Rapid Field Testing and Deployment of Circumvention Techniques within Russia

LEAP Encryption Access Project

April, 2025

Table of Contents

Summary....3
Circumvention Field Testing....5
  Methodology....5
  Analysis....5
  Results....8
Avos Provisioning....10
Tunnel Measurement Tooling....12
Lessons Learned....13

Summary

LEAP Encryption Access Project (LEAP) is field-testing and helping provision multiple circumvention techniques (CTs) tailored specifically to the Russian context. Our comprehensive approach, covering development, testing, provisioning, and secure invite distribution, addresses the complex and varied network conditions across Russia. We prioritize user safety through a privacy-preserving methodology and rely on first-hand user feedback to guide technical improvements.

Building on our previous report on obfs4 + KCP field testing, this document details current censorship conditions in Russia and presents findings from our field tests of three additional CTs: LEAP's Hopping Pluggable Transport + obfs4, QUIC, and the Hopping Pluggable Transport + QUIC. Following successful testing, all four CTs are being provisioned through Avos, a newly launched private LEAP VPN provider. Avos works directly with Russian civil society organizations (CSOs), distributing access securely to users via a privacy-preserving invite system.

Field testing and provisioning. To quickly deliver CTs to censored users, we developed a streamlined pipeline for field testing and deployment. We collaborated with a small group of technically proficient users to evaluate each CT's effectiveness, gathering both qualitative feedback and quantitative metrics. CTs that proved successful were then deployed by Avos. Both our initial field tests and Avos's rollout were deliberately small-scale and targeted, focusing on human rights organizations and media outlets. We employed a fractal distribution approach, in which the Avos team securely shared QR codes and invite links via end-to-end encrypted channels with trusted CSO contacts, who subsequently distributed them to individual users.

Measuring tunnel health. To measure and evaluate our circumvention techniques (CTs), we built a monitoring backend that collects field-testing reports through a REST API and presents aggregated data on throughput, ping times, retransmission rates, and download latency on provider dashboards using Prometheus, Grafana, and Plotly. We plan to enhance these tunnel-health metrics by incorporating advanced telemetry, automated and manual diagnostics, and active measurements. These improvements will better elucidate questions such as: How usable is the tunnel? What bandwidth is available? How long does the tunnel remain operational? Which circumvention strategies are most effective, and how long can a specific endpoint be reliably used?

Key findings. This report summarizes our field-testing results, which demonstrate that all tested protocols effectively provide Russian users with open-internet access — Avos users included. We're especially encouraged by the combination of our Hopping Pluggable Transport with QUIC, which distributes traffic across multiple IPs and ports while mimicking normal encrypted web sessions, making classification and blocking far more difficult. This approach aligns with emerging technologies like INVISV's MASQUE implementation, which enables tunneling of TCP/UDP traffic through web servers and services using HTTPS. We see this as a key direction for future work.

Our rapid field tests, iterative refinements based on real-world feedback, and targeted deployments have proven to be an effective way to evaluate censorship-evasion strategies.

Because these results come from small-scale provisioning, larger-scale deployments will be necessary to confirm performance under heavier usage.

Retention of volunteer field testers is difficult. We are exploring ways to incentivize field testing. For instance, for the most recent round of tests, we paid attention to properly explain to our tech savvy pool of testers what kinds of protocols we use, how they function, why this is an innovative and important effort and how they can contribute to improving it. We showed some of the results from the first rounds of tests and explained how they informed the mobile client development. This has helped to re-involve people from the first rounds of tests who could see the value of their contributions. We also offered them free invite codes for Avos VPN to share with their friends and family. Furthermore, we will expand the pool of testers beyond tech savvy users. For example, testing capabilities could be integrated into the main VPN app, developed as a separate application, and/or enabled only in testing builds. We will collaborate closely with the Avos user base in Russia to design these new features, ensuring we find the optimal strategy that meets their needs and preferences.

Collaboration to fight censorship. We aim to enhance monitoring of circumvention technologies, while also making the resulting data shareable and the tooling accessible to other bridge providers. To this end, we've built a parsing pipeline that translates our field-test results into an OONI-compatible format and created specifications for formatting and submitting aggregated tunnel metrics to the OONI collector. Aggregating data across multiple bridge providers will improve our understanding of censorial tactics and help identify effective circumvention techniques.

Collaboration in both design and implementation is crucial, and we have initiated discussions with folks from OONI, VPN Generator, DPI Inspector, Amnezia, and Mahsa Server. An important design consideration, for example, is our initial specification for timing of data publication. Releasing tunnel metrics prematurely can jeopardize effective circumvention techniques, and affected communities may prefer to keep this data private for a period of time. This underscores the need to involve users from censored countries early in the design process, co-develop solutions alongside them, and carefully manage the granularity and timing of published measurements —potentially including embargo periods to protect sensitive information.

Use obfsVPN. All circumvention techniques we've tested are ready for integration through our obfsVPN module. ObfsVPN contains a Go package offering server and client components that support variants of the obfs4 obfuscation protocol. It is designed to function as a drop-in transport for any UDP-based tunnel protocol, such as OpenVPN or WireGuard, though it can also serve other general purposes.

For testing, these technologies have been embedded in a Docker-based solution for the server and the client, and later deployed as part of the LEAP VPN stack.The full documentation is here, tutorials here and the architecture can be accessed here.

We will continue measuring and evaluating our production and test VPN endpoints to improve the effectiveness of our current circumvention tools. In addition, we plan to expand testing to more technologies, including MASQUE-based proxies and obfuscated WireGuard.

Circumvention Field Testing

Methodology

Our design methodology included a first step of field testing with a panel of tech-savvy users, able to use Docker, run automated tests and manual tests on a regular basis.

Our field tests combine qualitative and quantitative data. The quantitative part includes measuring packet loss, up- and download bandwidth, latency as well as baseline values that provide information about the network conditions under which the test is running. Qualitative data from the user describes how the tool and speed feel.

The enrollment happened in multiple waves. We invited trusted testers that were recommended to us, or that we could easily verify. Users communicated with one person from the team using end-to-end encrypted messengers (Signal or Delta Chat).

Each tester received a detailed installation and testing guide in Russian language, with a personalized URL for a personalized CryptDrive folder that contained the Docker image, as well as the testing guide and README. All tester data (such as logs or qualitative feedback) was shared using E2EE channels and anonymized using a code name. No personal details of testers were shared or stored.

Analysis

We have released, deployed, measured, tested and evaluated four circumvention technologies: obfs4 + KCP, Hopping Pluggable Transport + obfs4, QUIC, and Hopping Pluggable Transport + QUIC. For testing, these technologies have been embedded in a Docker-based solution for the server and the client, and later deployed as part of the LEAP VPN stack. The full documentation is here, tutorials here and the architecture can be accessed here.

The choice of the four CTs and their design is based on analysis and research of existing obfuscation techniques and the analysis of the state of traffic filtering in Russia. For many years, fully encrypted transport protocols like obfs4 have proven effective in bypassing Internet censorship. Obfs4 (short for obfuscation protocol version 4) is designed to make network traffic look like random noise, making it hard for Deep Packet Inspection (DPI) systems to recognize and block. Its main strength lies in the difficulty of classifying its traffic — since it is indistinguishable from high-entropy random data, blocking it risks overblocking legitimate encrypted internet traffic. This trade-off has helped obfs4 remain viable in countries with strict censorship regimes like China, Iran, and Russia.

However, obfs4 bridges have been targeted by Roskomnadzor since at least 2019, with a significant spike in interference observed in December 2021, when the Russian government began a broad crackdown on the Tor network. Blocking was primarily done through IP-level blacklisting of known Tor bridges, including many using obfs4. This became possible because Russians were mainly using the GetBridgesBot in Telegram and the censors could list and block the bridges distributed via this bot. In some cases, DPI systems deployed by Russian ISPs were used to identify obfs4 traffic based on entropy and connection patterns. Currently, obfs4 remains partially usable in Russia, especially with private (unlisted) bridges, which are harder for censors to discover. Public bridges, however, are often unusable within the country.

Our own research and measurements, as well as existing research on censorship shows an increasing evidence of entropy-based traffic classification being used in the wild — where high-entropy traffic is flagged as suspicious, potentially indicating a tunneling protocol. Censors also analyze connection characteristics such as:

- The duration of client connections (short vs. long-lived)
- The packet distribution pattern (e.g., bursty vs. steady flows)
- The volume of traffic associated with a specific server

By combining these indicators, censors can often identify circumvention proxies over time, particularly when throughput levels exceed typical thresholds for common web usage.

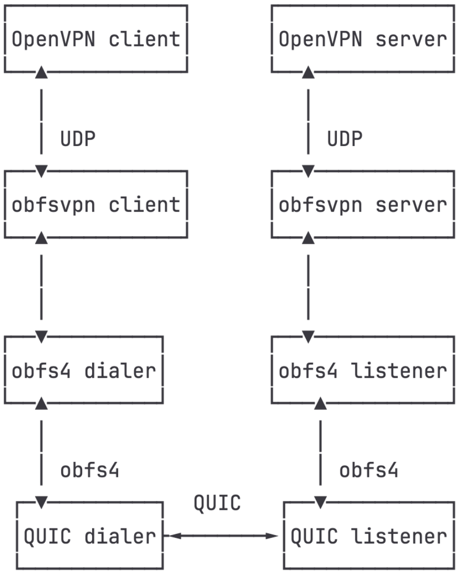
Obfs4 is inherently tied to TCP, which makes it vulnerable to TCP-focused DPI systems. To counter this, we developed obfs4+KCP — a combination that encapsulates obfs4 traffic inside KCP, a lightweight, reliable UDP-based transport layer. KCP is commonly used for real-time applications like gaming and streaming in East Asia, which gives obfs4+KCP the advantage of blending into legitimate UDP-heavy traffic patterns.

While KCP is not widely used in Russian networks, UDP traffic as a whole is often subject to throttling or blocking, particularly when it doesn’t match known service profiles (e.g., gaming or VoIP). However, because most Russian DPI is still optimized for TCP, using KCP can help obfs4 traffic avoid certain classification systems. It has not been observed as a specific blocking target in Russia yet. However, on some Russian ISPs, non-standard UDP traffic is either deprioritized or dropped, especially during periods of heightened network surveillance (e.g. during political protests).

Despite protocol-level evasion, connection behavior still poses a risk. To address this, we introduced Hopping PT— a port and IP hopping transport that fragments and redistributes tunneled traffic across a randomized (yet deterministic) sequence of ports or even multiple bridge IPs making it harder for censors to apply static rules based on known endpoints. Hopping PT operates orthogonally to other transports, meaning it can be layered over obfs4, obfs4+KCP, or QUIC to increase resistance against correlation attacks and port-based filtering.

In Russia this method has not been widely deployed or documented in the wild besides our project, but it directly counters the type of static port blocking commonly used by Russian ISPs. For example, Tor users have frequently reported port-specific interference, where only connections over common Tor ports (e.g. 9001, 443) are blocked. By constantly changing the surface of the connection, Hopping PT avoids many of the basic filtering rules currently in place in Russian networks. As our observations and user research with Russian NGOs have proven (and also confirmed by data from sources such as OONI explorer or IODA), regional blockings are increasing in Russia. Hopping PT technique is particularly promising in circumventing regional blacklists, where blocking rules are often based on observed user behavior and target a narrow range of bridge IPs or ports.

The QUIC protocol, originally developed by Google and now standardized by the IETF, represents the future of modern web traffic. QUIC combines encryption, multiplexing, and connection establishment into a single UDP-based protocol and is already used by platforms like YouTube, Facebook, and Google Search. Although QUIC traffic is currently blockable due to limited adoption in some regions, this will become increasingly impractical as more services migrate to it. Our QUIC-based tunneling protocol leverages this evolution: rather than hiding traffic through obfuscation, it aims to blend into normal web activity— a "camouflage" strategy instead of "invisibility".

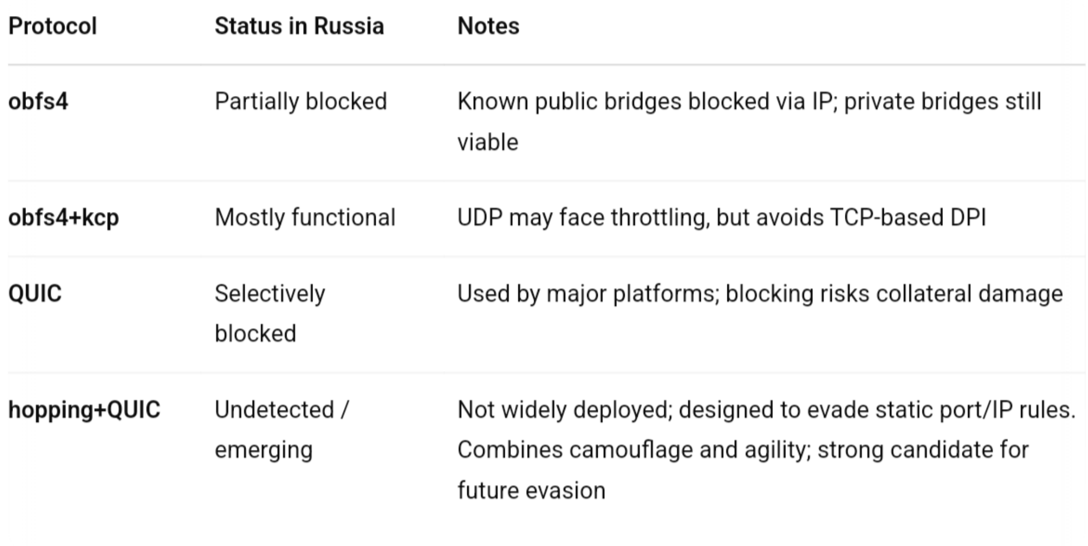


** ObfsVPN's current network stack scheme when running in QUIC mode.*

As early as 2021, some Russian ISPs began blocking or throttling QUIC connections to major services like YouTube in an attempt to degrade performance. This was part of broader efforts to pressure Western tech platforms. However, because QUIC is increasingly used for regular web traffic, blanket blocking carries high risk of collateral damage. Nonetheless, Russia has experimented with selective QUIC blocking using domain name-based filtering and SNI inspection. Our QUIC-based tunneling protocol is making use of a different circumvention approach and attempts to blend in into modern web traffic instead of trying to be "unclassifiable".

The use of Hopping PT in conjunction with QUIC opens a promising path: distributing traffic across multiple IPs and ports while mimicking mainstream encrypted web sessions makes classification and blocking far more difficult. In Russia this hybrid approach is not yet widely detected, as it blends QUIC's modern encryption profile with Hopping PT's agility. There is currently no known classifier targeting this specific combination. It sidesteps both traditional entropy-based filtering and static rule-based blocking. As QUIC adoption grows, hopping+QUIC becomes harder to isolate without risking disruption to legitimate services. This approach aligns with emerging technologies like INVISV's MASQUE implementation, which enables tunneling of

TCP/UDP traffic through web servers and services using HTTPS. We see this as a key direction for our future work and integration.

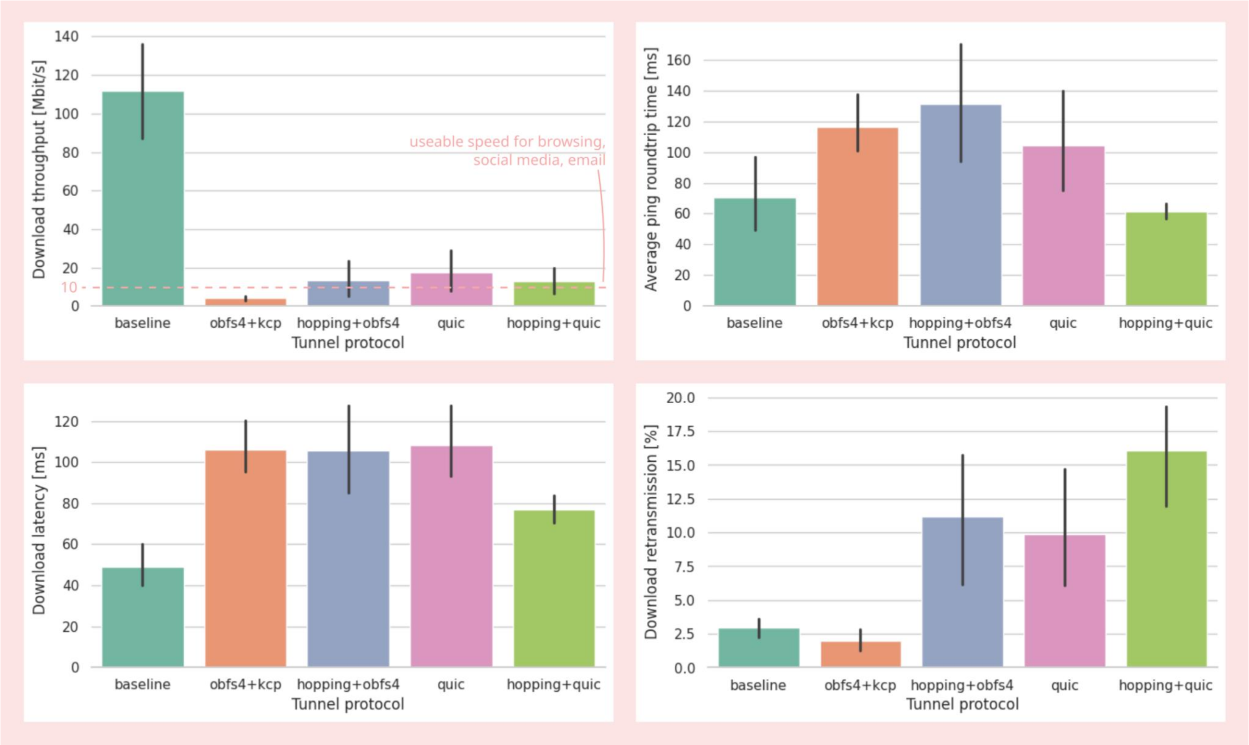


Protocol	Status in Russia	Notes
obfs4	Partially blocked	Known public bridges blocked via IP; private bridges still viable
obfs4+kcp	Mostly functional	UDP may face throttling, but avoids TCP-based DPI
QUIC	Selectively blocked	Used by major platforms; blocking risks collateral damage
hopping+QUIC	Undetected / emerging	Not widely deployed; designed to evade static port/IP rules. Combines camouflage and agility; strong candidate for future evasion

** Blocking status of CTs in Russia*

Results

To date, we have involved a total of 48 users in field testing and conducted 200 tests across 8 Russian regions and 30 ISPs.



** Baseline is a plain connection without VPN. Obfuscation protocols are on top of OpenVPN*

Of particular interest was the better performance of QUIC, which consistently outperformed KCP in throughput. Overall QUIC seems to be a more mature protocol and offers better reliability properties. To give an example, in contrast to QUIC, KCP is missing control messages for fine tuned connection management. The QUIC measurement results are especially encouraging as LEAP plans to integrate the MASQUE protocol, which is built on top of QUIC. Even when using hopping — where connections are frequently rotated across IP:port tuples — QUIC maintained a similar throughput despite higher retransmission rates. We expect QUIC to be a core component of future circumvention strategies.

As seen in the chart, initial field tests of obfs4+KCP revealed surprisingly poor throughput, prompting immediate investigation. This led to the discovery of a critical misconfiguration in the KCP setup. Once addressed, performance improved, with internal benchmarks exceeding 20 Mbit/s (see reference). This rapid feedback loop — testing in the field, identifying bottlenecks, iterating on the implementation, and validating improvements — affirms our development methodology. Test users reported noticeable improvements, and additional rounds of field testing will provide further insight and refinement.

Retransmission data create useful insight clarifying that the obfs4+KCP bottleneck was not tied to packet loss or retransmissions, but purely to misconfiguration — an issue now resolved. While hopping+QUIC showed higher retransmission rates, these did not translate into significant performance loss. In the case of hopping, elevated retransmissions are expected due to its design, which closes and reopens connections across different IP:port pairs. This architecture is built to ensure continuous packet flow by always maintaining at least one live connection before terminating an older one. Although there's room to optimize this process further, the high retransmission rate had little impact on throughput — an encouraging result for resilient, connection-hopping strategies

In our analysis of latency across the tested setups, we observed no significant differences among the various configurations. Notably, QUIC demonstrated a slight performance advantage over both obfs4 with hopping and obfs4 with KCP. However, it is important to consider that this performance difference may be influenced by other factors, such as the geographical distance between the field testers' locations and the bridge. Additionally, the user set was not consistently the same throughout the testing period, which further explains the observed deviations in latency results.

Avos Provisioning

The second phase of our project focused on deploying a new private provider and testing of an Android client that uses the various circumvention technologies. For the mobile client we chose to use our multi-provider client, Bitmask (see repository, a public mirror can be found here). Bitmask is used by different user groups, with a wide global distribution. It is a generic VPN client that can be used with any VPN provider based on LEAP's VPN stack. Bitmask and the LEAP VPN stack have been audited in 2022 and in 2024 by external security labs and LEAP has always responded in a timely manner to raised security issues.

Through long term trust relations, we pass invite codes to CSOs. Their users are able to download Bitmask from Google Play store, F-droid, or an APK and scan the QR code from the invite or directly input the URL from the invite. With Bitmask, both the API discovery and the VPN tunnel are obfuscated. Bitmask contacts dedicated bridges via Snowflake or a proxy to acquire the correct configuration information. Bitmask is then able to create tunnels through the gateways to bypass censorship.

The name for our VPN service - *Avos network* derives from Russian word "avos'", a word that is proper to Russian culture and conveys several important meanings: the tinkering, the "good luck" and the unpredictability. The latter is connected to the nature of Russian censorship strategies, that constantly evolve and become more and more sophisticated. With this new branding we have approached the selected organizations to distribute invite codes.

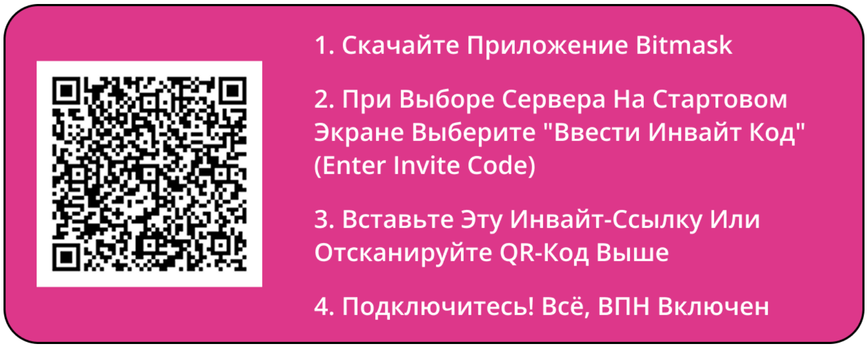
Trust has and always will continue to be an important challenge. With a new provider, users need to know that it's OK to trust it. We are building the trust by working through direct trusted contacts with CSOs who then extend this to their users. This also involves providing technical explanations to our contacts, about our obfuscation strategies and how they are different from other VPN approaches. Our power users are tech-savvy, have a good understanding of how VPNs work while regular CSO users are not, and require additional explanations. This work continues.

We have distributed codes to multiple CSO's including a major human rights defense organization, a LGBTQ rights organization, a digital rights organization, a human rights focused legal team, an environmental group, and a digital help line. The human rights organization reached back to us and requested to reuse our source code to deploy their own version of Avos, due to the strict security policies of their group, that rely only on self-hosted solutions. We are following up with them on the deployment process and we think this is a very interesting development, because we have initially designed our CTs to be reusable by other teams and we encourage their further propagation.

Some of these groups are starting to use our service and shared positive report backs. For example: "We could connect much faster than to (VPN service name) or (VPN service name), whose bridges are overloaded. Thanks for the invites". And “F*#k what a great thing humans are doing for humans. F*#k. Many thanks to these artists of camouflage.” They also underlined the easy-to-use UI of the Bitmask client and the opportunity to shift between CTs. We will roll out invite codes to an additional 8 CSOs. For these organizations it was important to find out that Avos is based on a program of research and testing, and also that our project is not public.

Indeed, one of the main challenges for this phase was to roll out and distribute VPN invite codes securely. While we initially planned to publish them on decoy websites, we abandoned this technique after some user research. We discovered that Russian censors were actively monitoring and blocking web pages and Telegram bots and other distribution channels containing VPN codes and any mentions of circumvention technologies. Our researcher conducted interviews with several Russian digital rights experts who confirmed that since the last year the civil society organizations switched to smaller, self-hosted solutions (for example, Amnezia VPN), while bigger commercial VPNs were mainly used for personal reasons and were discovered and blocked faster.

This is where the Avos VPN finds its legitimate place and its user base. With a focus on human rights activists and media organizations, we implemented a "fractal distribution" strategy. We established a communication channel from the Avos team to trusted representatives of the CSOs and distributed QR codes and invite links over end-to-end encrypted channels. The CSO representatives then distributed them further down to the end-users.



1. Скачайте Приложение Bitmask

2. При Выборе Сервера На Стартовом Экране Выберите "Ввести Инвайт Код" (Enter Invite Code)

3. Вставьте Эту Инвайт-Ссылку Или Отсканируйте QR-Код Выше

4. Подключитесь! Всё, ВПН Включен

* The invite link format (link purposely broken):   obfsvpnintro://12.34.56.89/?cert=ffXUm5B7JAJP%2B0Gc2zHZX0I2RwXuwR0jz937PrpR%2FNopWkuJFkBQwN%2Bza4ib%2BXvaPrFqOg&fqdn=api.private.network&kcp=0&auth=avos_zYOreyzAh%2Bx79jj

* where 12.34.56.89 is the IP of a proxy which routes the users' requests to the configuration API. It is not possible to use the proxy for any other traffic than contacting our API. These IPs will change over time and clients will get updates about gateway changes via the configuration API.

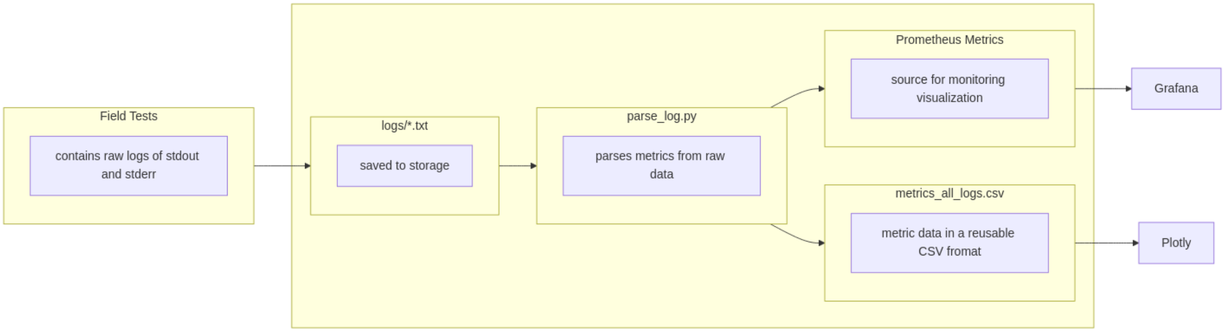
As with all VPN products, we have seen that users move their trust from their ISPs to the operators of the VPN. Avos VPN system is built with relative anonymity in mind. There are no identities tied to single users, no accounts, no direct payment, no logging beyond short term anonymized operational logs that help to monitor the health of the system. Logs users see in their app are kept on the device and are not sent, unless users decide to send them.

For now Avos is focusing on censorship circumvention and providing protection against the ISPs. While illegal targeted attacks could be done in case Russia wants to invest the resources, it is quite unlikely because Avos is a small-size R&D VPN service that is focused on offering VPN to specific CSOs. We bear in mind that a risk pertains to be targeted as an activist just by using a particular network route. This is why our further steps are to blend in within a general (possibly commercial) VPN offering.

Tunnel Measurement Tooling

Bridge and Tunnel Monitoring

In order to measure and evaluate our deployed circumvention technology we have built a monitoring backend that accepts field testing reports via a REST API. The backend contains a parsing pipeline for incoming field tests. Upon uploading the test results are pushed to Prometheus and visualized in a Grafana dashboard for live monitoring purposes. Simultaneously they are transformed into a CSV which is used as the basis of a Plotly dashboard allowing more general insights into the whole dataset.


Field Tests
contains raw logs of stdout and stderr
logs/*.txt
saved to storage
parse_log.py
parses metrics from raw data
Prometheus Metrics
source for monitoring visualization
metrics_all_logs.csv
metric data in a reusable CSV fromat
Grafana
Plotly

** Field test parsing pipeline*

Currently we measure throughput, ping times, retransmission rates, and download latency.

Standardized formatting and data sharing

Our goal is to not only enhance monitoring of our circumvention technologies, but to do so in a way that is shareable and extensible. To this end, we built a parsing pipeline that translates our field test results into an OONI compatible format. To do so, we designed a generic format that describes a way to send aggregated data to either the VPN provider or to OONI. This specification allows different granularity regarding the shared data. While the VPN provider can have a very detailed view on a single test including test timestamps, test user ID, client regions, bridge IPs and protocols used as well as detailed information about the performance of the obfuscation protocol, the format also enables us to aggregate and anonymize data points so that it can be shared with 3rd parties in a privacy preserving manner.

Work is under way to turn this into a sharable library that will produce the desired OONI-compatible JSON format. As a starting point of this development we have integrated our tunnel-telemetry library into bitmask-core which in turn is used both by our Android and Desktop client.

Our future plan is to propose to the censorship circumvention community the development of a commonly shared data protocol and an accompanying library. This initiative aims to enhance the interoperability and agility of VPN projects, facilitating more effective knowledge sharing among stakeholders. We are therefore currently reaching out to open-source VPN projects in the field.

Lessons Learned

Keeping volunteer testers engaged is tough. Participation often drops off over time, especially without incentives. We're exploring options to sustain engagement, like providing invite codes for Avos, and providing measurement data and more explanations about the functioning and evolution of our protocol stack which helps motivate tech savvy testers, as they can see the value of their efforts. We're also working on expanding our tester base by building opt-in testing tools directly into the VPN app — so less tech-savvy users can easily provide data for analysis.

At the end of 2024 our testers reported that CryptPad was unavailable from Russia without a VPN. As we were relying on CryptPad for distributing Docker clients and collecting feedback and logs, the blocking became an extra obstacle for testers to access or share information. While this slowed engagement, a recent wave of tests showed that motivated testers still could update the client and run the tests.

All tested circumvention techniques (CTs) worked at our targeted scale. Our results show that none of the protocols were blocked when used by a limited, targeted user base. While this approach works well at a small-scale, it remains to be seen whether the same holds true as usage increases. We plan to provision these bridges to a larger public user base, and expect that one of the larger LEAP VPN providers, with thousands of users in Russia and other censored regions, will deploy new bridges using the latest CTs. These rollouts will help test if a mid-scale use case increases the risk of detection and blocking and if our Hopping PT will ward against this threat. Currently a mid scale LEAP VPN provider is experiencing DDoS-like issues with their public obfs4 bridge. Eager for an explanation, we are helping to test and troubleshoot.

QUIC is showing real promise. Its performance and resistance to blocking are encouraging. Based on these results, we're planning to test — and potentially provision — MASQUE, which builds on QUIC and offers even greater obfuscation potential.

Tunnel telemetry is already useful and will be even more valuable as we expand it. We're expanding the ways to measure metrics like failure rates, performance indicators as well as aspects concerning a censor's infrastructure and methodology such as the client's ASN, the region and the time stamp as optional, user privacy preserving in-app features. We structure the data in a shareable format which will allow for deeper analysis and better collaboration across the circumvention tech ecosystem.

Censorship evasion is an ever-evolving challenge, with censors employing techniques like deep packet inspection to block protocols, alongside traffic analysis and active probing to identify and block VPN endpoints. Even in countries with centralized blocking infrastructure, anecdotal evidence shows regional variations in how censorship is applied, often influenced by political circumstances. This underscores the need for broad monitoring coverage and adaptable strategies to continuously study the effectiveness of circumvention techniques.