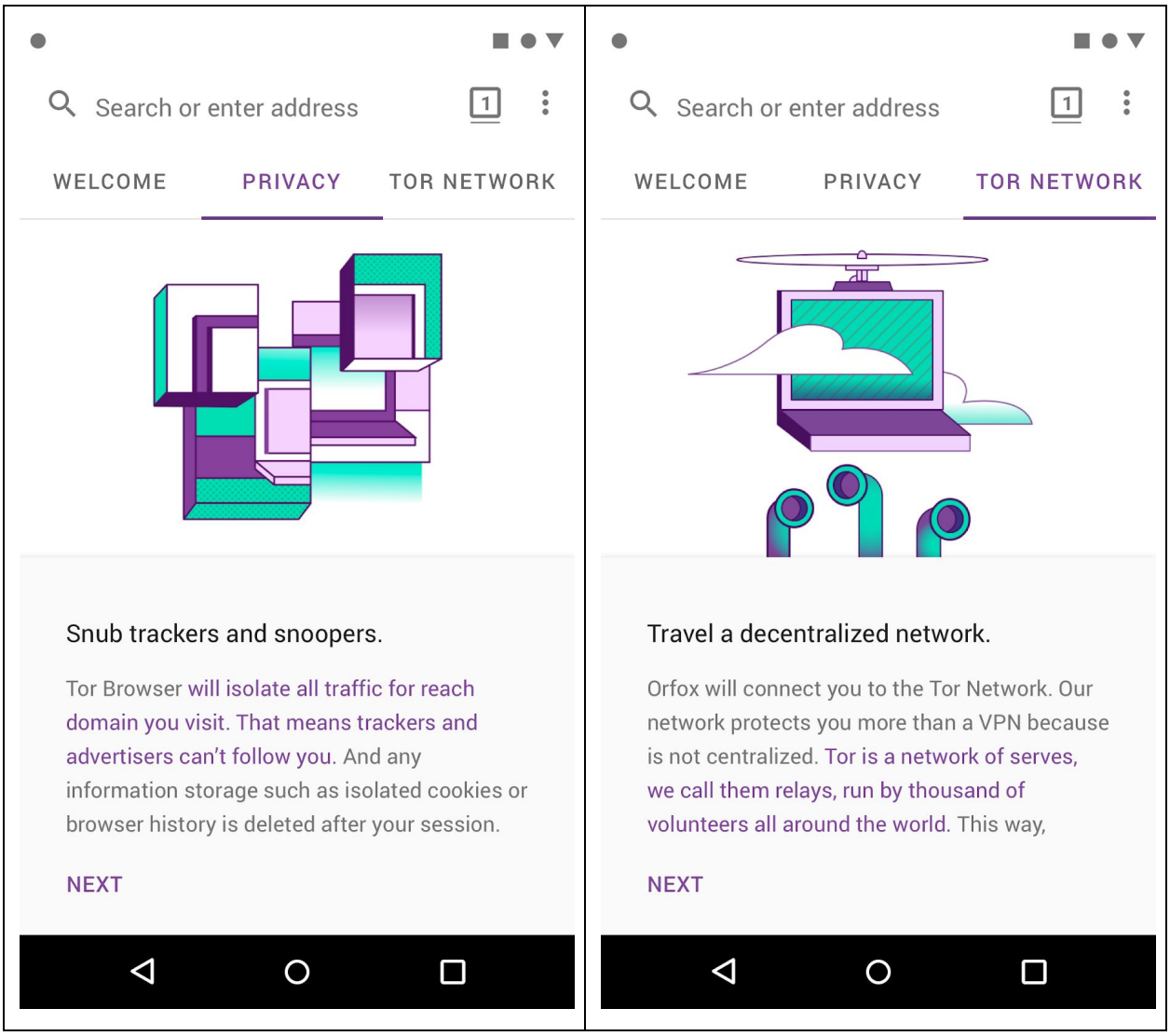
Risks Assessments:
N/A

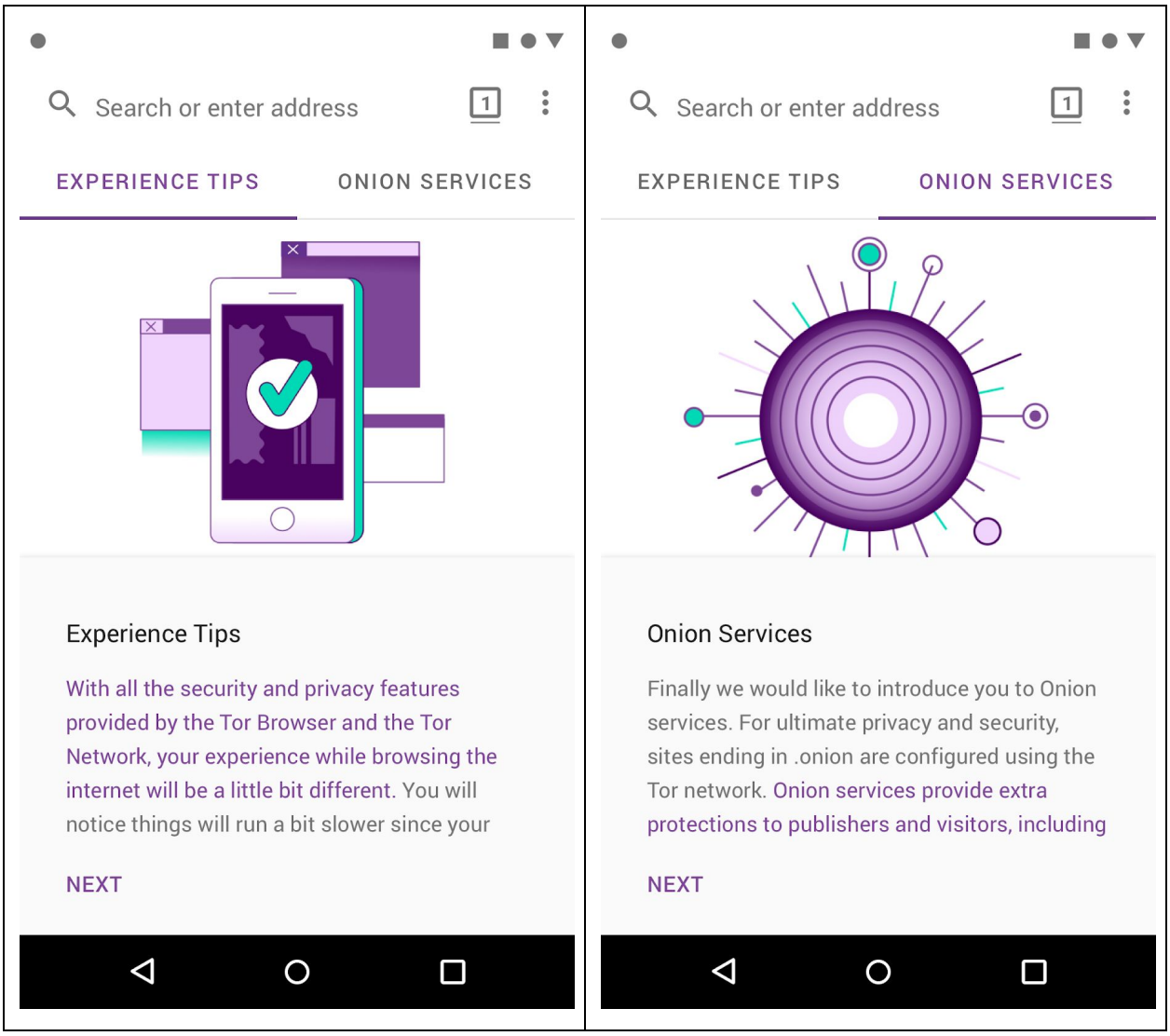
Next Report [Q3 2018]:
For Q3 we will continue to work on this task, aiming to have our stable release completely independent of Orbot.

Activity 02.2: Improve usability of Tor Browser for Android, relative to Orfox, including anti-censorship bridges.

Progress Report:
During Q2 we continued to bring all UX improvements we are building to Tor Browser for desktop to our Android app. We are making a major change to our new user experience, we redesigned our about:tor page (landing page at our Browser) and added a new user onboarding guide explaining Tor Browser features and benefits to the user.


Search or enter address
Tor Browser
Explore.
Privately.
Search or enter address
WELCOME
PRIVACY
TOR NETWORK
You're ready.
Tor Browser offers the highest standard of privacy and security while browsing the web. You are now protected against tracking, surveillance and censorship. Learn how we are doing it by following this quick onboarding.
NEXT


Search or enter address
1
WELCOME
PRIVACY
TOR NETWORK
Snub trackers and snoopers.
Tor Browser will isolate all traffic for reach domain you visit. That means trackers and advertisers can't follow you. And any information storage such as isolated cookies or browser history is deleted after your session.
NEXT
Search or enter address
1
WELCOME
PRIVACY
TOR NETWORK
Travel a decentralized network.
Orfox will connect you to the Tor Network. Our network protects you more than a VPN because is not centralized. Tor is a network of serves, we call them relays, run by thousand of volunteers all around the world. This way,
NEXT



Our team is still iterating with the copy and some parts of the experience, but we are very excited that this will part of our alpha launch in Q3.

Risks Assessments:

N/A

Next Report [Q3 2018]:

Our goal is to lend this experience together with our circuit display and other small UX improvements with our first alpha. We will also start working on other pieces that are part of the mobile experience such as the ‘tor launcher’ piece.

Activity 02.3: Improve software and Tor Network architecture to improve usability for low-speed networks and low-power, low-RAM devices.

Progress Report:

Binary size

We did the initial work to make modules conditionally compiled[21]. The idea is to try to extract modules out of Tor into compile time options in order to help the modularization effort and shrinking the binary size down for mobile. We completed all the following tasks:

- Identify a list of tor module[22].
- Basic modularization preparation[23].
- Modularized directory authority subsystem[24].
- Better safeguard authdir_mode_v3() if dirauth module is disabled[25].
- Write documentation in doc/ on how to write a module[26].

CPU usage

We also started to investigate how Tor is working for ARM64 mobile systems[27]. Orbot is an app that could benefit from us reviewing how Tor is compiling with it, if there are problems or anything we could improve.

Bandwidth consumption:

We worked on the controller[28] so when it tells Tor to disable the network, Tor wakes up very infrequently. Even when Tor is running, it makes Tor way up much frequently: see for example #25373[29].

We wrote proposal 293[30], "Other ways for relays to know when to publish", which will let us do some follow-up work so that we can later deploy proposal 275[31], "Stop including meaningful "published" time in microdescriptor consensus," which will reduce download size even further -- making consensus diffs roughly 50% smaller. (We won't be able to deploy 275 until all relays implement 293, so it won't be deployed during the scope of this contract.)

[21] https://trac.torproject.org/projects/tor/ticket/25494
[22] https://trac.torproject.org/projects/tor/ticket/25495
[23] https://trac.torproject.org/projects/tor/ticket/25498
[24] https://trac.torproject.org/projects/tor/ticket/25610
[25] https://trac.torproject.org/projects/tor/ticket/25990
[26] https://trac.torproject.org/projects/tor/ticket/25991
[27] https://trac.torproject.org/projects/tor/ticket/25496
[28] https://trac.torproject.org/projects/tor/ticket/25500
[29] https://trac.torproject.org/projects/tor/ticket/25373
[30] https://gitweb.torproject.org/torspec.git/tree/proposals/293-know-when-to-publish.txt
[31] https://gitweb.torproject.org/torspec.git/tree/proposals/275-md-published-time-is-silly.txt

Risks Assessments:
N/A

Next Report [Q3 2018]:
We are hoping to make more improvements at RAM consumption for Q3.

Activity 02.4: Improve the Tor Network's controller interface to allow mobile apps to reduce bandwidth and battery use.

Progress Report:
We fixed a couple of issues related to features at our controller:

- We work on how Tor requests authority certificates on first bootstrap[32]. We don't want to launch a certificate fetch always during the scheduled periodic consensus fetch but only in those cases when consensus are waiting for certificates.
- Fixed: Reset bootstrapping state on shutdown[33]. The current behavior of our embedding API could break restart on mobile.

Risks Assessments:
Still trying to get more feedback from mobile developers at our API.

Next Report [Q3 2018]:
During Q2 we reached out to all projects we contected to give us feedback at our API as we have received very little feedback so far and is hoping more developers get back to us on that. Our plan is to address these feedbacks in Q3.

Activity 02.5: Enable better reporting of network and connection errors to apps that use Tor Network

Progress Report:
We continued to work on 'Report intermediate PT bootstrapping status'[34] so we can give better error messages when a pluggable transport is not working to connect to the Tor network.

Risks Assessments:
N/A

[32] https://trac.torproject.org/projects/tor/ticket/24740
[33] https://trac.torproject.org/projects/tor/ticket/24587
[34] https://trac.torproject.org/projects/tor/ticket/25502

Next Report [Q3 2018]:

For Q3 we hope to finish the PT bootstrapping status work and also finish our work related to progress bar percentage completion report[35]. Our goal here is to have percentage reports that reflects better which stage of progress the network bootstrapping is, so the client better know where it's stuck and be able to debug it.

[35] https://trac.torproject.org/projects/tor/ticket/22266