Censorship resistance à la carte

Mechanism	Evades	Notes on evasion	Implementation difficulty	Implementation notes	Cons	Viability	References
HTTP/3 (QUIC)	HTTPS (SNI)-based filtering	The SNI is obfuscated (but not encrypted) in QUIC. This makes implementing the HTTPS requests in HTTP/3(QUIC) a good short-to-medium term tactic to evade SNI-based censorship. DNS and IP based censorship will be unaffected.	Easy-medium	Switching to HTTP/3 should be of easy-medium difficulty. Libraries like Cronet are supporting it out of the box.	Iran could decide to block all HTTP/3 traffic or UDP traffic. There is some conflicting evidence whether Iran currently targets QUIC traffic, but the most recent research paper suggests that this not the case. Does not evade blocking of IP address endpoints, that will still need a separate strategy.	Medium in the long term (Iran could censor based on SNI in QUIC flows in the future. However, there is no real downside of defaulting to HTTP/3, except if Iran tries to block all QUIC traffic. Iran has blocked QUIC traffic in September 2022 and December 2023.	A Quick Look at QUIC Censorship | OONI Technical multi-stakeholder report on Internet shutdowns: The case of Iran amid autumn 2022 protests | OONI QUIC blocking in Iran - Cloudflare Radar Web censorship measurements of HTTP/3 over QUIC
Encrypted Client Hello (ECH)	HTTPS (SNI)-based filtering	With the SNI being encrypted, ECH is a great long-term strategy for evasion of SNI-based filtering in Iran. The only way for the Iran government to make it ineffective would be to block all ECH traffic.	Hard	Implementing ECH on the client-side might be a challenge. Tunnelbear has a small description of how they did it, but Android/iOS libraries don't support it out-of-the-box yet.	ECH-enabled TLS traffic 'sticks out', i.e. it can be selectively targeted and blocked. In the future, Iran could block all TLS traffic that uses ECH. Does not evade blocking of IP address endpoints, that will still need a separate strategy.	High in the long-term (ECH support and deployment is likely to go up steadily. There is no downside of deploying ECH, except if Iran or some other country decides to block all ECH traffic altogether.)	Introducing Encrypted Client Hello (ECH) Encrypted Client Hello - the last puzzle piece to privacy
DNS over TLS (DoT)	DNS-based censorship	DNS queries are encrypted.	Easy	Android and iOS ship with DoT support, and this should be fairly easy to implement as just a TLS request in the code as well.	There is already some evidence some Iranian ISPs are interfering with DoT requests based on the IP and SNI of the DNS sever. We have to select a DoT server instead of the relying on the system/client one (poisoning is still possible!). DoT runs over a specific port (853) and can be easily blocked entirely.	Low viability and need. DoH is a better alternative that achieves the same censorship-reslience with much less downsides.	Measuring DoT/DoH Blocking Using OONI Probe: A Preliminary Study DNS over TLS blocked in Iran | OONI
DNS over HTTPS (DoH)	DNS-based censorship	DNS queries are encrypted. Additional advantage over DoT is that DoH operates over the normal HTTPS port.	Easy	Android and iOS ship with DoH support, and this should be fairly easy to implement as just a HTTPS request in the code as well.	There is some evidence that Iran interferes with DoH requests based on the SNI and IP of the DNS server. This could be easily evaded by running a DoH proxy or less-known DoH server. We have to select a non-censorious DoH server instead of relying on the system one (poisoning is still possible!).	High in the long-term (It is impossible to block DoH without having large collateral damage to HTTPS traffic. Iran could block well-known DoH servers like Google's or Cloudflare's based on SNI and IP address, but one could use less-known or a self-hosted DoH proxy.)	[2202.00663] Measuring the Accessibility of Domain Name Encryption and Its Impact on Internet Filtering Measuring DoT/DoH Blocking Using OONI Probe: A Preliminary Study
DNS over QUIC	DNS-based censorship.	DNS queries are encrypted. Additional advantage (compared to DoT and DoH) is that the server name will be obfuscated in QUIC flows. So destination DNS server matters less than in those situations.	Easy	Support for DNS over QUIC should be increasing in the short term, as it has been recently standardized at the IETF.	The entire blocking of the QUIC protocol will make this infeasible.	QUIC is likely to be deployed widely in the future, making the blocking of the entire protocol infeasible (similar to DNS over HTTPS). The possibility of blocking well-known DoH servers is also low (see notes on QUIC).	RFC 9250 - DNS over Dedicated QUIC Connections

TCP packet segmentation	SNI-based censorship	"By reducing the TCP window size of the SYN+ACK packet, it induces the client to segment the forbidden request. This works because the middleboxes [...] appear incapable of reassembling TCP segments, so once the forbidden request is segmented, it is uncensored."	Easy-medium	This only requires a change to the server, and not the client.	Requires a modification to how the server works.	Viable in the medium-long term: Not many middleboxes are re-assembling TCP requests, so it is unlikely that Iran can begin detecting this circumvention method overnight.	Come as You Are: Helping Unmodified Clients Bypass Censorship with Server-side Evasion
TLS Record Fragmentation	SNI-based censorship	Works similarly to TCP packet fragmentation, but relies on splitting the handshake (specifically the SNI) into two TLS messages. Most TLS servers support fragmented TLS messages.	Easy-medium	Requires a modification to the client (and the server should support fragmented TLS records).	Requires a modification to the client (and the server should support fragmented TLS records).	Viable in the medium-long term: Not many middleboxes are re-assembling TLS messages, so it is unlikely that Iran can begin detecting this circumvention method overnight.	Circumventing the GFW with TLS Record Fragmentation | System Security Group
Domain fronting	SNI-based censorship	Domain fronting has one pre-requisite: we need to find a domain hosted on the same hosting service. Say our domain is blocked.com and another domain hosted on the same service is notblocked.com. How domain fronting works is: use notblocked.com in the SNI, but in the (encrypted) HTTP request, use blocked.com in the HOST header. The service will forward the HTTP request to blocked.com	Easy-medium	This is only a trivial change to clients.	There is a need to find (or host) an innocuous domain name on the same hosting service. Domain fronting stops working (or is not required) if ECH or ESNI is in play. Domain fronting is not supported by most major cloud providers (Google, Amazon, Cloudflare stopped in April 2018, Azure stopped in 2022). Fastly may also drop supporting it by February 2024.(*) We will need to find a service that is promising to offer domain fronting in the long-term.	Low-medium viability in the long term. Requires sticking with a service provider. There is no guarantee that the service provider will continue to support domain fronting.	DEF CON Safe Mode - Erik Hunstad - Domain Fronting is Dead, Long Live Domain Fronting Using TLS 1.3 Generally available: Block domain fronting behaviour on newly created customer resources | Azure updates Fastly announces plans to block domain fronting in February 2024 · Issue #309 · net4people/bbs · GitHub [2310.17851] Measuring CDNs susceptible to Domain Fronting
Domain hiding	SNI-based censorship	Domain hiding is domain fronting 2.0, and relies on the use of Encrypted SNI. We use BOTH ESNI and unencrypted SNI. In the SNI, we use notblocked.com, while in the ESNI, we use blocked.com. In the HTTP HOST, we use blocked.com (version 1).	Hard	True target can be hosted anywhere. DNS of both must be run via Cloudflare.	Needs clients to use a modified TLS library that is capable of sending both ESNI and SNI.	Low viability in the long-term. Since ESNI efforts were superseded by ECH, there may be little long-term support for this strategy in libraries and on cloud services. Cloudflare now has stopped support for version 1 (which is true 'domain hiding'). The strategy will also stop working if ESNI is blocked entirely for Iran.	DEF CON Safe Mode - Erik Hunstad - Domain Fronting is Dead, Long Live Domain Fronting Using TLS 1.3
		We could also leave the SNI field unpopulated. (version 2)		Cloudflare managed DNS is free.	True domain must have DNS provided by the same service provider.		GitHub - SixGenInc/Noctilucent: Using TLS 1.3 to evade censors, bypass network defenses, and blend in with the noise

Last updated: 23 April 2024. Only meant for limited circulation. Thanks to Alex Linton and the Session team, and participants in the workshop organised by ICD on 18 April 2024. Contact: Gurshabad Grover <gurshabad@protonmail.com>