Ethereum Name Service

Hello!

Me: Brantly Millegan

Work: True Names LTD (Singaporean non-profit)

Open source project: Ethereum Name Service

Github: https://github.com/ensdomains

Website: https://ens.domains

Lead developer: Nick Johnson

Initial patron: Ethereum Foundation

Outline

- Introducing ENS
- How ENS Works
- DNS-ENS Integration

Introducing **ENS**

Ethereum Name Service

Alternative Internet naming infrastructure with all the security, censorship-resistance, and programmability of the public Ethereum blockchain.

Our **Goals**

Short-term
Complement the DNS tech stack in parallel, expanding the capabilities of the existing DNS namespace

Long-term
Merge with/replace/upgrade the DNS tech stack

Our **Philosophy**

Better infrastructure > competing namespace

- Support ICP-3 and ICANN jurisdiction over namespace
- ENS-native TLD .ETH
  - Experimentation with full benefits (ICP-3.5)
  - Mostly for blockchain projects
- DNS namespace integration
  - .XYZ, .LUXE, .KRED, .ART, .CLUB integration
  - Soon all DNS TLDs

Our **Philosophy**

Naming is basic Internet infrastructure

- Open source, non-profit, community driven
- No ICO, no investors
- Support from the Ethereum Foundation

Use **Cases**

- One name for all of your cryptocurrencies
  - E.g., Send BTC, ETH, DOGE, et al to brantly.xyz
- Usernames for dapps (reverse resolution)
- Decentralized websites: ENS+IPFS
- Naming for Tor .onion addresses
- Text records for personal data or custom record types
- DNS records (.KRED; part of long-term goal)

Growing **Ecosystem**

- +50 cryptocurrency wallets signed-up
- +65 dapps with integration
- Native browser int.: Opera, Status, MetaMask Mobile
- MetaMask ext.: Chrome, Firefox, Edge, Brave, & Tor Browser
- ICANN research project
- Partnership with Protocol Labs (managers of IPFS)
- TruffleSuite integrations
- DNS records with .KRED

ENS **vs. DNS**

- Simpler infrastructure (servers → smart contracts)
- Zero down time, can't DDOS attack
- Decentralized, more censorship resistant
- Built-in cryptography and individual ownership
- Better security
- Programmability
- Interactivity with other Ethereum smart-contracts

How **ENS Works**

What Is a **Blockchain?**

- Shared database that users can change only under specific conditions

- Enforced by cryptography, PoW/PoS, ability to fork, existence of 1000s of copies, et al

- Database exists on nodes that anyone can run

What Is a **Blockchain?**

- Updated in batches of changes/transactions called "blocks" that are cryptographically "chained" together.

- Accounts are public/private key pairs

Bitcoin: Peer-to-peer electronic cash system
Ethereum: Dapp platform

How **ENS Works**

No servers

Runs entirely as a set of smart contracts on the Ethereum blockchain

No permission needed to use the system

How **ENS Works**

To make any change (register name, set records, transfer ownership, et al): you must submit a certain kind of Ethereum tx or set of txs

→ **Requires having Ethereum wallet & costs ETH**

How **ENS Works**

For example:

Registering .ETH name requires 2 txs

 - 1 to initiate process, 1 to finalize

(might be reduced to 1 tx in future with clever trick)

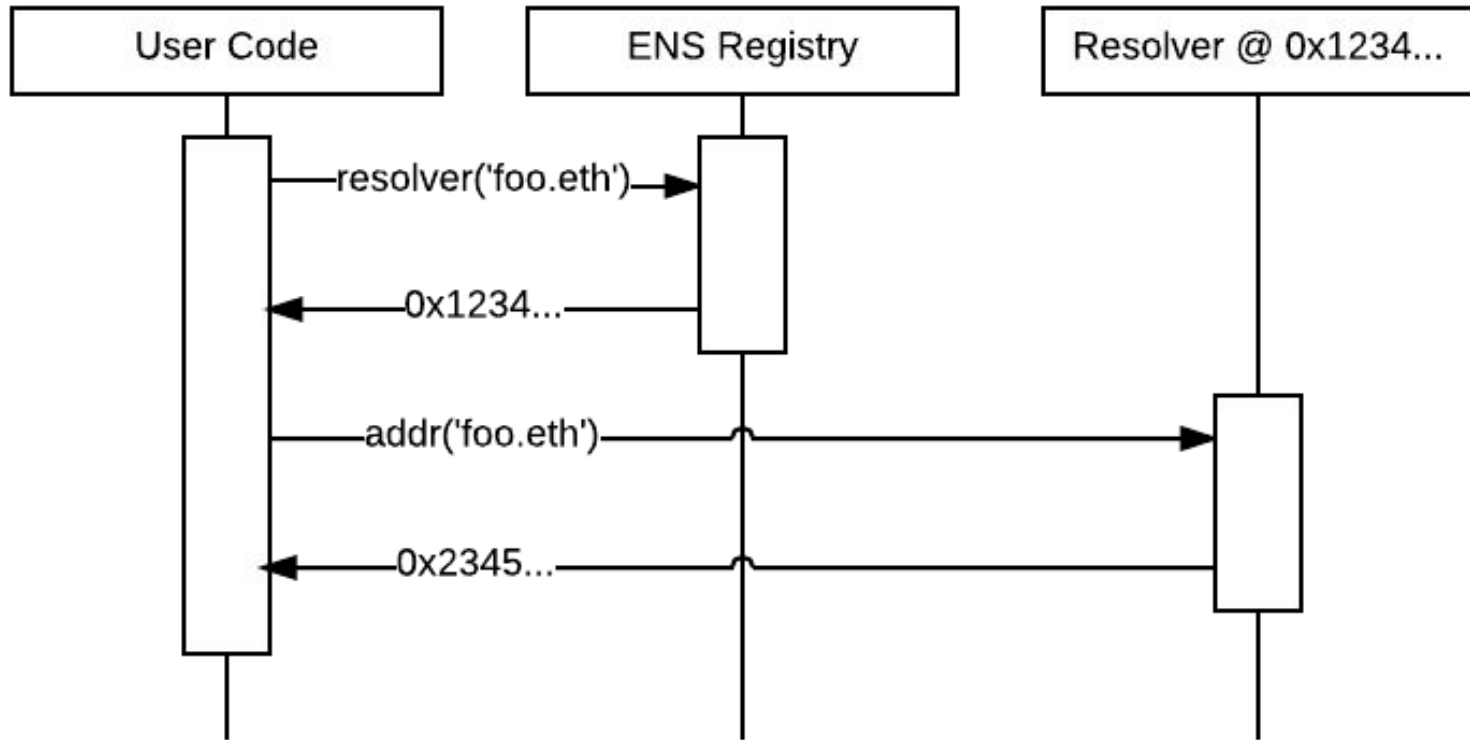→ Having 2 txs stops "**front running**" attack vector

How **ENS Works**

Note:

Lookups (of records, whether name is available, et al) do not cost anything because you're simply looking up information on the public Ethereum blockchain

➜ **Must have some access to Ethereum blockchain** (local full node, lite client, or service, e.g. Infura)

ENS Record **Lookup**

Inside **ENS Records**

"Public Resolver"
- **Address:** Ethereum address
- **Other Addresses**: Other cryptocurrencies
- **Content:** IPFS, Swarm, Tor .onion
- **Text records:** Arbitrary records
- **DNS records** (not available in Manager UI)

You can also make your own custom resolver

DNS-ENS Integration

DNS-ENS **Integration**

- DNS namespace integration

- Storing and serving DNS records with ENS

DNS Namespace Integration

DNS Namespace **Integration**

Goal

People can make ENS records for DNS domains they already own through the normal DNS registration system.

DNS Namespace **Integration**

example.com   → DNS record
                         → ENS record

Not example.eth

E.g. Visit **brantly.xyz** website via DNS, send cryptocurrency via ENS

DNS Namespace **Integration**

Two step process

1) Prove ownership of DNS domain to ENS contract on Ethereum

2) Create and manage ENS record on Ethereum

Proving **Domain Ownership**

DNSSEC oracle on Ethereum w/ DNS Root public key

Can verify any DNS records

Saves proofs already submitted (which makes other proofs cheaper to verify)

Proving **Domain Ownership**

_ens.example.tld

At least one text record field must have this:
a=[insert your Ethereum address]

e.g.
a=0xfB6916095ca1df60bB79Ce92cE3Ea74c37c5d359

Proving **Domain Ownership**

You submit the proof to the DNSSEC oracle on Ethereum as an Ethereum transaction

If proof succeeds, the smart contract automatically grants ownership of domain to Ethereum address provided in text record

Proving **Domain Ownership**

No registration fees to have an ENS record for a DNS domain

- Unlike .ETH names

- Since you've already paid fees

Create **ENS Record**

All of this is done by interacting with the ENS smart-contracts on Ethereum

Our Manager app (app.ens.domains) has an easy-to-use UI.

.XYZ **Integration**

You can do this right now with .XYZ domains

Use .XYZ as your Ethereum Wallet!

● ● ●

XYZ and Ethereum have partnered to allow you to pair your .xyz domain with your Ethereum wallet via the main ENS network.

brantly.xyz on **DNS**

HOME   CONTACT

HELLO!

Thanks for visiting my website. Scroll down 👇 to see what I've been up to.

BLOCKCHAIN

Ethereum Name Service (ENS), Director of Operations (January 2019 – present)

ENS is a decentralized name system built on Ethereum.

.

brantly.xyz on **ENS**

DNS TLD Integration

DNS TLD **Integration**

Just as owners can claim 2LDs, TLD owners can claim them on ENS

DNS and ENS both work hierarchically

Hierarchy of control: TLD owner can control 2LDs, etc

.LUXE **ENS Integration**

.LUXE has this kind of integration today
→ https://nic.luxe/

It's super easy...

Step One
Register a .luxe domain name like helen.luxe or safestore.luxe

Step Two
Associate the domain name to an existing 42-character Ethereum identifier. Ethereum enabled registrars have already integrated this functionality so all you have to do is add your Ethereum public identifier.

helen.luxe BUY → → 0X32A4C9FA3 D46AB6E87F

.LUXE **ENS Integration**

Advantages for TLD owners:

- Can make process easier for customers

- Can manage their namespace as it exists on ENS

- Can use it for a special project - e.g. .ART

What Should **TLD Owners Do?**

DNS **Integration**

Make sure you have DNSSEC enabled for your TLD, and accessible to users

After that, we recommend most TLDs do nothing: **simply allow users to claim their 2LDs on ENS** with DNSSEC (.XYZ approach)

DNS **Integration**

We recommend you only claim your entire TLD's namespace on ENS (.LUXE/.KRED/.ART approach) **if you have a specific objective** you're trying to accomplish with it

Registrars can make claiming an easier process

DNS Records **with ENS**

DNS **Records**

.KRED implementation

ENS for naming logic
Leverage IPFS for storage

EthDNS: CoreDNS plug-in that can retrieve records from ENS and IPFS

DNS **Records**

Benefits
- Better security
- Users can verify records
- Individual management of records
- Less hardware requirements for DNS provider

Useful **Links …Questions?**

- **Website**: ens.domains
- **Manager**: app.ens.domains
- **Docs**: docs.ens.domains
- **Medium**: medium.com/the-ethereum-name-service
- **Twitter**: @ensdomains
- **Email**: brantly@ens.domains