
dnsregistrar Documentation

Makoto Inoue

Dec 05, 2018

Contents:

1	Introduction	1
2	Libraries	3
2.1	DnsRegistrar	3
3	Indices and tables	5

CHAPTER 1

Introduction

In addition to *DNSRegistrar* Truffle based artifact which you can call the smart contract directly, we provide a javascript wrapper which looks up DNS record, extract a proof, submit the proof via DNSSec Oracle, and register to ENS via *DNSRegistrar* using the proof (or delete if the entry does not exist).

Example

```
var DNSRegistrarJs = require('@ensdomains/dnsregistrar');
dnsregistrar = new DNSRegistrarJs(provider, dnsregistraraddress);
dnsregistrar.claim('foo.test').then((claim)=>{
  claim.submit({from:account});
})
```


2.1 DnsRegistrar

DnsRegistrar lookups a domainname for a given DNS type and returns a claim which you use to submit proof into or delete proof from DNSSEC Oracle as well as register the domain to ENS through the DNSRegistrar contract

DNSRegistrar.**claim** (*name*)

returns a claim object which allows you to claim the ownership of a given name on ENS by submitting the proof into DNSSEC oracle as well as claiming the name via the registrar

Arguments

- **name** (*string*) – name of the domain you want to claim

Claim.**getOwner** ()

returns owner ETH address from the DNS record.

Claim.**getOracle** ()

returns Oracle object

Claim.**getResult** ()

returns DnsResult object

Claim.**submit** (*params*)

Either submit or delete proof depending on the state of DNSRecord and DNSSEC Oracle

- it deletes an entry if DNS record returns NSEC/NSEC3(Next Secure)
- it submits proof to DNSSEC Oracle contract if not all proofs are in the contract
- it submits proof to DNSSEC Oracle contract and claims via DNSRegistrar contract if not all proofs are in the DNSSEC Oracle contract

Arguments

- **params** (*Object*) – optional parameter to send to smart contract, such as from, gas, etc

CHAPTER 3

Indices and tables

- `genindex`
- `modindex`
- `search`

C

Claim.getOracle() (Claim method), 3
Claim.getOwner() (Claim method), 3
Claim.getResult() (Claim method), 3
Claim.submit() (Claim method), 3

D

DNSRegistrar.claim() (DNSRegistrar method), 3