



84

ANNUAL
GENERAL
MEETING



SAC132: The Domain Name System Runs on Free and Open Source Software (FOSS)

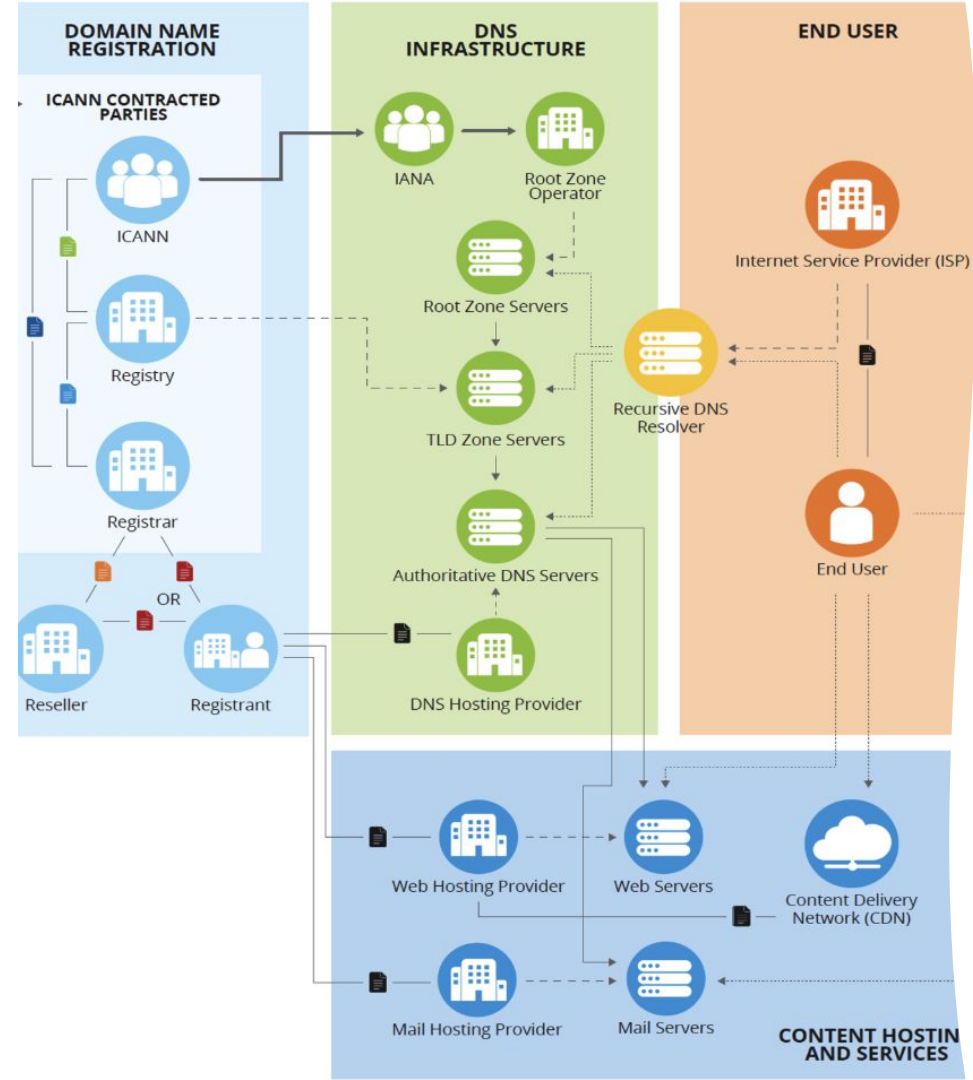
October 2025

Agenda

1. How DNS Works
2. FOSS Prevalence in the DNS ecosystem
3. Understanding FOSS
4. Guidelines for Policymakers & ICANN Community



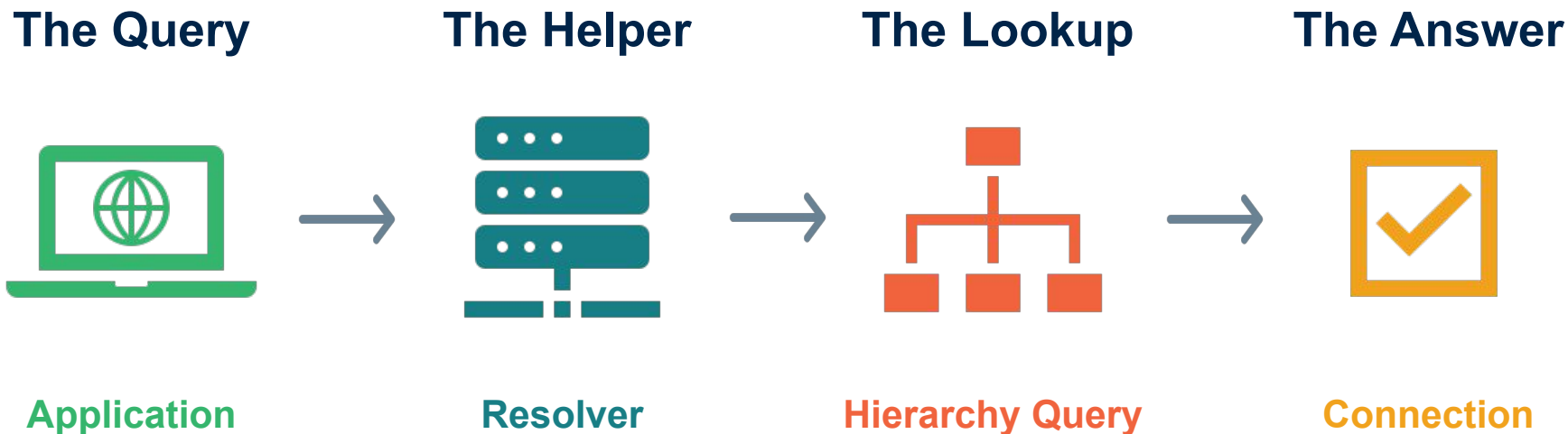
How DNS Works



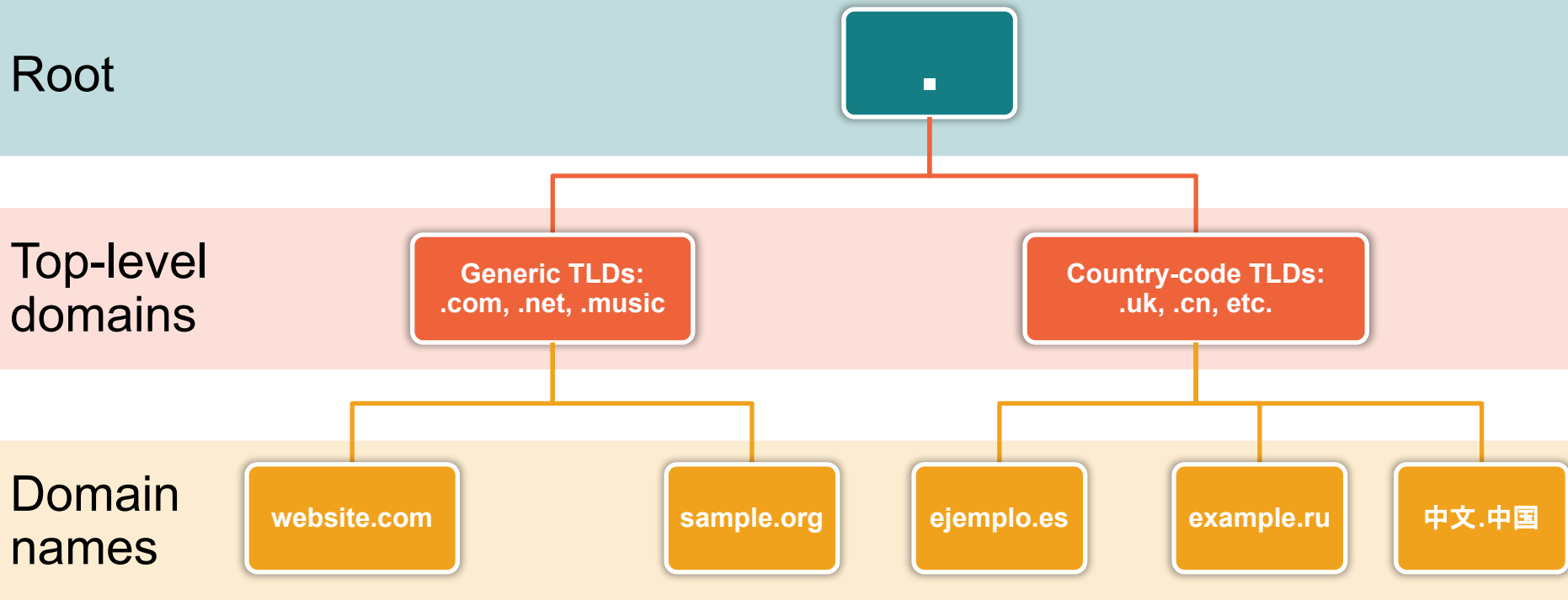
Internet Ecosystem¹

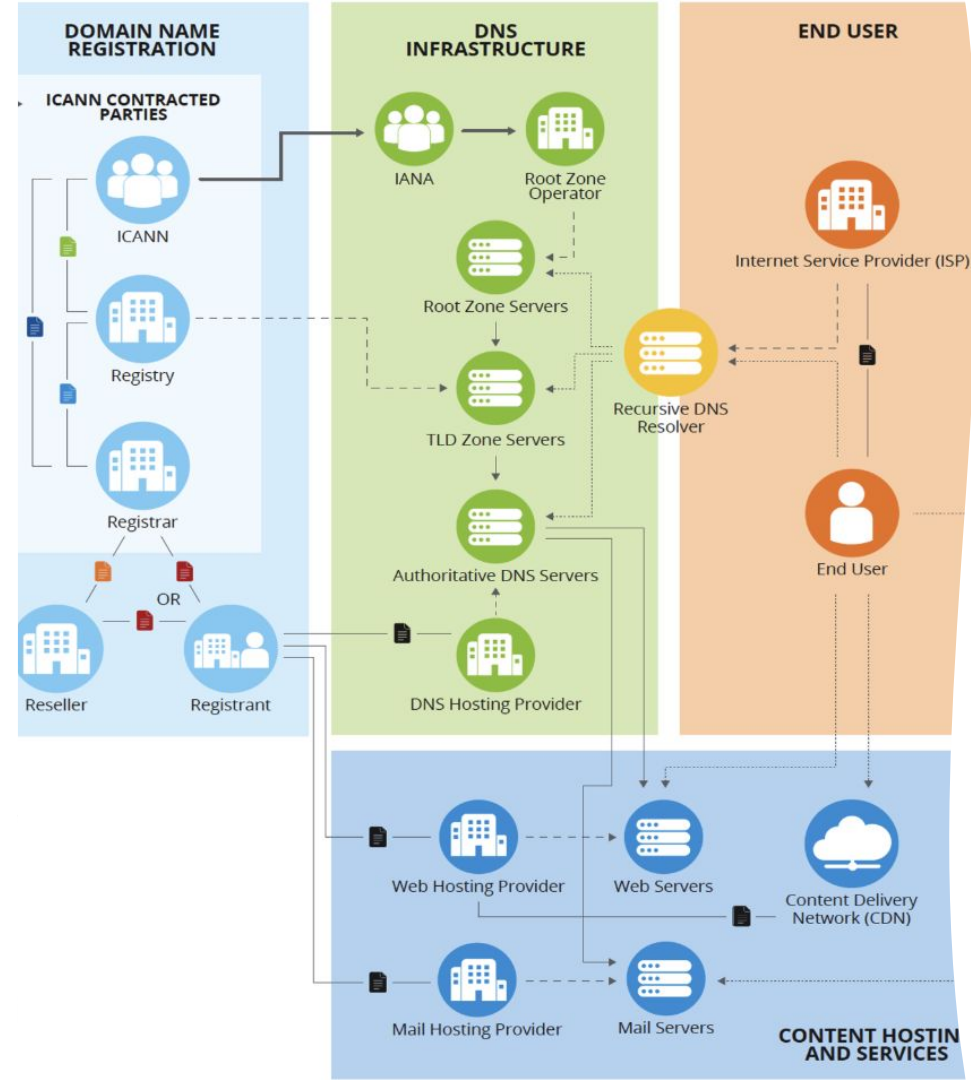
¹ The Internet Ecosystem. Drazek, Keith. "DNS Security: Ongoing Community Work to Mitigate Domain Name System (DNS) Security Threats." Verisign Blog, 7 December 2021.
<https://blog.verisign.com/domain-names/ongoing-community-work-to-mitigate-domain-namesystem-security-threats/>

The DNS Query Flow (Simplified)



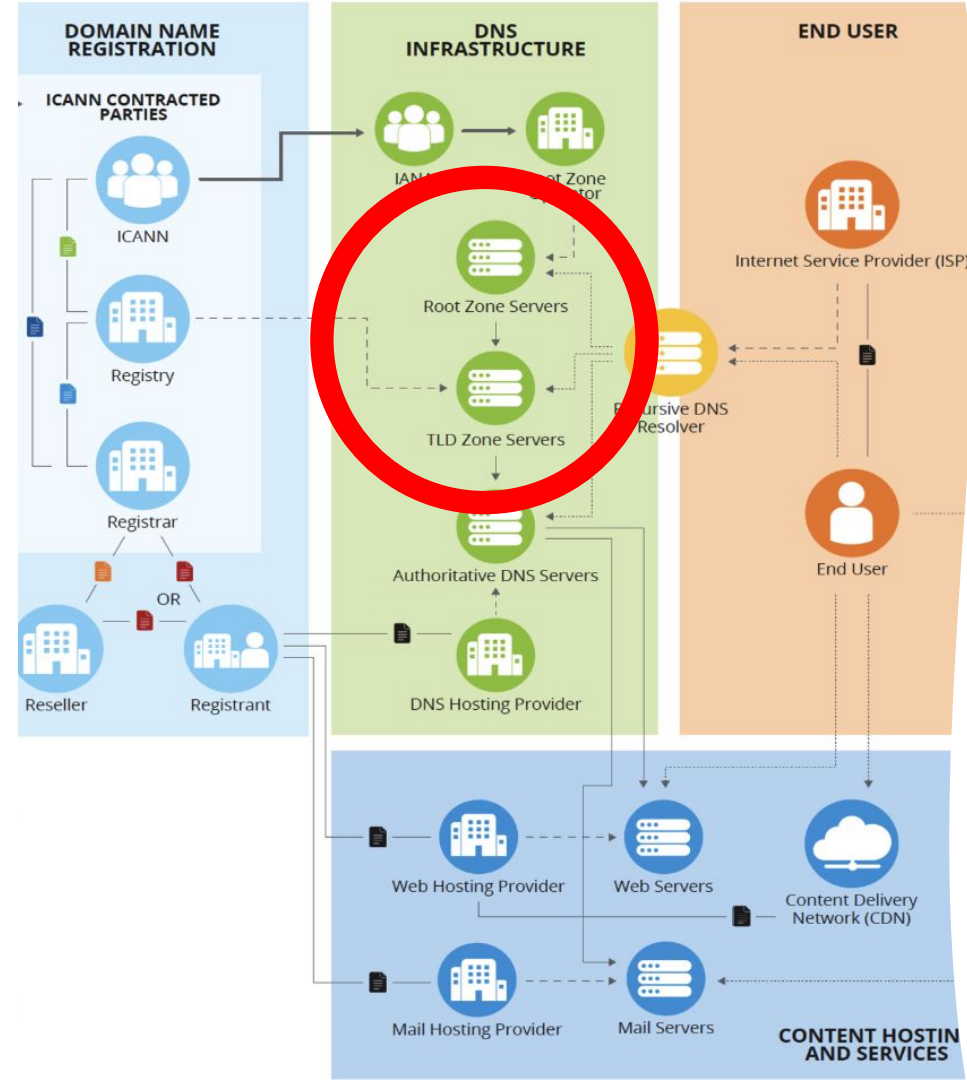
The DNS Namespace is Hierarchical





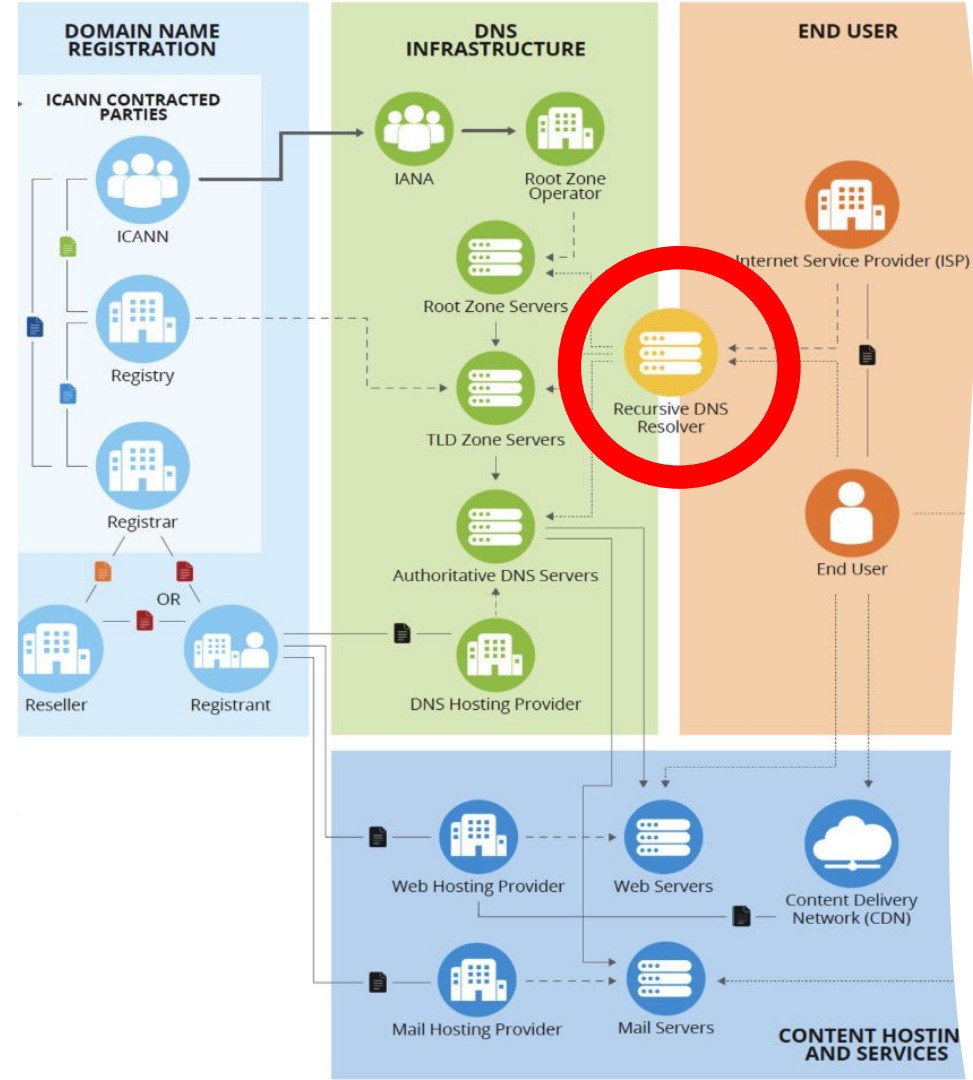
Software powers everything

¹ *The Internet Ecosystem*. Drazek, Keith. "DNS Security: Ongoing Community Work to Mitigate Domain Name System (DNS) Security Threats." Verisign Blog, 7 December 2021.
<https://blog.verisign.com/domain-names/ongoing-community-work-to-mitigate-domain-namesystem-security-threats/>



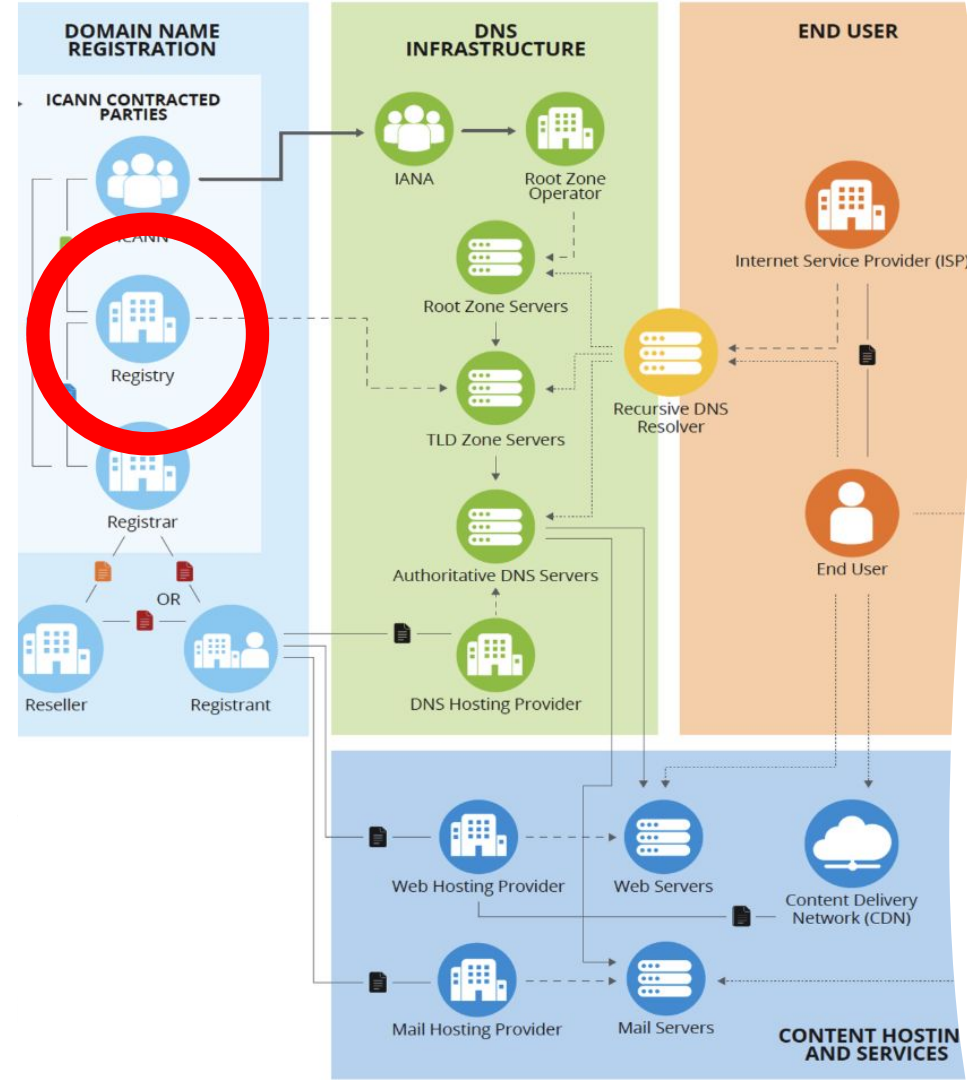
Software powers everything

¹ The Internet Ecosystem. Drazek, Keith. "DNS Security: Ongoing Community Work to Mitigate Domain Name System (DNS) Security Threats." Verisign Blog, 7 December 2021.
<https://blog.verisign.com/domain-names/ongoing-community-work-to-mitigate-domain-namesystem-security-threats/>



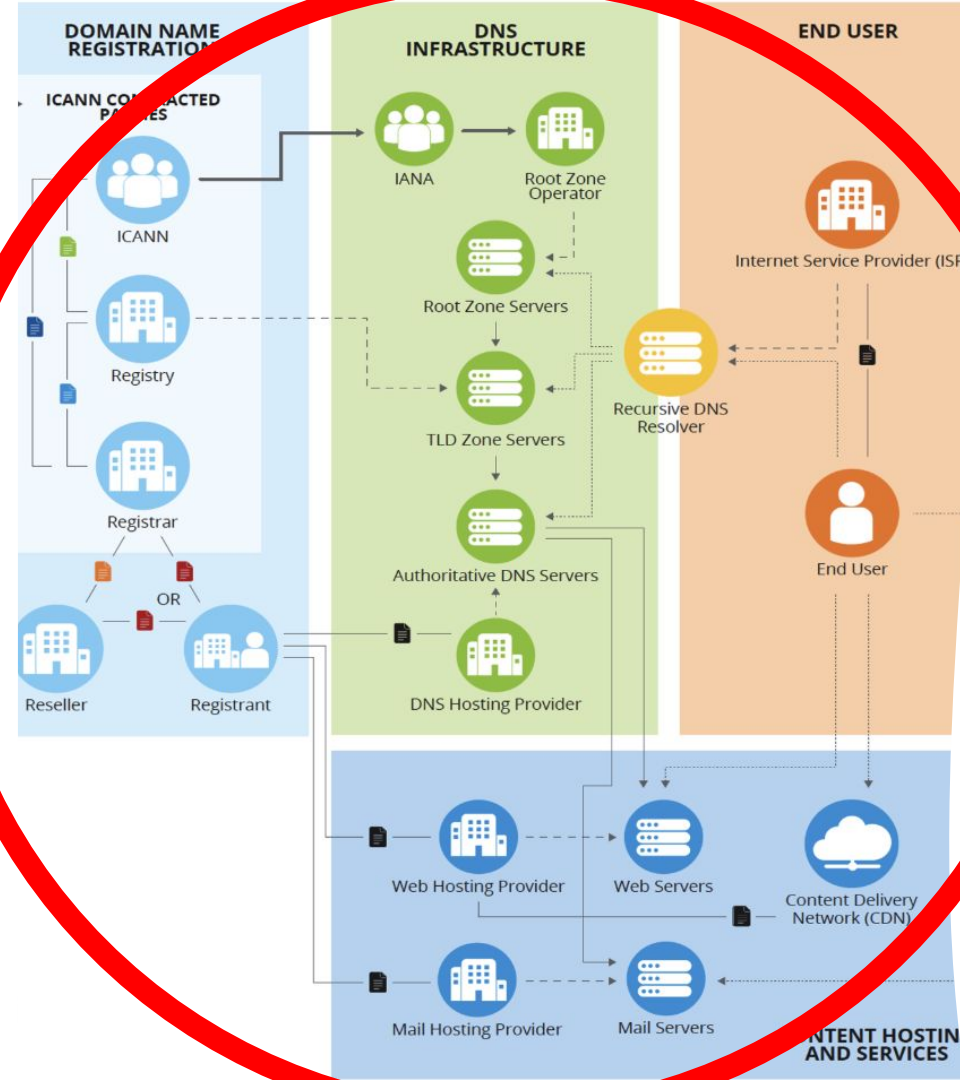
Software powers everything

¹ *The Internet Ecosystem*. Drazek, Keith. "DNS Security: Ongoing Community Work to Mitigate Domain Name System (DNS) Security Threats." Verisign Blog, 7 December 2021.
<https://blog.verisign.com/domain-names/ongoing-community-work-to-mitigate-domain-names-system-security-threats/>



Software powers everything

¹ The Internet Ecosystem. Drazek, Keith. "DNS Security: Ongoing Community Work to Mitigate Domain Name System (DNS) Security Threats." Verisign Blog, 7 December 2021.
<https://blog.verisign.com/domain-names/ongoing-community-work-to-mitigate-domain-namesystem-security-threats/>



Software powers everything

¹ The Internet Ecosystem. Drazek, Keith. "DNS Security: Ongoing Community Work to Mitigate Domain Name System (DNS) Security Threats." Verisign Blog, 7 December 2021.
<https://blog.verisign.com/domain-names/ongoing-community-work-to-mitigate-domain-namesystem-security-threats/>



FOSS Prevalence in the DNS ecosystem

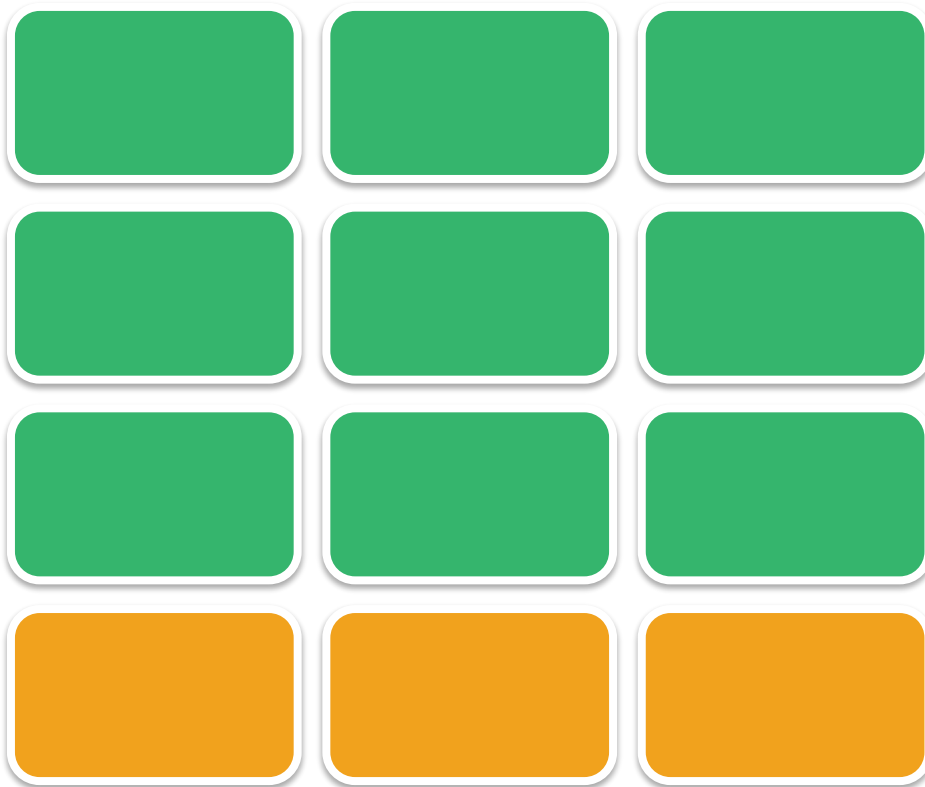


The DNS is built and sustained on Free and Open Source Software.

This is not a niche practice, but the *dominant reality*.



FOSS in the Root Server System



**9 / 12 Root Server
Operators use
FOSS exclusively**

TLD Operators

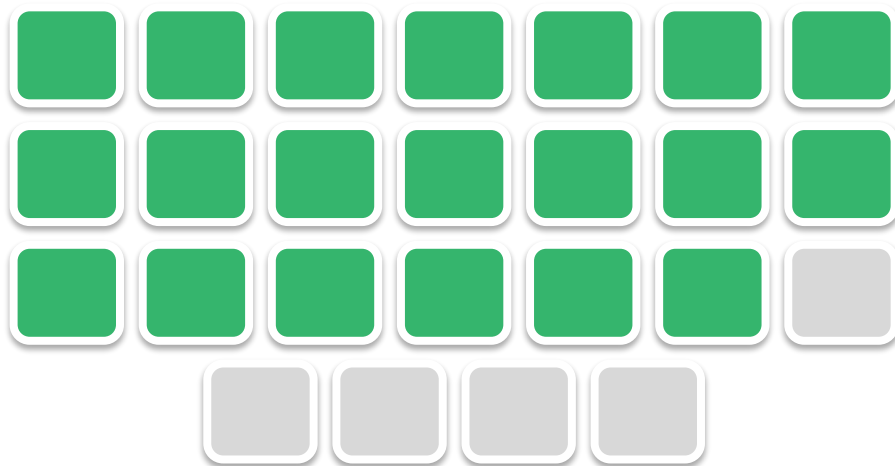
Top TLD Providers

9 / 10



Top ccTLD Providers

20 / 25



FOSS in Registry Systems

Model 1 Entirely FOSS

Viable, complete, end-to-end registry platforms.

Examples:

- FRED
- NomulusX

Model 2 Proprietary on FOSS

Custom, proprietary logic built on a FOSS foundation.

Examples:

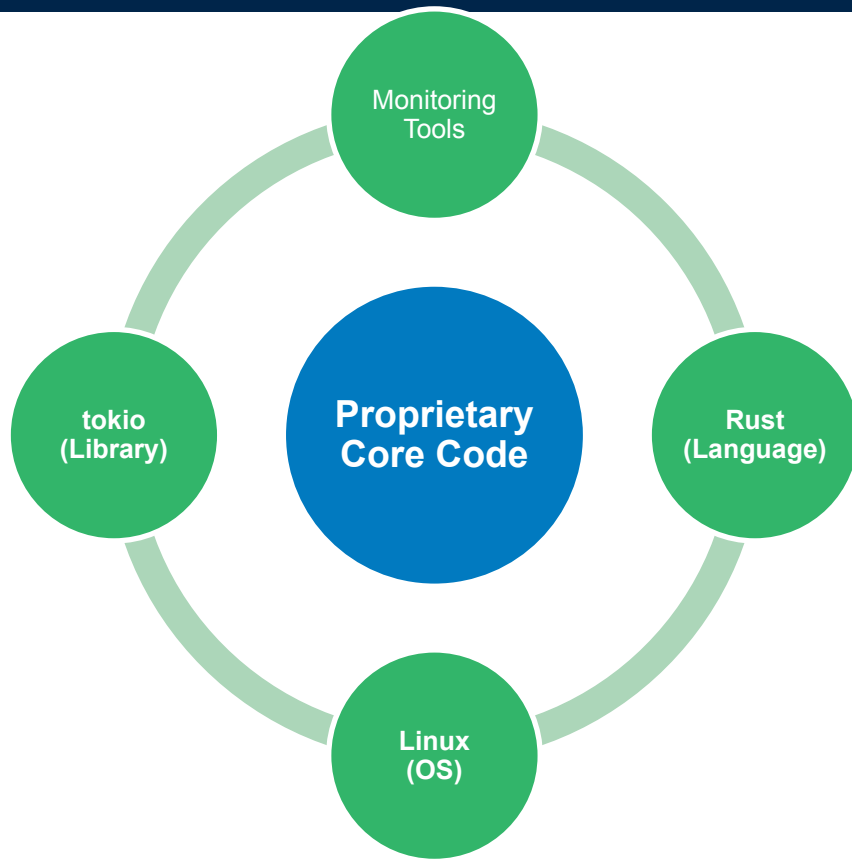
- PostgreSQL
- nginx

10 out of 10 major registry platforms we surveyed incorporate FOSS components.

Case Study: Cloudflare 1.1.1.1

“Cloudflare's proprietary software behind 1.1.1.1 operates at scale thanks to FOSS, not in spite of it.”

- SAC132



Resolver Infrastructure

FOSS has a ***substantial*** presence across the entire resolver ecosystem, from local networks to global cloud platforms.

Local ISP/Enterprise

- ~80% of users worldwide
- Predominantly FOSS

Cloud Platforms

- At least 4 major hyperscalers rely on FOSS

Public Resolvers

- Quad9 (9.9.9.9)
- DNS4EU
- Cloudflare 1.1.1.1
 - Proprietary core, surrounded by FOSS

FOSS is the *foundation* of critical DNS infrastructure

≥9/12

Root operators use FOSS exclusively

10/10

Major registries use FOSS components

9/10

Top TLD providers use FOSS

Substantial

FOSS presence across resolver ecosystem

Understanding FOSS

FOSS: More Than "Free" Software

THE FOUR FREEDOMS



USE

The software for
any purpose.



STUDY

How the software
works.



SHARE

And redistribute
copies.



CHANGE

And release
improvements.

Traditional Software Model



Vendor □ Customer relationship

Driven by purchase contracts

Includes support agreements & SLAs

Clear legal entity responsible

Policy lever: Contractual obligations

FOSS Model



Maintainer □ Operator relationship

License grants freedoms, NO contract

Typically "AS-IS, NO WARRANTY"

Often no single legal entity

Policy lever: ????

Key Roles in the FOSS Ecosystem



Maintainer

Stewards the project, reviews contributions, and ensures quality.



Contributor

Submits improvements, bug reports, and fixes. Can be anyone, anywhere.



Operator / User

Deploys the software, runs infrastructure, and is responsible for updates.

Inherent Risks of FOSS in General



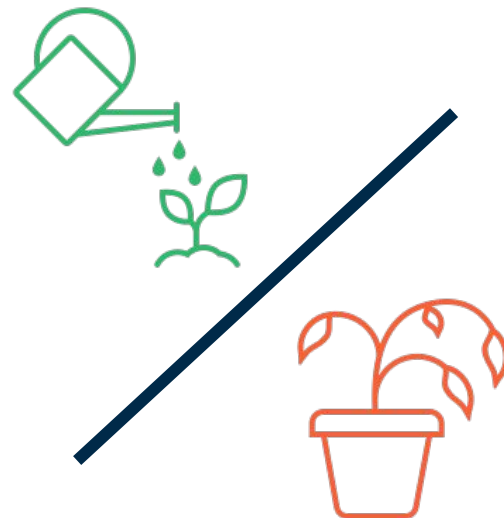
Maintainer Burnout

- Most FOSS relies on single, unpaid volunteers.
There is a corresponding risk of burnout and project abandonment.



No Warranty, No Guaranteed Support [by default]

- No entity required to fix problems
- No default Service-Level Agreements
- Operators must take responsibility



Financial Fragility

- Funding decoupled from Use ("Free-rider")
- Small initiatives/orgs are vulnerable to funding shocks resulting from new regulatory burdens.

Strengths of FOSS in DNS



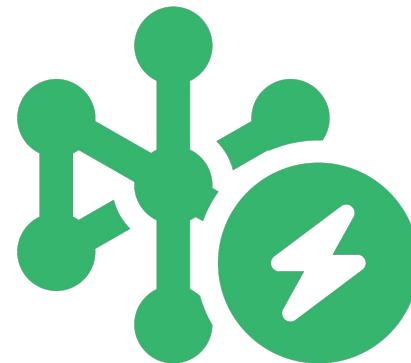
Transparency & Collaborative Security

- Academic & operator community maintain a strong, active culture of openly scrutinizing FOSS DNS software.
- Flaws are "openly discussed and fixed at top speed"
- Operators appreciate the ability to diagnose, verify and "expedite patching" of vulnerabilities



Stability of Core DNS Projects

- Most popular DNS projects are supported by long-lived, stable organizations that have been maintaining the software for 20+ years.



Operational Resilience Through Diversity

- Multiple implementations enable diverse software stacks, avoiding single points of failure and preventing vendor lock-in.

Security - Not Better or Worse, but Different

Understanding what truly determines software security

What DOESN'T Determine Security

- ✗ Open vs. closed source code
- ✗ Paid vs. volunteer maintenance
- ✗ Commercial vs. non-profit

What DOES Determine Security

- ✓ Quality of development process
- ✓ Code review rigor
- ✓ Testing practices
- ✓ Vulnerability disclosure process
- ✓ Patching speed
- ✓ Maintenance sustainability

Why Traditional Regulation Backfires on FOSS

Traditional Regulation



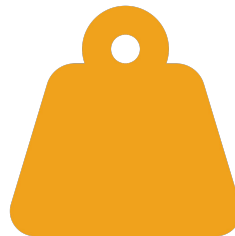
Assumes a vendor-customer model with **financial transactions** and **contracts** at every step.

The FOSS Reality



- No guaranteed vendor, contract, or payment.
- Software is often maintained by **volunteers**.

The Unintended Burden



Compliance costs and **legal risks** are imposed on maintainers who have no ability to absorb them.

The Backfire



- Maintainers **abandon projects**, switch to proprietary licenses, or avoid the region.
- The software we depend on becomes **less secure** and **less available**.

What's at Stake



If We Get This Wrong

Infrastructure Instability

Innovation Suppression

Digital Autonomy Erosion



If We Get This Right

**Strong, Sustainable FOSS
Ecosystem**

Secure, Stable DNS Infrastructure

Continued Innovation & Diversity

Accessible Infrastructure for All

Guidelines for Policymakers & ICANN Community

Contemporary Approaches to FOSS Regulation

Key strategies from recent global regulatory frameworks



Allocate Responsibility Appropriately

2023 US CYBERSECURITY STRATEGY · 2025 UK CODE

Put obligations on commercial integrators & deployers, not maintainers.



Incentivize Sustainable Maintenance

EU CYBER RESILIENCE ACT

Introduce optional 'steward' role to incentivize community support and maintenance for critical project.



Adapt Supply Chain Concepts

EU NIS 2 IMPLEMENTING ACT

Recognize that no contracts means no direct supplier relationship, adapt regulatory strategy accordingly.



Avoid Conflicting Regional Regimes

EU NIS 2 DIRECTIVE

Prevent regulatory overlaps for globally critical infrastructure like root servers

Five Actionable Guidelines for Policymakers

1 Acknowledge the Critical Role of FOSS

2 Consult the FOSS Community

3 Make Use of Contemporary Cases

4 Incentivize FOSS Sustainability

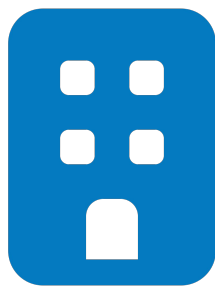
5 Address Systemic Risks Collectively

What ICANN Community Members Can Do



GAC Members

- ✓ Share
- ✓ Advocate
- ✓ Reference



Rys / Rrs

- ✓ Support
- ✓ Establish
- ✓ Share



Technical

- ✓ Document
- ✓ Contribute
- ✓ Educate



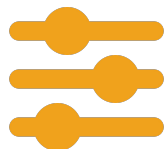
All

- ✓ Advocate
- ✓ Bridge the gap

Key Takeaways from SAC132



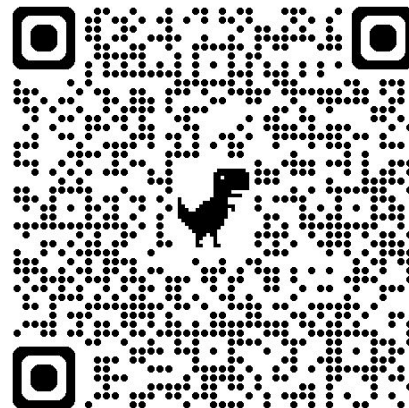
The DNS Runs on FOSS



FOSS is Different
↳ *Policy Must Adapt*



We Can Get It Right
↳ *Good Examples Exist*





Backup Slides

Backup Slide 1 - Research Methodology

- Authoritative Server Analysis
 - Fetched root zone from IANA
 - Mapped TLDs to nameserver IP addresses
 - Used IP-to-ASN mapping to identify operators
 - Calculated market share by number of TLDs served
 - Manual verification of top 25 operators' software use
 - Separate analysis for ccTLDs including IDN variants
- Limitations
 - No central registry of all DNS software deployments
 - Reliant on public statements and direct knowledge for some operators
 - Resolver ecosystem data based on market share analysis, not exhaustive survey
- Confidence Level
 - Very high confidence for root and major TLDs (direct data)
 - High confidence for resolver prevalence (market analysis)
 - See Appendix B of SAC132 for complete methodology

Backup Slide 2 - FOSS License Examples

- Mozilla Public License 2.0 (MPL 2.0)
 - Used by: BIND9
 - Key feature: File-level copyleft - modifications to MPL files must be shared, but can combine with proprietary code
- BSD 3-Clause License
 - Used by: NSD, Unbound
 - Key feature: Very permissive, minimal restrictions, allows proprietary derivatives
- GNU General Public License v2/v3 (GPL)
 - Used by: PowerDNS, Knot, Dnsmasq
 - Key feature: Strong copyleft - derivatives must also be GPL
- Apache License 2.0
 - Used by: CoreDNS, Nomulus
 - Key feature: Permissive, includes patent grant

Backup Slide 3 - DNS-OARC Survey Results

- 98 Responses
- 93% Use Open Source
- 77% Aware of Regulations
- Top Concerns About Software Regulation
 - 72% worry some FOSS projects may be abandoned
 - 66% expect compliance to increase software costs
 - 49% concerned projects may switch to proprietary licensing
 - 29% worried about inhibiting their own FOSS publishing
- Operator perspective: "Open source QUICKLY exposes any flaws... getting patched far faster than any commercial entity." - Survey respondent