Proposal
Towards a Framework for DHT Distributed Computing

Andrew Rosen

Georgia State University

May 27th, 2016

What is D^3NS

Distributed Decentralized Domain Name Service

- The goal is to create a secure distributed DNS.
- Requirements
  - Decentralization
  - Authentication
  - Reliable
  - No end user modification

Motivation

- Recent events have demonstrated that centralized authorities are not as secure a previously hoped.
  - There is little cryptographic protection against the subpoena.
  - Poorly constructed laws targeting DNS.
- A distributed approach for authentication is much less vulnerable.

Overview

- Use a Distributed Hash Table (DHT) to organize a P2P network
    - UrDHT
- Use a variant of NameCoin's blockchain to secure shared list of keys and domains.
- Use the DHT to load balance and distribute responsibility for hosting DNS and keys.
- DNS server frontend (PowerDNS)

Distributed Hash Tables

- Means of organizing communication and responsibility in a P2P network
- Each peer is responsible for a verifiable span of hash values
- Facilitates one-to-one communication and one-to-many communication

UrDHT

- Abstract DHT backend
- Handles:
  - Organizing nodes into a DHT or other DHTs
  - Plugin Services
-
- Subject of other research

UrDHT Details

- DHT organization mechanism.
- Uses Voronoi regions on an n-dimensional torus to assign responsibility.
- Can define how to compute the regions to emulate almost any DHT topology.
- Node responsibility:
  - Node is responsible for its space, defined by its neighbors.
  - If a node leaves/fails, each neighbors assumes that it is responsible until corrected by maintenance.

Fault Tolerance

- Churn creates a period where i/o can fail. With UrDHT:
- Reads of backed up data are successful.
- Writes to the region are successful.
- Reads of new data are vulnerable until stabilization (<2 sec currently).
- This means a much smaller window. Writes never fail.[1]

[1] They may occur out of order

Blockchain

- Based on the blockchain verification of Bitcoin
- Allows for a shared, immutable and secure public records
- One block can include the validation of a new server's public key
- One block can include a DNS record or change
- Blocks require a proof of work to authenticate, causing records to be produced at a semi-fixed rate.

Blockchain

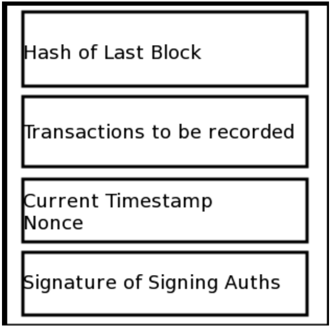

Figure : Contents of an individual block.

Man in the Middle In a DHT

- Need to have a distributed, reliable way to authenticate
- Given: an existing network where nodes have exchanged keys securely
- Given: a new peer who wishes to join the network and share their public key

Prevention

- At least 2 members of the network interrogate the new peer for its public key
- Those interrogators compare their results
- If those results match
  - The new peer creates an authentication record
  - The interrogators sign that record
  - The new record is distributed across the network
- If the results do not match
  - An attack is detected and reported to the new peer by all authenticating servers.
  - A member of the network may make a ban of the compromised peer
  - Otherwise the joining process can be repeated.