Lloyd's Projects for 2nd half of 2020

secp256kFUN!

This my mid-level API for secp256k1 that is targeted towards researchers.
I am very happy with how it's turning out and want to continue to maintain and improve it.
It is necessary for a lot of the ideas I want to experiment with in Bitcoin below.

Implement "Generalized Bitcoin-Compatible Channels"

In a recent paper, Aumayr et al.[1] had a breakthrough idea for a new payment channel state revocation mechanism. Instead of each party having a different transaction so that the transaction identifies the offender, the transactions signatures can do this instead. This means both parties have the same transaction representing a lightning state but different adaptor signatures on it. This removes asymmetric transactions from the protocol in favour of asymmetric knowledge which makes it simpler and more efficient than the current lightning design. The authors provide a proof-of-concept python implementation but I believe this idea is promising enough that it deserves a more thorough implementation and scrutiny.
My plan is base a PoC on rust-lighting and to use secp256kfun to for the cryptographic primitives. The hope is that this can help the lightning research and engineering community evaluate the idea as a potential upgrade path.

DLC oracle spec

Discreet log contracts (DLC)[3] are finally gaining some momentum with multiple organisations and individuals working together to create a BOLT like specification[4] for the p2p contract execution protocol. However, In order for the idea to work people have to run and maintain trustworthy oracles. We need a common specification on how oracles should attest to the outcomes of events and an open implementation for that specification.
Currently, I've got a to-be-released MVP of an implementation in rust that uses BIP-340 signatures (from secp256kfun). The idea is the oracle server is notified of various changes of event states from the outside which lets it announce and publish signatures through HTTP.
My plan is to release the implementation soon and use it to develop the specification for the HTTP API in collaboration with others working on DLCs. More ambitiously I want to eventually bring the idea from my paper "How to Make a Prediction Market on Twitter with Bitcoin"[2] to life in some form.

[1]: https://eprint.iacr.org/2020/476.pdf
[2]: https://github.com/LLFourn/two-round-dlc/blob/master/main.pdf
[3]: https://adiabat.github.io/dlc.pdf
[4]: https://github.com/discreetlogcontracts/dlcspecs