Arti: Rewriting Tor in Rust

Alexander Hansen Færøy

October 28, 2026

Copenhagen Rust Community



About Me

- Core Developer at The Tor Project since early 2017. Team Lead of the Network Team since late 2019.
- Free Software developer since 2006.
- Worked with distributed systems in the Erlang programming language, WebKit-based mobile browsers, embedded development, lots of security, cryptography, and general software development consulting.
- Co-organizing the annual Danish hacker festival BornHack on Funen.

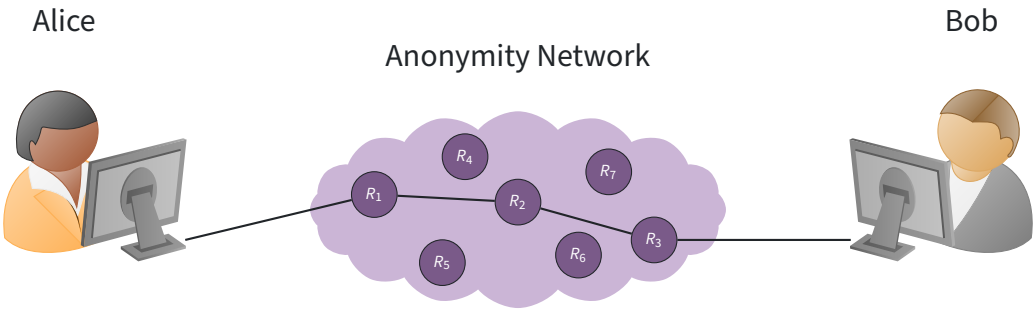


What is Tor?

- Online anonymity, censorship circumvention, and fingerprinting resistance.
  - Free software.
  - Open network.
  - Open specifications.
- Community of researchers, developers, users, and relay operators.
- U.S. 501(c)(3) non-profit organization.
- Almost 70 people working on Tor as their dayjob. Everything presented here is a team effort.

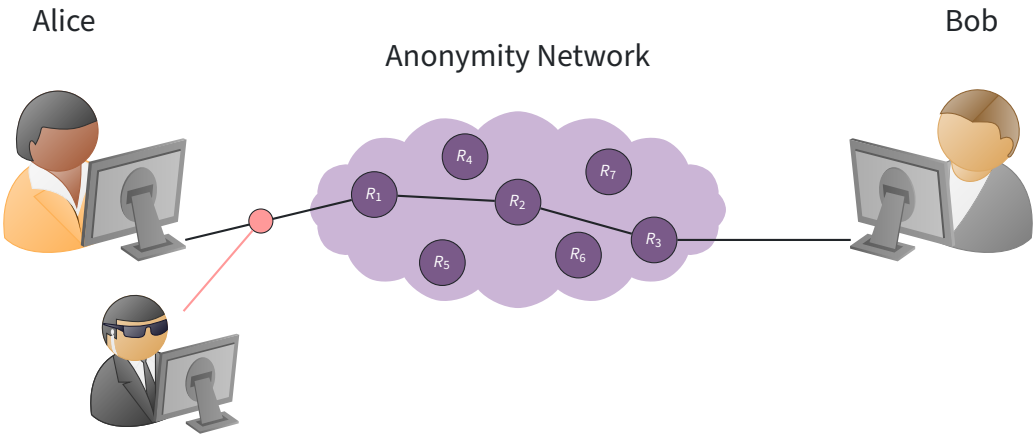


What is Tor?

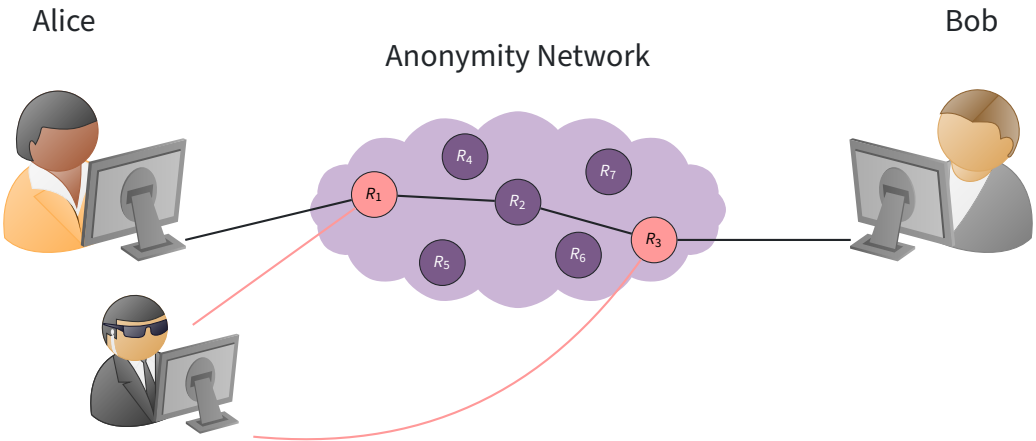


What can the attacker do?

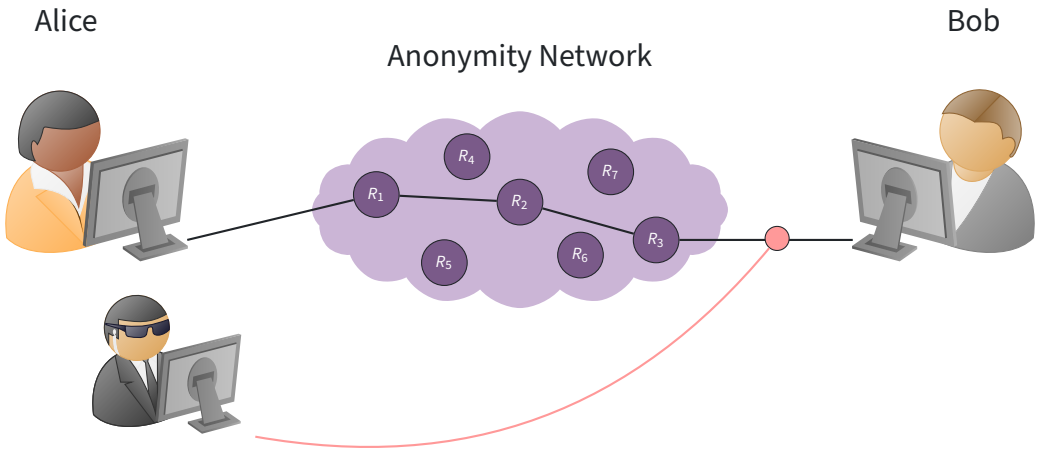
What is Tor?



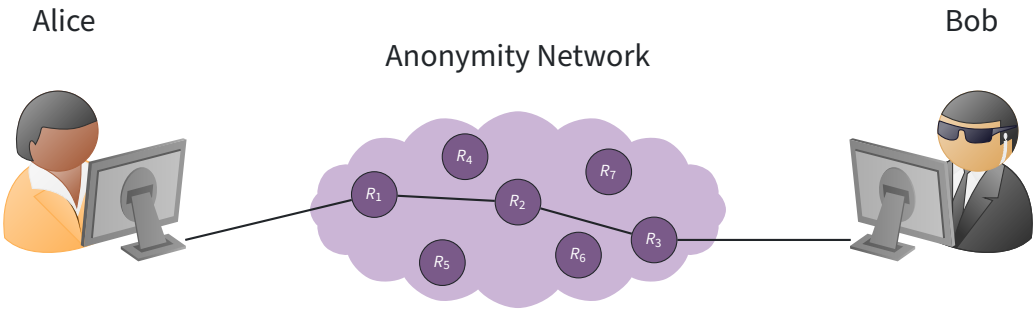
What is Tor?



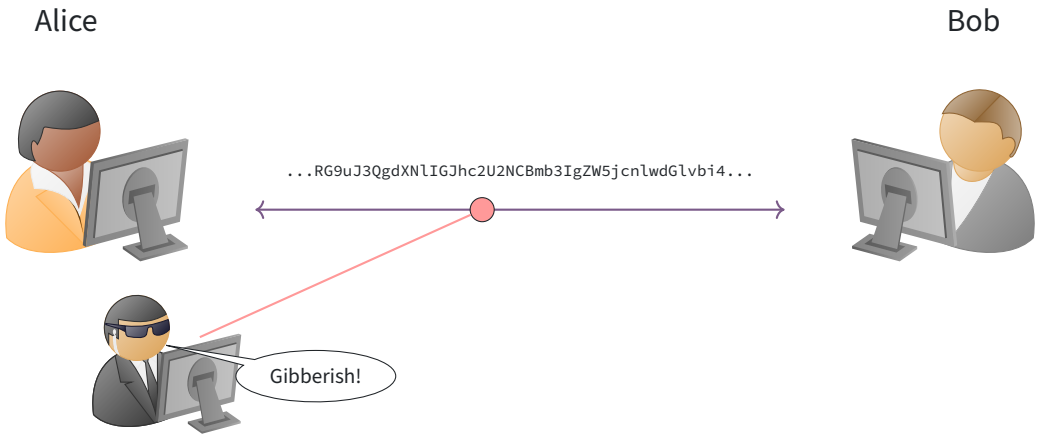
What is Tor?



What is Tor?



Anonymity isn’t Encryption

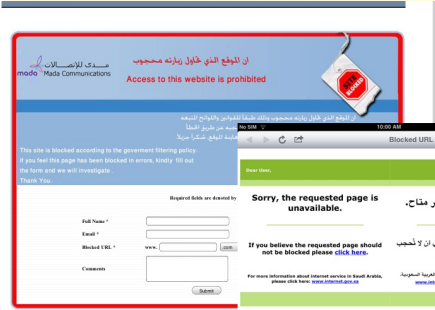


Encryption just protects contents.

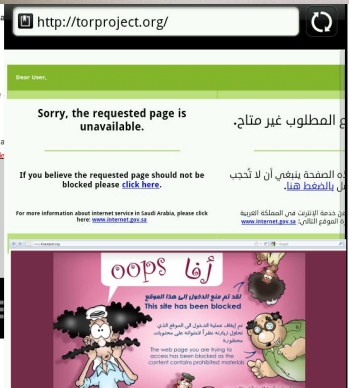

يالله بالستر ...!
تصفح بأمان!
عذراً، هذا الموقع غير متاح في دولة الإمارات العربية المتحدة.
تشكل شبكة الانترنت وسيلة لتواصل والمعرفة وخدمة متطلبات حياتنا اليومية. وقد تم حجب الموقع الذي تريد بدخوله لاشتماله محتوى مدرج تحت "فئات المحتويات المحظورة" حسب تصنيف "السياسة التنظيمية لإدارة النفاذ للإنترنت" لهيئة تنظيم الاتصالات بدولة الإمارات العربية المتحدة.
إذا كانت لديك وجهة نظر مختلفة، الرجاء النقر هنا.
Surf Safely!
This website is not accessible in the UAE.
The Internet is a powerful medium for communication, sharing and serving our daily learning needs. However, the site you are trying to access contains content that is prohibited under the 'Internet Access Management Regulatory Policy' of the Telecommunications Regulatory Authority of the United Arab Emirates.
If you believe the website you are trying to access does not contain any such content, please click here.
© 2009 zammitcre FZ LLC.


خطر!
تصفح بأمان!
تشكل شبكة الانترنت وسيلة للتواصل والمعرفة وخدمة متطلبات حياتنا اليومية. وقد تم حجب الموقع الذي تريد بدخوله لاشتماله محتوى مدرج تحت "فئات المحتويات المحظورة" حسب تصنيف "السياسة التنظيمية لإدارة النفاذ للإنترنت" لهيئة تنظيم الاتصالات بدولة الإمارات العربية المتحدة.
إذا كانت لديك وجهة نظر مختلفة، الرجاء النقر هنا.
Surf Safely!
This website is not accessible in the UAE.
BOMB
Site is now TRUSTED


هذا الموقع محظور
This site is blocked
الوصول إلى هذا الموقع غير مسموح به حالياً لأنه مصنف ضمن فئات المحتويات المحظورة بموجب أحكام السياسة التنظيمية لإدارة النفاذ إلى الإنترنت في دولة الإمارات العربية المتحدة
Access to this site is currently blocked. The site falls under the Prohibited Content Categories of the UAE's Internet Access Management Policy.
إذا كنت ترغب في إضافة النظر في تصنيف هذا الموقع، يرجى التفضل بتعبئة استمارة الملاحظات
If you would like the classification on this site to be reviewed, please fill in and submit the Feedback Fo


Access Denied
our request was denied because of its content categorization: "Computers/Internet;Proxy Avoida


عزيزي العميل : تم حجب هذا الموقع بناء على القوانين والقوانين
بأن هناك خطأ يرجى إرسال رسالة على البريد الالكتروني unblock.kw@kw.zain.com مع ذكر عنوان الموقع الذي تم حجبه.
Dear Customer: This site has
categorizing this site, please


http://torproject.org/
Dear User,
Sorry, the requested page is unavailable.
If you believe the requested page should not be blocked please click here.
For more information about internet service in Saudi Arabia, please click here: www.internet.gov.sa
عزيزي المستخدم،
عفواً، الموقع المطلوب غير متاح.
إذا كنت ترى أن هذه الصفحة ينبغي أن لا تُحجب تفضل بالضغط هنا.
مزيد من المعلومات عن خدمة الإنترنت في المملكة العربية السعودية: www.internet.gov.sa

مدى للاتصالات
mada Mada Communications
ان الموقع الذي حاول زيارته محجوب
Access to this website is prohibited
This site is blocked according to the government filtering policy.
If you feel this page has been blocked in errors, kindly fill out the form and we will investigate.
Thank You
Required fields are denoted by
Full Name *
Email *
Blocked URL *
Comments
Submit

9:28 AM
Site Blocke...
http://torproject.org/
Dear User,
Site Blocked
This web site has been blocked for violating regulations and laws of Kingdom of Bahrain.
If you believe the requested page should Not be blocked please click here.
الموقع محظور
هذا الموقع مغلق لمخالفته الأنظمة والقوانين في مملكة البحرين.
إذ كنت ترى أن هذه الصفحة ينبغي أن لا تحجب تفضل بالضغط هنا.

oops! أوقا
This site has been blocked
The web page you are trying to access has been blocked as the content contains prohibited materials

Du var på vej ind på en ulovlig hjemmeside

Vi vil meget gerne hjælpe dig med at finde den film eller serie, du søger.

Søg med FilmFinder →

Hvis du er på udkig efter musik, bøger eller møbler

Gå til SHARE WITH CARE →

SHARE WITH CARE

Hjemmesiden er blevet blokeret, fordi den er dømt ulovlig ved en dansk domstol. Brug Share With Care til at finde det, du leder efter, lovligt. På den måde passer du både på dig selv og på kulturen. **Læs mere om Share With Care**

Kulturministeriet · RettighedsAlliancen · TI Tele Industrien · DI Digital

Copenhagen Rust Community | Meetup

https://www.meetup.com/copenhagen-rust-community/

ورود به سیستم Google یا Meetup

ادامه دادن

Olbernhau, DE

English Log in Sign up



Copenhagen Rust Community

4.8 • 309 ratings

Copenhagen, Denmark

1,093 members

Share

About Events Members Photos Discussions

Join this group

About us

A place for aspiring and experienced Rust enthusiasts to come together to discover and discuss the latest Rust developments, showcase Rust projects and crates, network and meet each other in a supportive community. 🦀🧡

We meet on a monthly basis somewhere in Copenhagen and are open to everyone interested in learning more about Rust, even if you have no prior experience with the language.

Organizers

Troels Hoffmeyer and 7 others

Message

Members 1,093

See all

What is Tor?

Online anonymity is provided by C Tor and/or Arti.

Censorship circumvention is provided via our Pluggable Transport ecosystem: Snowflake, obfs4, meek, ...

Fingerprinting resistance is provided via Tor Browser, Tor Browser for Android, Tails, and other apps with domain specific needs for isolation and fingerprinting resistance.

2017: Meeting in Amsterdam

In 2017, The Tor Project community meets in Amsterdam for our annual developer meeting.

During this meeting, a session is held about building Tor implementations in alternative languages.

As you can likely imagine, everyone had different views about the direction here: some wanted Go, some wanted C++, some wanted Python, I wanted Erlang, a few folks expressed interest in Rust.

Sebastian Hahn pulled an "all nighter" and got a very basic "Hello world" in Rust running as part of C Tor's build system (autotools).

C Tor and Rust

Sebastian's work allowed us to do module-by-module conversion of the C Tor codebase to Rust. This would allow us to slowly migrate more sensitive, and network exposed code to Rust earlier and get some value right away.

Sebastian Hahn, Chelsea Komlo, isis lovecruft, Taylor Yu, teor, Nick Mathewson, and others wrote most of the early code here.

Sadly, C Tor's architecture became a problem.



C Tor and Rust

We discovered interesting bugs in our `protover` code, but we also discovered that calling in and out of Rust with our current architecture was going to be a pain.

We cleaned up the directory layout and split things into modules in the C Tor code tree, making it easier to maintain.

Slowly, we started giving this project less attention as other priorities showed up :-(

C Tor Pain Points

To get an overview of some of our pain points here with C Tor that we are interested in fixing:

- Unix focused architecture of the application, designed in the late 90ies.
- One binary, many purposes.
- "Mobile" support hacked in: "use less memory, please", "make it a library, please", "how do I restart Tor from my app?", etc.
- Our event loop: libevent: mighty fine on Unix. Falls back to `select()` on Windows (only WinSock supported).
- Single-threaded architecture leads to poor performance, particularly in the Onion Services and Relay setting.

2020

In the beginning of 2020, the arrival of COVID-19 impacted most parts of the world: massive economic uncertainty.

The Tor Project's funding situation was negatively impacted as well: 13 out of 35 laid off.

How can we make the organization sustainable in the long run to avoid this situation again in the future?

Is having a browser a good idea for such a small organization?

Arti is born

In May 2020, Nick Mathewson’s work on "Arti" begins happening more openly and gets attention from other people within Tor’s Network Team.

We work internally on a strategy where Nick can focus his full time on Arti while the rest of us tries to cover the deliverables from our grants. We keep this somewhat secret as we at this point have no idea how it will go.

We slowly begin looking for funding to turn Arti into a "real" Tor product.

Arti Client Example

```rust
#[tokio::main]
async fn main() -> Result<()> {
    tracing_subscriber::fmt::init();

    let config = TorClientConfig::default();
    let tor_client = TorClient::create_bootstrapped(config).await?;

    let mut stream = tor_client.connect(("example.com", 80)).await?;
    stream
        .write_all(b"GET / HTTP/1.1\r\n\
                     Host: example.com\r\n\
                     Connection: close\r\n\r\n")
        .await?;
    stream.flush().await?;

    let mut buf = Vec::new();
    stream.read_to_end(&mut buf).await?;

    println!("{}", String::from_utf8_lossy(&buf));

    Ok(())
}
```

Goals for Arti

Write a Rust implementation of Tor that is usable as a client (at first) and allows people to generally "work with our ecosystem" using its modules.

This would help us reduce other codebases written in Python, Java, and Go that are used for analysis, and metrics processing: each one of these contained their own encoders/decoders for different types of objects that exists in our ecosystem, and they all had compatibility issues.

Tor and Memory Safety

21 out of 34 of Tor's TROVE's was related to memory issues allowed by the C programming language.

This number predates the current world reality where LLM/AI-assisted security scanning exists.

We will return to this later...

Funding Arti Development

Over the years, we have managed to find sponsors for this project:

For Client and Onion Services work: Zcash Community Grants, OpenSats, Ethereum Foundation.

Currently working on the Arti Relay project that is sponsored by U.S. Department of State Bureau of Democracy, Human Rights, and Labor.

And of course, donations :-)

torproject.org/about/supporters

Arti: The Original Roadmap

0.1.0	API stability. Year 1
1.0.0	Usability, performance, and stability. Year 1
1.1.0	Anti-censorship. Year 1
1.2.0	Onion services. Year 2
2.0.0	Ready to replace the C client. Year 2
Now	Relay, bridge, directory authority, etc.

We are now in year 5

Arti: Design Choices

Arti currently contains some pluggable components:

You can relatively easily make it use your async run-time of choice. Originally, we focused on `async-std`, but slowly `tokio` became our main priority. We also support Smol and are working with Ethereum Foundation on an experimental, single-threaded, WASM run-time for Arti.

Arti Client supports multiple TLS implementations.

Arti Relays is only going to support Tokio (and rustls).

2020: TorVPN is born

Internally, those of us left begin thinking about how we can build tooling where potentially others can solve the problem of fingerprinting, such that we can focus on Anonymity and Censorship circumvention.

We request funding for building a TorVPN application, for Android, with the C Tor implementation as its anonymity engine.

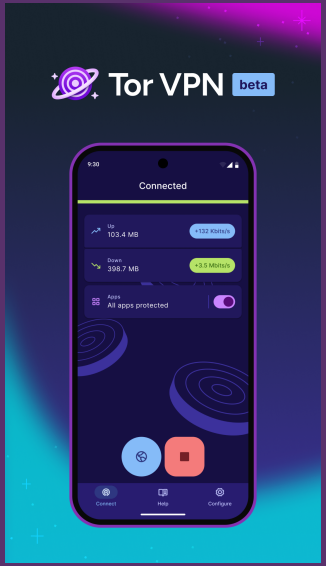
At this point, very few people inside of Tor are even aware of Arti existing.

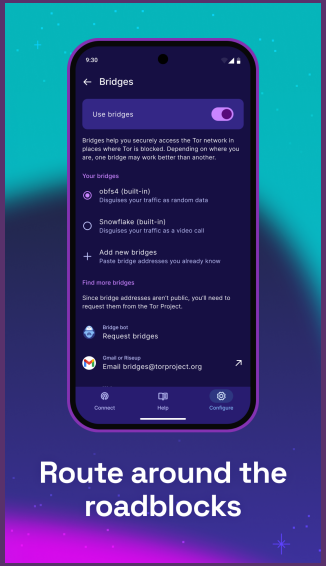
TorVPN Architecture

Arti is making excellent progress in 2020, 2021, and 2022. We rethink our original idea of using C Tor for the TorVPN application.

David Goulet and I spend two weeks building a prototype using Arti and Smoltcp. We call this project Onionmasq.

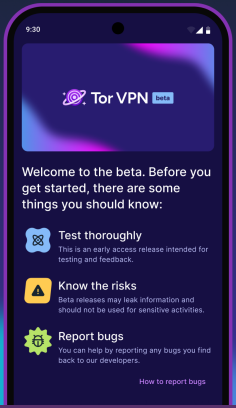
Smoltcp came with some desirable properties thanks to being a library focused on embedded hardware (ESP32): no dynamic allocations.





Route around the roadblocks

Join the beta and start testing today



TorVPN

- Create a VPN-like experience using the Tor network.
- Focus on Google's Android firstly. Apple's iOS platform is difficult due to Network Extension memory limits.
- Use per-app circuit isolation using the platform's network stack and by faking DNS responses.
- Built to support other experiences than just the browser.
- Tor beyond HTTP/Socks proxies: UDP support.
- Ended up being our first end-user product built with Arti.
- Available via F-Droid and Google Play Store.

TorVPN: Development Process

- Greatly enhanced our tooling for running isolated Tor networks. Both locally and distributed using Shadow and Chutney.
- The "VPN-engine" (Onionmasq) is also working on other platforms with access to TUN/TAP devices.
- Created Oniux as an alternative to torsocks/torify on Linux (uses Onionmasq and Linux namespaces for isolation).
- Stronger padding state machines when the network is transitioned to Arti.
- VPN leak investigation with hackers from GrapheneOS.

LLM/AI Security Bugs

In March 2025, the quality level of security bugs we received from obvious LLM/AI users went up. By a lot.

We started seeing more and more security bugs targetting C Tor. This was great in that we found legit security bugs, but it was bad for team capacity: security bugs requires a significant amount of attention.

We end up closing our HackerOne bug bounty system, and now work more directly with the large model providers. Particularly, Calif.io and Anthropic are giving us very high quality bug reports on C Tor, but also Arti.

LLM/AI Security Bugs

We recently received over 300 security bugs from Anthropic on C Tor. We are still working our way through that.

They also did an analysis of Arti: here they found 40 bugs, but nothing as concerning as the ones we got for C Tor.

Ethereum Foundation also did an agentic audit: main takeaway from there was poor choices of algorithms in some highly specific places in Arti.

Arti gets a release once per month, we now do C Tor releases every two weeks.

Arti: Current Roadmap

The team is currently working on the Arti Relay support: Relay, Bridges, and Directory Authorities. We hope to have a Test Network running with mixed Arti and C Tor nodes in Q1 2027.

Handle the LLM/AI security bug intake while slowly making progress with Arti.

Get TorVPN out of beta, please try it out and report bugs!

Find ways to "close doors" for our C Tor needs:

- Get Onion Services operators to move to Arti.
- Get Tails and Tor Browser to move to Arti.
- Move consensus algorithm in C Tor to an implementation written in Rust using a plug-in.

Working in the Rust ecosystem

- `derive-deftly` for data structure driven macros
- `fs-mistrust` for filesystem access checks.
- `safelog` for handling sensitive data in logs.
- `std::SystemTime::{MIN, MAX}` implementation.
- Linux kernel performance fixes as part of our Shadow simulation software.
- Participate in programs such as Outreachy and Google Summer of Code.
- Generally, our team loves the Rust community, and want to contribute to its success!

Working in the Rust ecosystem: `derive-deftly`

```
use derive_deftly::{define_derive_deftly, Deftly};

define_derive_deftly! {
    ListVariants:

    impl $ttype {
        fn list_variants() -> Vec<&'static str> {
            vec![ $( stringify!( $vname ) , ) ]
        }
    }
}

#[derive(Deftly)]
#[derive_deftly(ListVariants)]
enum Enum {
    UnitVariant,
    StructVariant { a: u8, b: u16 },
    TupleVariant(u8, u16),
}

assert_eq!(
    Enum::list_variants(),
    ["UnitVariant", "StructVariant", "TupleVariant"],
);
```

Arti: Post Relays

Arti is an investment into Tor's future: Arti itself is interesting not so much in terms of what it is, but what we hope it can provide:

- Make it easier to build new features with confidence in the security.
- Experiment with radical changes to the Tor protocol: UDP support, Walking Onions, Post-quantum Cryptography, etc.
- A more performant Tor network: likely also a smaller Tor network.
- Being able to attract talented engineers interested in heterogeneous distributed systems with a significant amount of interesting cryptography.
- Focus on supply chain security (matches Rust Foundation Strategic Plan: 2026-2028).

Tracking Arti Engineering Efforts

All Arti development is free software and development happens in public:

- Day-to-day development happens over IRC at #tor-dev @ OFTC and via Matrix #tor-dev:matrix.org.
- Gitlab is used for engineering work at gitlab.torproject.org/tpo/core/arti.
- Protocol specifications available at spec.torproject.org.
- We are encouraging contributions from upcoming Tor hackers! Rust knowledge useful – you will learn about Tor as you progress.

How can you help?

- Build cool tools with Arti. We have example code for Axum, ureq, etc.
- Find, and maybe fix, bugs in Arti.
- Test Arti on your platform of choice.
- Work on some of the many open research projects.
- Run a relay or a bridge!
- Teach others about Tor and privacy in general!



Network Team Job Openings

Interested in hacking on Arti? Our team will likely soon have two fully remote job openings at Tor:

- Rust Software Engineer (Onion Services)
  Help making our Onion Services ecosystem stronger.
- Rust Software Engineer (Client Integration)
  Help internal and external teams with integrating Arti.

Pay attention on social media and via torproject.org/about/jobs.

Questions?

ahf@torproject.org

OpenPGP:
1C1B C007 A9F6 07AA 8152
C040 BEA7 B180 B149 1921

@ahf@mastodon.social

@ahf.me



This work is licensed under a

Creative Commons
Attribution-ShareAlike 4.0 International License

