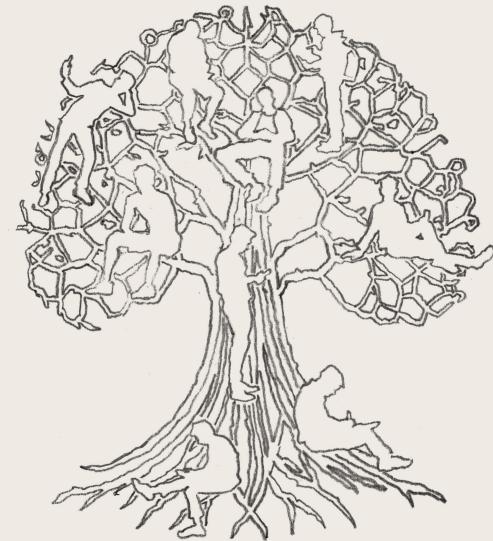
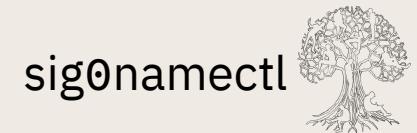


sig0namectl

sig zero name control



About



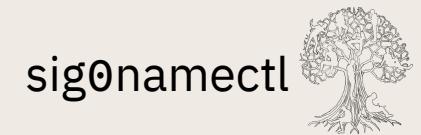
sig0namectl is an empowering technology that allows on-line communities to securely browse, share and publish their own local services & resources.

It achieves this by utilising SIG(0) public keys stored within the Internet's secure Domain Name System (DNSSEC) which allows verification of DNS responses and strong access control to all updates.

This allows participants to request an unused label within a DNS zone and securely publish information resources at or below the requested label.

WHY? To empower people to publish access details to local resources and digital services to their own neighbourhoods and communities.

Social Relevance



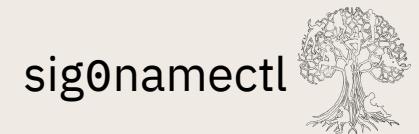
Until now, only Internet Name Registrars, Data Centre Owners, or “cloud resellers” had the ability to update resource information directly into the DNS fabric – offering only limited insecure web portals for their customers to log in and make limited changes indirectly.

With so many service provider security hacks, such as domain hijacking and password leaks, many communities are looking for ways of securely managing their own Internet domain data, free from the risks of relying on third parties doing it on their behalf.

sig0namectl offers more secure, self-sovereign control over Internet domain resources which aligns with emerging EU cyber-security best practices.

Freifunk, the umbrella group for German community networks, is a development partner aligned in the aim to allow members to host services securely within their own existing decentralised community network infrastructure.

Test Group: Freifunk

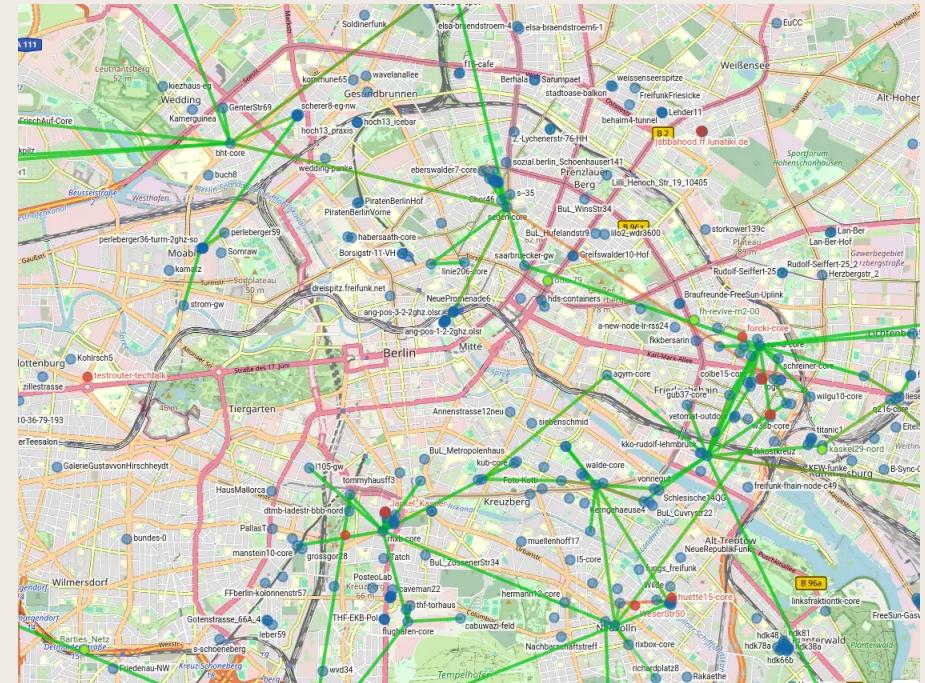


Freifunk is a German non-profit initiative providing free network access and open infrastructure with a mission to democratise communication.

In Berlin, Freifunk's radio network is large, with over 500 nodes and WLAN hotspots spread throughout the city.

sig0namectl empowers Freifunk community users to collaborate, publish & make accessible local web resources:

- on their own computers
- in their own premises
- over their own local networks.



Design Criteria



Autonomy

- Local first, unmediated access
- Local services for local needs

Sustainability

- Minimal resource footprint
- Increased use of existing resources

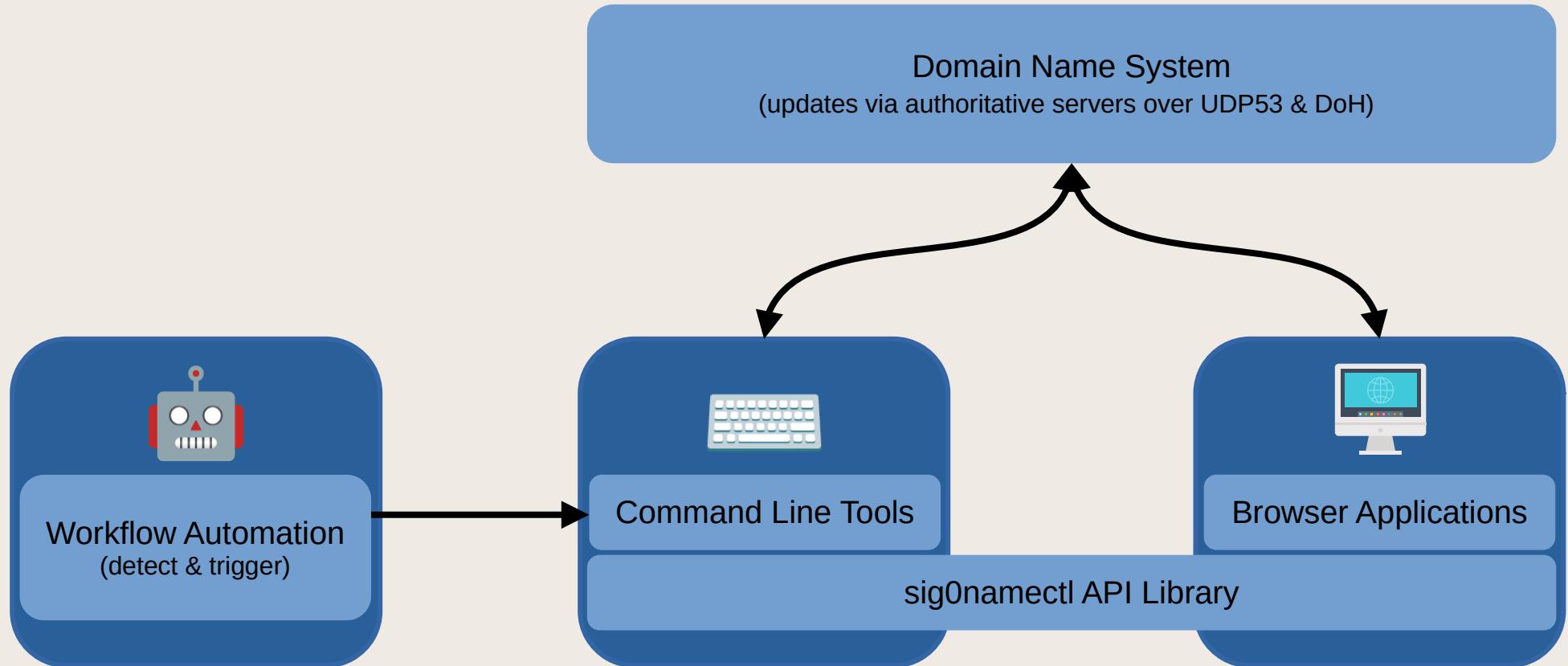
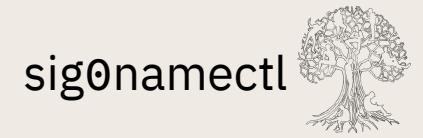
Freedom

- Community key trust anchor with DNSSEC
- Alternatives to ISP & big tech services

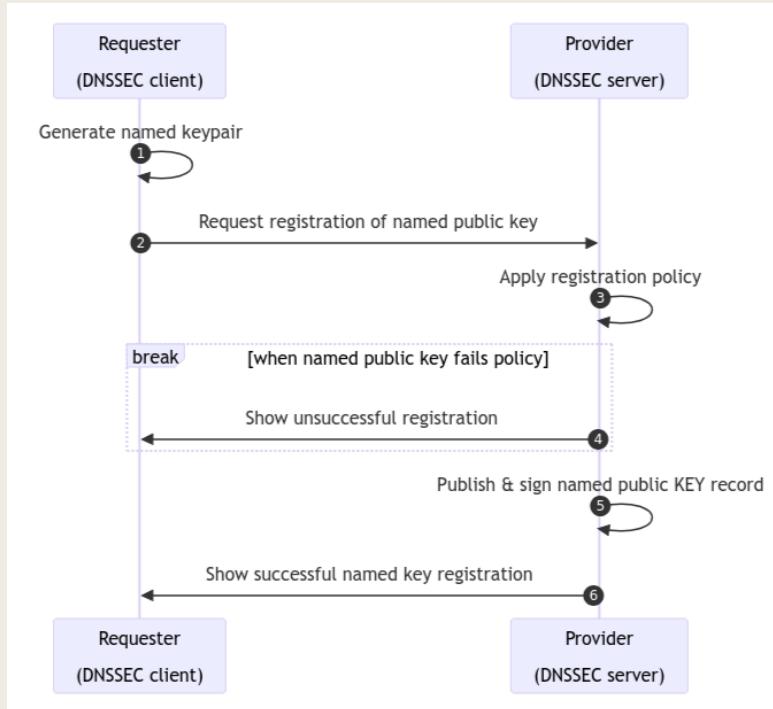
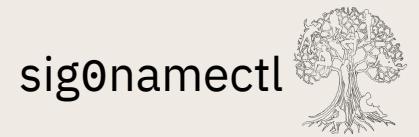
System Requirements

- Low system requirements for resource constrained systems
- Platform independent
- Browser based GUI & command line development environments
- Real-time DNS updates (minimal caching TTLs)
- Decentralised local storage of all credentials
- Modular API library for ease of maintenance

High Level Architecture



DNS Update KEY Requests



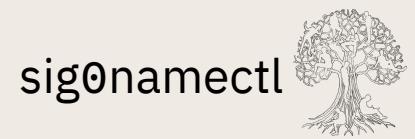
First Come, First Served KEY allocation

Owners of KEY RRs at a zone apex apply a FCFS policy to process further KEY requests. KEY requests for a zone label are provisioned if & only if no label RRsets already exist.

Present mechanism: uses open update _signal zone & `process_requests` workflow tool.

Future mechanism: SRP (think 'DHCP' leases in EDNS0 header)
see <https://datatracker.ietf.org/doc/draft-ietf-dnssd-srp/>

BIND9 config example



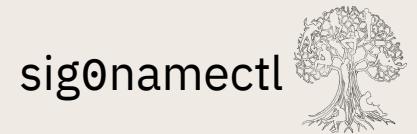
```
# Example BIND9 zone configurations for zenr.io
zone "zenr.io" IN {
    type master;
    file "dynamic/zenr.io/named.zenr.io";
    key-directory "dynamic/zenr.io";
    // auto-dnssec maintain;
    dnssec-policy "default";
    allow-transfer { 138.201.89.108; 2a01:4f8:c17:3dd5::1; };
    update-policy {
        grant "zenr.io" name zenr.io. ANY;
        grant "zenr.io" subdomain zenr.io. ANY;
        grant * selfsub . ANY;
    };
};
zone "_signal.zenr.io" IN {
    type master;
    file "dynamic/_signal.zenr.io/named._signal.zenr.io";
    key-directory "dynamic/_signal.zenr.io";
    check-names warn;
    // auto-dnssec maintain;
    dnssec-policy "default";
    allow-transfer { 138.201.89.108; 2a01:4f8:c17:3dd5::1; };
    allow-update { any; };
};
```

zone apex key configuration

The initial zone apex KEY zenr.io is generated and granted permissions at and below its FQDN.

Subsequent added keys are given similar permissions at and below their label.

Workshop Tech Demo



Checkout the GitHub Repo

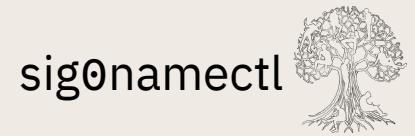
 <https://github.com/NetworkCommons/sig0namectl>

Try Examples & Tutorials for BASH, Golang & WASM JS

 <https://sig0namectl.networkcommons.org/tutorials/>



GUI: Service Discovery



The screenshot shows a web-based interface for managing DNS service discovery. On the left, a sidebar lists 'Domains' such as zembla.zenr.io, dns-sd.org, testaph20.zenr.io, and fehrbuehnenstrasse.bet.freifunk.net. The main area is divided into several sections: 'Browse Domains (PTR Entries)', 'Service Types (PTR Entries)', 'Service Instances (PTR Entries)', and 'Service (SRV Entries)'. Under 'Service (SRV Entries)', there is a table with one entry:

Service	Target	Port	Weight	Priority
imap	zembla.zenr.io	80	0	0

Under 'TXT Entries', there is a single entry: 'txtvers=1,path=/imap.html'.

Service Discovery Inspector

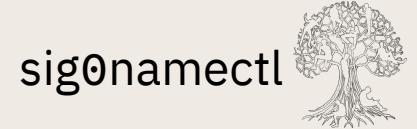
Wide Area DNS-SD uses the same technology computers use to find printers & resources on a local network, but works across the entire Internet.

In expert mode, the SD Inspector web application lists and manages web sites & other network service types & resources published across different DNS domains.

Try live with QR code.



GUI: Domain Manager



Your Domains

Request a new domain:

chorinastrasse request



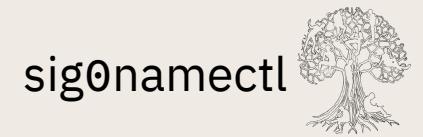
Domain Manager: Request New Domain

The domain manager application handles requests for new domain names.

The default “first come, first served” policy for sig0namectl means that if the full domain name does not exist already, the request will be granted.

Try it live with the QR code.

GUI: Domain Manager



The screenshot shows a web browser window with the following details:

- Header bar: File, Edit, View, Go, Bookmarks, Tools, Settings, Window, Help.
- Address bar: http://localhost:8822/domains.html
- Content area:
 - Your Domains** section with one entry: chorinastrasse.beta.freifunk.net (active).
 - Request a new domain form:
 - Input field: your-new-subdomain
 - Dropdown: zenr.io
 - Button: request

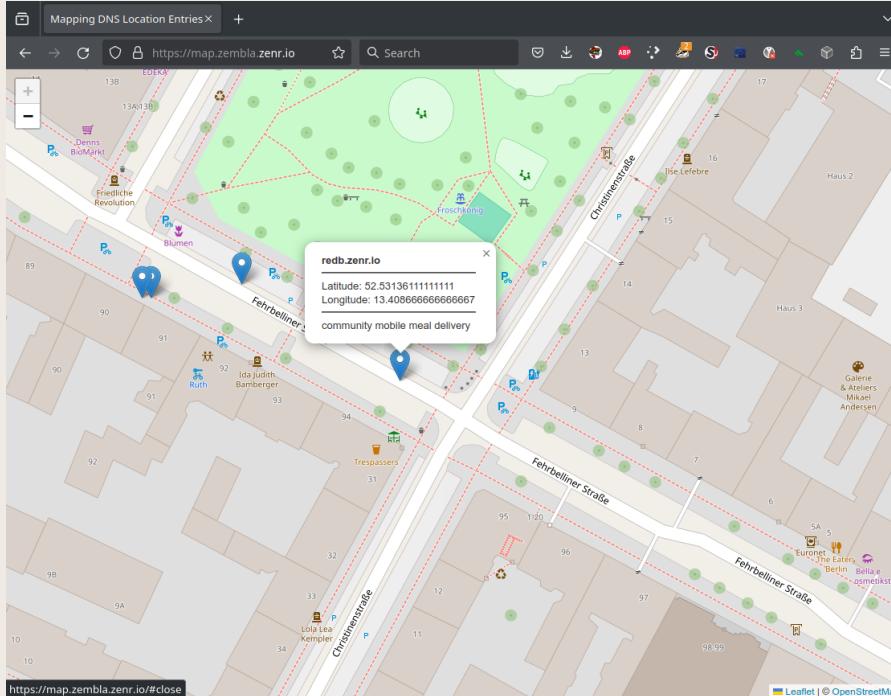
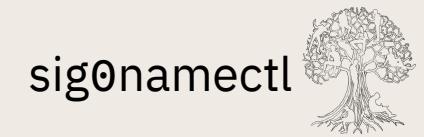


Domain Manager: Manage New Domain

Once successful domain requests are listed as active, DNS update rights for this domain have been granted & DNS resource records can be updated at or below this new domain by selecting it.

Note the domain management application is still under heavy development and is subject to change.

GUI: Resource Maps



Map of zembla.zenr.io's resource locations
with detail of node redb.zenr.io offering local mobile meal delivery

Resource group location map

A map application for local services & resources.

This example shows **zembla.zenr.io**'s resource map where each map point is added to zembla's list. Each resource itself can dynamically update its own location & details in its own distinct DNS name space.

Try live with QR code.



Automation at Work

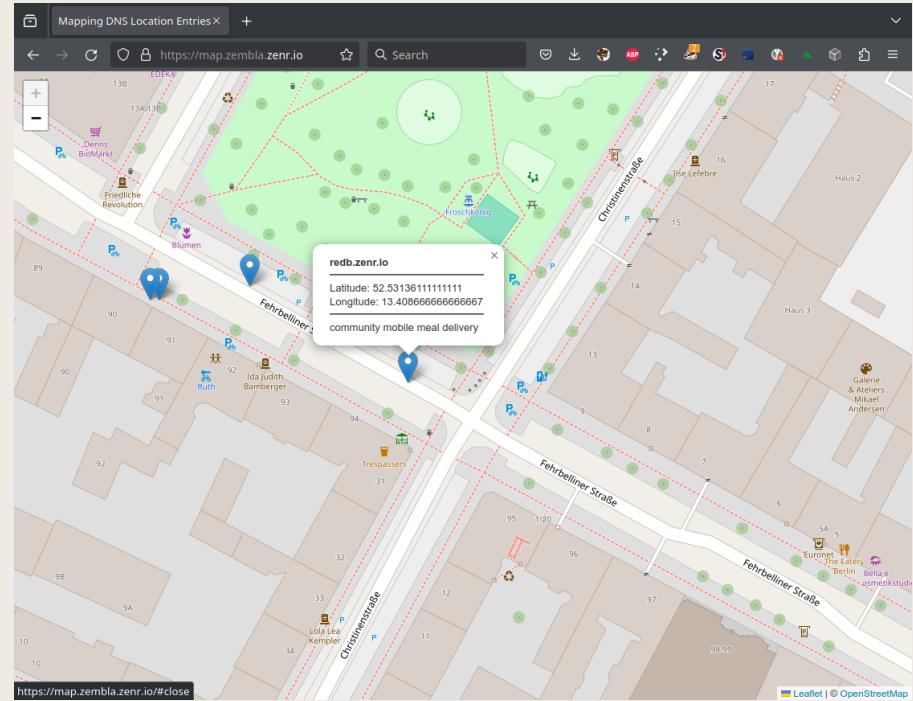
sig0namectl

redb.zenr.io is an Android mobile phone.

It is regularly updating its own position into DNS with information from its GPS hardware by running the provided `sig0namectl start_loc_loop.sh` automation command line script.

See our documentation for further automation possibilities.

Domains	Browse Domains (PTR Entries)	Service Types (PTR Entries)	Service Instances (PTR Entries)	Service (SRV Entries)
zenbla.zenr.io	zenbla.zenr.io	_ftp._tcp.zenbla.zenr.io _loc._udp.zenbla.zenr.io _ssh._tcp.zenbla.zenr.io _http._tcp.zenbla.zenr.io _gopher._tcp.zenbla.zenr.io _telnet._tcp.zenbla.zenr.io	opf6._loc._udp.opf6.zenr.io redb._loc._udp.redb.zenr.io zenbla._loc._udp.zenbla.zenr.io bluebox._loc._udp.bluebox.zenr.io cephal13._loc._udp.cephal13.zenr.io	Target redb.zenr.io Port 80 Weight 0 Priority 0
dns-sd.org	zenbla.zenr.io			TXT Entries
testapb20.zenr.io				community mobile meal delivery
fehrlinierstrasse.bela.freifunk.net				
add domain for inspection				



`redb.zenr.io` autonomously updates its own geolocation details into DNS

Status so far ...



Technical innovations

- Scalable, decentralised & secure dynamic DNS updates using SIG(0)
- WebAssembly API for further Javascript in-browser application development
- Standards-based dynamic DNSSEC and Wide Area DNS-SD compatibility
- Fully open, easy to maintain core library developed in Golang

Infrastructure & deployment

- Dynamic DNSSEC deployed across the Freifunk and Freifunk Berlin domain names
- Service now online under zenr.io and beta.freifunk.net & beta.berlin.freifunk.net

Community Involvement

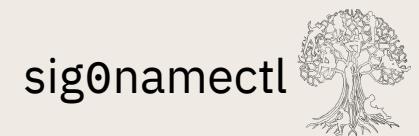
- Freifunk Berlin monthly meetings with presentations & project updates
- Freifunk Radio show appearances with interview & discussions
- Project presentation at international Community Networking event (PicoPeering 2024)

Next Steps

- Browser app development
- Further golang tool capabilities
- OpenWRT router image integration
- DANE, SSHFP, and new WALLET RR
- Wireguard zeroconfig as DNSSD service



Contact



 <https://sig0namectl.networkcommons.org>

 sig0namectl@networkcommons.org

