BNS, the Savior of the Internet

November 21, 2021

Philosophy

- BNS must be an extension of current domain name system.
- BNS must be fully distributed and decentralized.
- BNS must be self-organizing and not depend on administrators or centralized infrastructure.
- BNS must be open and permit new peers to join.

Introduction

Unwrap BNS, we can see

- A decentrized domain name system (DDNS)
- A Chord based distributed hash table (DHT)
- A crosschain Decentralized Identifiers System (DID)
- A anonymous network for hidden hosting and traffic mixing

Related work

- GNUnet[2] & GNS
  - GNUnet is a framework for the NG. of Internet protocols
  - GNS is GNUnet's domain name system, which is a decentralized database.
- I2P [3], Tor network
- ENS, unstoppable Domains
- Handsake, blockstack, Namecoin

DDNS

Based on ethereum, Can be resoluted to:

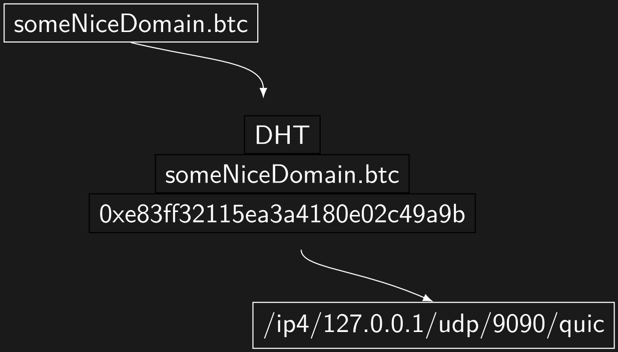
- classic internet URL
- IPFS resources, MultiAddr
- crosschain wallet address
- values of BNS DHT

DDNS details

A ‘domain name‘ is A:

- ERC721 of ethereum
  BNS Domain Name will be present as an ERC721 NFT
- peer-key in BNS DHT
  BNS Domain Name can be also used as web3 DID.

DDNS



someNiceDomain.btc

DHT

someNiceDomain.btc

0xe83ff32115ea3a4180e02c49a9b

/ip4/127.0.0.1/udp/9090/quic

BNS DHT

We use Chord DHT to support:

- ad-hoc message
  - filesharing - instant messaging - hidden routing and anonymous traffic network (ATN)
- anonymous hosting - support multiAddr-DHT key binding
- zkp data transfer & trading
  - based on vertex-perdson commitment

BNS DHT

- stores key-value pairs with values up to (approximately) 64k in size
- works with many underlay network topologies (small-world, random graph), underlay does not need to be a full mesh / clique
- support for extended queries (more than just a simple 'key'), filtering duplicate replies within the network (bloomfilter) and content validation
- provides content replication to handle churn

BNS anonymous Network

related work:

- i2p mixnet
  - a distributed free route mixnet
- tor - DoS and phishing risk.

Why POS

- Preventing DoS attacks [1]
- anti brute force attacks
- Reward DHT Nodes
- Negative Punishment

Solve denial (of services) with POS

```
                                        +----------------------------------+
                                       |                                  |
+-------+ INTRO1  +-----------+ INTRO2 +--------+                         |
|Client |-------->|Intro Point|------->|  POS   |-----------+             |
+-------+         +-----------+        |Verifier|           |             |
                                       +--------+           |             |
                                       |                    |             |
                                       |                    |             |
                                       |         +----------v---------+   |
                                       |         |Intro Priority Queue|   |
                                       +---------+--------------------+---+
                                                        |  |  |
                                             Rendezvous |  |  |
                                               circuits |  |  |
                                                        v  v  v
```

implementation

- onchain domain name register system
- lightweight browser extension - implement light DHT with webRTC - support DNS querys
- full features nodes - get reward from traffic - can host hidden services

traffic proof

$\mathbb{D}$: hash of target data

$[D_0, D_1, \cdots, D_n]$: slides of data

$\mathbb{G}$: an ECC group.

g : a point on $\mathbb{G}$

$\mathbb{D}.g$ is stored on chain

For single request, all participating nodes should proof that

$$\sum_{i=0}^{n} D_i.g = \mathbb{D}.g$$

Reference

asn.
How to stop the onion denial (of service).

GNU.
Reference manual for gnunet version 0.15.4-alpha.1-2-gc5e203bf6.

i2p.
The invisible internet project (i2p).