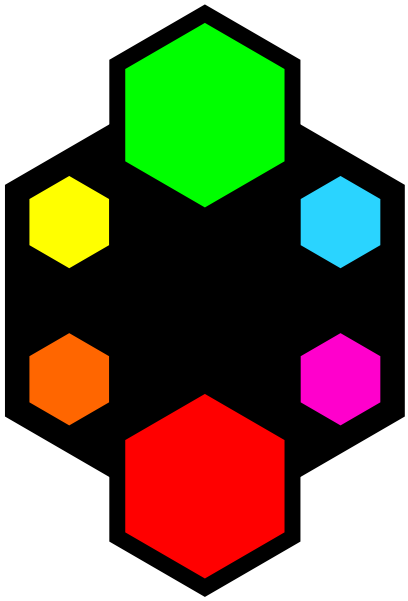


Indra Routing Protocol White Paper

Programmable onion routing distributed virtual private network protocol with anonymised payments to create scaling incentives.

David Vennik September 2022

> Markdown format of this document is created with Typora which renders the sequence and other graphs found in this document correctly. The PDF format may lag from the current state of the markdown document.

Abstract

The state of counter-surveillance technologies has remained largely unchanged in the 20 years since the inception of the Tor network. The primary use case has always been obscuring the location information of users from clearnet sites, and the more it has been used for this purpose, the more hostile clearnet sites have become towards this network, due to its frequent use to launch attacks on web services.

With the increasing amounts of value being transported in data packets on the Internet since the appearance of the Bitcoin network, the need for eliminating the risks of geographical correlation between payments and user locations continues to rise.

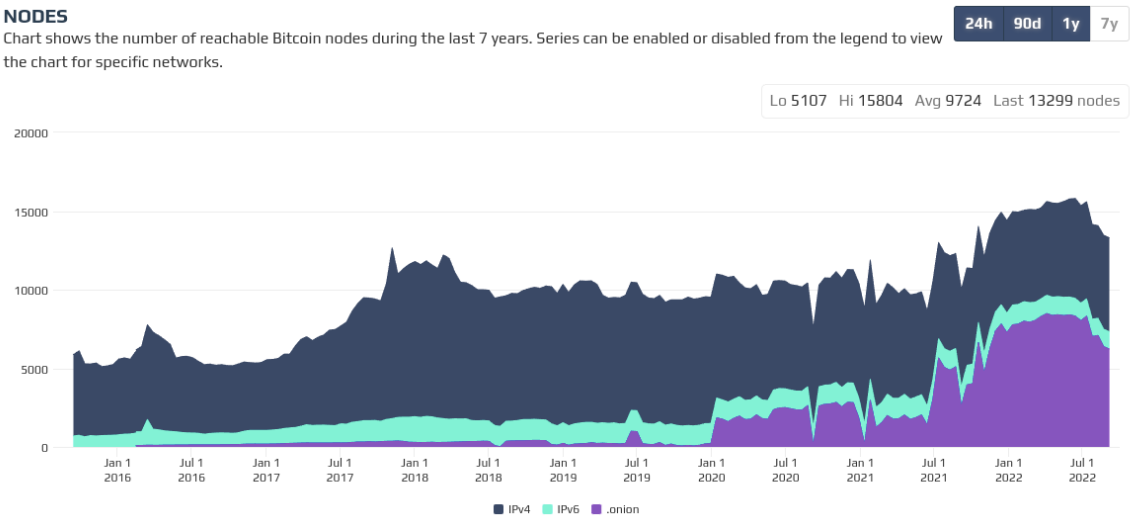
However, without any way for users to pay routers without creating an audit trail, the networks have a severe scaling problem in that in anonymising data, there is an increase in privacy with the larger number of nodes and users, and thus well heeled attackers have largely been able to keep pace and pluck off high value targets, such as the Carnegie Mellon University .

Thus, it is the central thesis of this paper to demonstrate how decorrelation between payments and session usage can be achieved and create a marketplace in routing services which can economically increase to a size that is beyond the capabilities of a state sized actor to fund an attack.

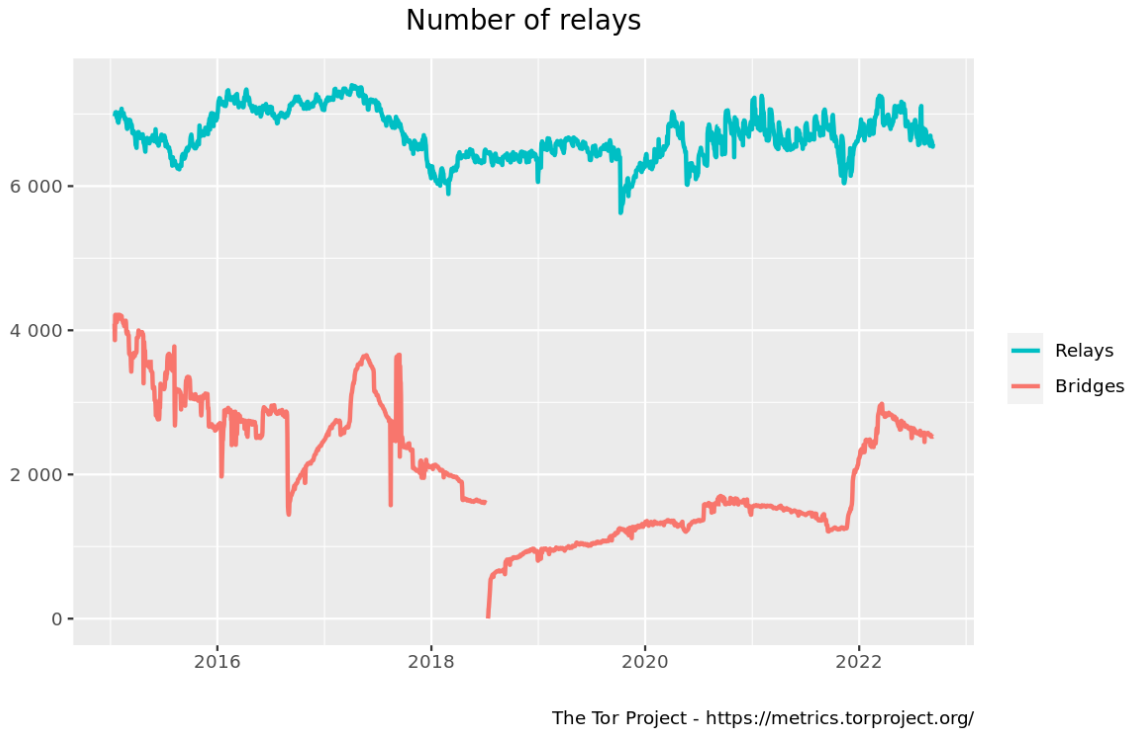
Indra creates mechanisms for anonymous purchase of chaumian vouchers used to initiate traffic sessions with router nodes, which then compensates routers for their running costs, and further, focuses on hidden services and Bitcoin/Lightning (and potentially other Bitcoin related systems) in order to reduce the attack surface from large actors who have thus no open justification for censoring the network.

Tor isn't Scaling, but Bitcoin Needs Onion Routing

For comparison, this is Bitcoin's node count:



Versus Tor in a comparable period:



It is not hard to see: Tor is not growing, it's flatlined. Bitcoin is growing. Not only that, you can also see that onion routing is forming an increasingly large component of Bitcoin connectivity.

Goals of Indra Routing Protocol

Three key elements of the Tor protocol make it less than desirable in general.

1. the establishment of circuits is quite slow, taking a large number of steps to "telescope" into a channel. Source routing would be preferable.
2. once a circuit is running, when it fails, the failure is opaque to the client side, and there is no way to provide a latency guarantee or connection stability. An anonymised probing mechanism would help fast recovery and avoid timeouts.
3. Three, as above, there is no profit motive to drive expansion of router capacity, and as such it has definitively flat-lined, and there is clear signs that a growing number of nodes are in fact operated by Bitcoin users.

Indra aims to provide a source routing mechanism with a feedback mechanism for determining unresponsive routers in the path and a payment mechanism integrated with the Lightning Network that produces tokens that clients can use to construct routing paths without interactive key negotiation.

Anonymised Session Purchase Protocol

The initial purchase of a routing session presents a chicken and egg situation. Without an existing circuit, how does a client node acquire a token to use for creating hops in an onion route?

For this, we borrow from BOLT#4 which we cannot use by itself as it has no provision for returning an arbitrary package of data, nor the notion of an interacting midpoint in the loop, with the chain going back to the sender.

As distinct from this Lightning onion protocol, we use ed25519 for signatures and curve25519 for the ECDH, which is then used with AES-GCM to encrypt packets, with a two factor re-keying for each new packet (up to 64 Kb per packet).

Session Tokens

Session tokens are an arbitrary random value that must be present in the header of encrypted data and is hashed in a chain to provide a counter for the packet sequence within a session. This functions also as authentication for the session, and is then also used to hash with the secret key in the ECDH to derive subsequent ECDH public keys to pass with the sequence hash chain element to provide a constantly changing cipher that is invulnerable to reverse derivation (newer cipher cannot be used to derive past ciphers).

Path Hop Acknowledgements

In order to ensure the session purchase protocol is properly executed, in each layer of the onion there is a multi layer onion message that is to be sent back to the previous step in the path, which contains further steps backwards until the buyer, as well as a specified cipher to use on the Lightning Network (LN) payment confirmation to carry back to the buyer.

As each step proceeds, the router receives a payment from the previous via LN, and then the purchase onion message, unpacks their acknowledgement onion and returns it with the encrypted acknowledgement of their forwarding of the payment to the next hop.

This reaches the router that is selling the session, which then forwards the encrypted session key and pays forwards the two remaining hops, again each step returning the acknowledgement onion, and in this way the buyer can expect 5 acknowledgements to be successful and then receive the session key from the 5th node in the circuit.

The acknowledgement onions are constructed so that nodes do not know what step they are, so each has space for 5 steps, which are masked using methods as described in BOLT#4.

Source Routing

As distinct from TOR, IRP uses source routing, thanks to the magic of the session tokens and ECDH, means that in the event of a route path failing, a new path can be generated when a timeout occurs, and in addition, a timeout triggered reverse path diagnostic based on the Path Hop Acknowledgement onions above, and for time sensitive interactive applications, can be used at every step to ensure the moment one hop latency exceeds a threshold the source routing algorithm can then swap out a different node in the route and provide a strong latency guarantee.

Latency Guarantee and Path Timeout Diagnostics

Some applications are very time sensitive. Real-time interactive shared environments such as games can have very serious consequences (to the players) when their connection starts to increase in latency putting them at disadvantage against their opponents, and in general, a sluggishness of the interactivity.

Thus, there is two parameters that can be set on a route, one is a constant on path acknowledgement, which consumes a substantial extra segment of the space used for payload, the other is to turn this feature on when the circuit fails to return through the circuit.

For extra security, a third parameter, can be that the return path used for acknowledgements can be further obfuscated to take two hops in each case back to the sender, which can be randomised for each one.

Circular Paths

One of the key inventions of Indra is the notion of circular paths. These are two hops out to the end point, and two hops backwards, on a different path, for the return.

By using this circular topology, the source can provide a return path that is not the same as the forward path, and when the path timeout diagnostics are not in play, there is no visible reverse path confirmation timing. That is to say, the packets appear to always only be going forward, and no correlation is easily made between, therefore, forward, or reverse paths, which is not the case with telescoped TOR protocol packets, and for most general purposes in IRP are avoided when traffic achieves the intended forward path without excessive latency.

The return paths also serve as a conduit for the endpoint to return data back to the sender, while allowing the sender to dictate this return path.

Dancing Paths

Thus, as well as introducing a mechanism for monitoring the progress of packets through paths, it then becomes possible, due to the source routing strategy, for every single path to be different, aside from the exit point of the path.

Redundant/Fragmented Parallel Paths

In addition, a further feature for future work is the use of Shamir's Secret Shares, as well as Reed Solomon Forward Error Correction to provide a reduction in path length or higher guarantee of messages passing in one try and failing paths to not impede signals if failures are below the RS parameters.