

Centralization risks and mitigation

Risk: Obol hosting the relay infrastructure

Mitigation : Self-host a relay

One of the risks associated with Obol hosting the [libP2P relays](#) infrastructure allowing peer discovery is that if Obol-hosted relays go down, peers won't be able to discover each other and perform the DKG. To mitigate this risk, external organizations and node operators can consider self-hosting a relay. This way, if Obol's relays go down, the clusters can still operate through other relays in the network. Ensure that all nodes in the cluster use the same relays, or they will not be able to find each other if they are connected to different relays.

The following non-Obol entities run relays that you can consider adding to your cluster (you can have more than one per cluster, see the `--p2p-relays` flag of [charon run](#)):

Entity Relay URL [DSRV https://charon-relay.dsrvlabs.dev](https://charon-relay.dsrvlabs.dev) [Infstones https://obol-relay.infstones.com:3640/](https://obol-relay.infstones.com:3640/) [Hashquark https://relay-2.prod-relay.721.land/](https://relay-2.prod-relay.721.land/) [Figment https://relay-1.obol.figment.io/](https://relay-1.obol.figment.io/)

Risk: Obol being able to update Charon code

Mitigation : Pin specific docker versions or compile from source on a trusted commit

Another risk associated with Obol is having the ability to update the [Charon code](#) running on the network which could introduce vulnerabilities or malicious code. To mitigate this risk, operators can consider pinning specific versions of the code that have been thoroughly tested and accepted by the network. This would ensure that any updates are carefully vetted and reviewed by the community.

Risk: Obol hosting the DV Launchpad

Mitigation : Use [create cluster](#) or [create dkg](#) locally and distribute the files manually

Hosting the first Charon frontend, the [DV Launchpad](#), on a centralized server could create a single point of failure, as users would have to rely on Obol's server to access the protocol. This could limit the decentralization of the protocol and could make it vulnerable to attacks or downtime. Obol hosting the launchpad on a decentralized network, such as IPFS is a first step but not enough. This is why the Charon code is open-source and contains a CLI interface to interact with the protocol locally.

To mitigate the risk of launchpad failure, consider using the `create cluster` or `create dkg` commands locally and distributing the key shares files manually.

Risk: Obol going bust/rogue

Mitigation : Use key recovery

The final centralization risk associated with Obol is the possibility of the company going bankrupt or acting maliciously, which would lead to a loss of control over the network and potentially cause damage to the ecosystem. To mitigate this risk, Obol has implemented a key recovery mechanism. This would allow the clusters to continue operating and to retrieve full private keys even if Obol is no longer able to provide support.

A guide to recombine key shares into a single private key can be accessed [here](#) . [Edit this page](#) [Previous Errors & Resolutions](#) [Next Handling DKG failure](#)