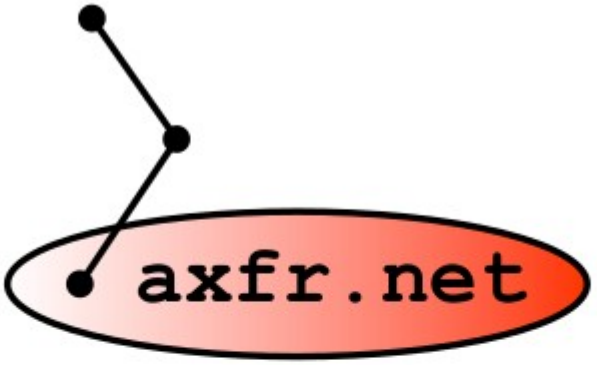
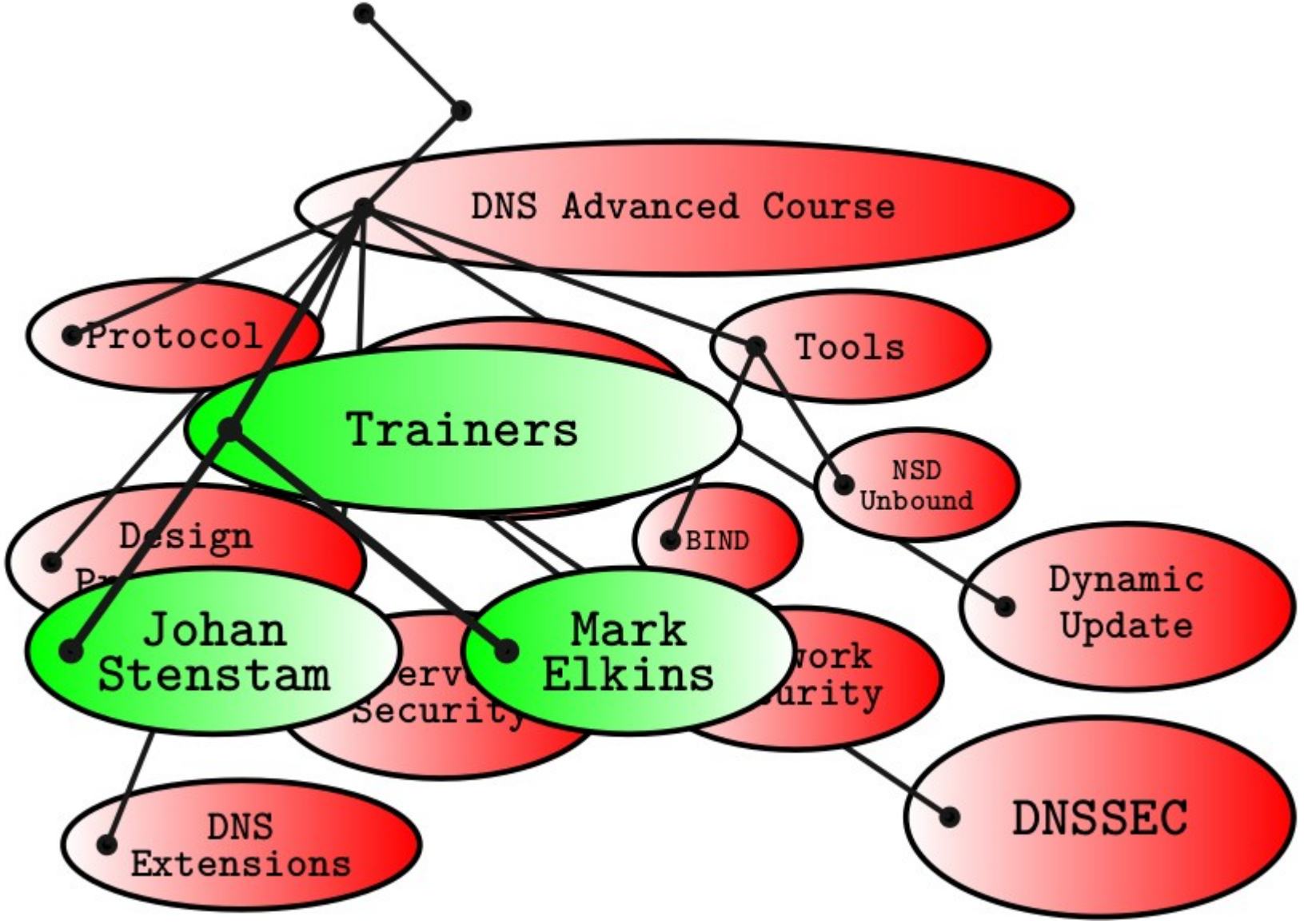
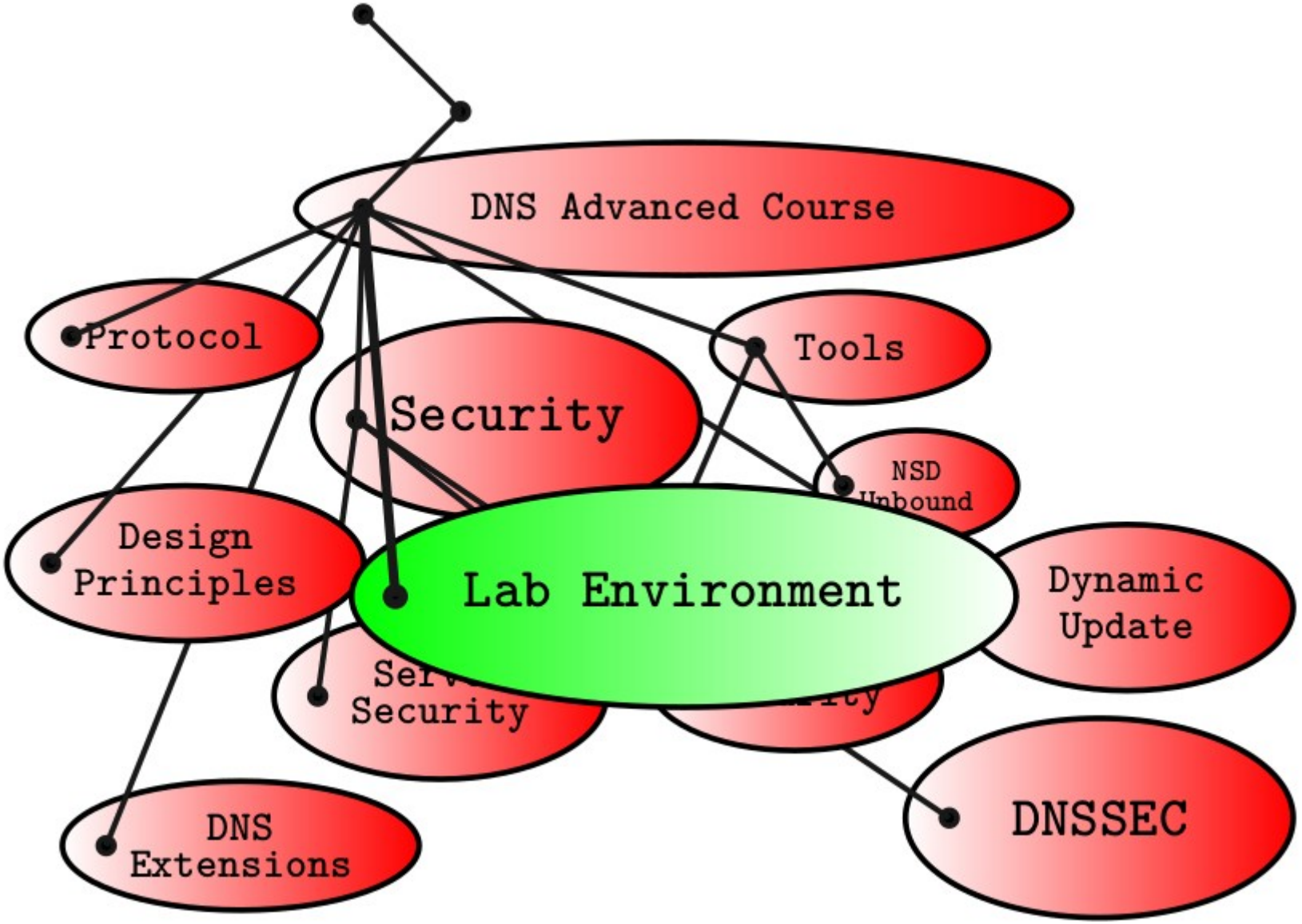


.zadna

DNS Advanced Course


DNS Advanced Course
Protocol
Tools
Trainers
NSD
Unbound
Design
BIND
Dynamic
Update
Johan
Stenstam
Mark
Elkins
Security
DNS
Extensions
DNSSEC

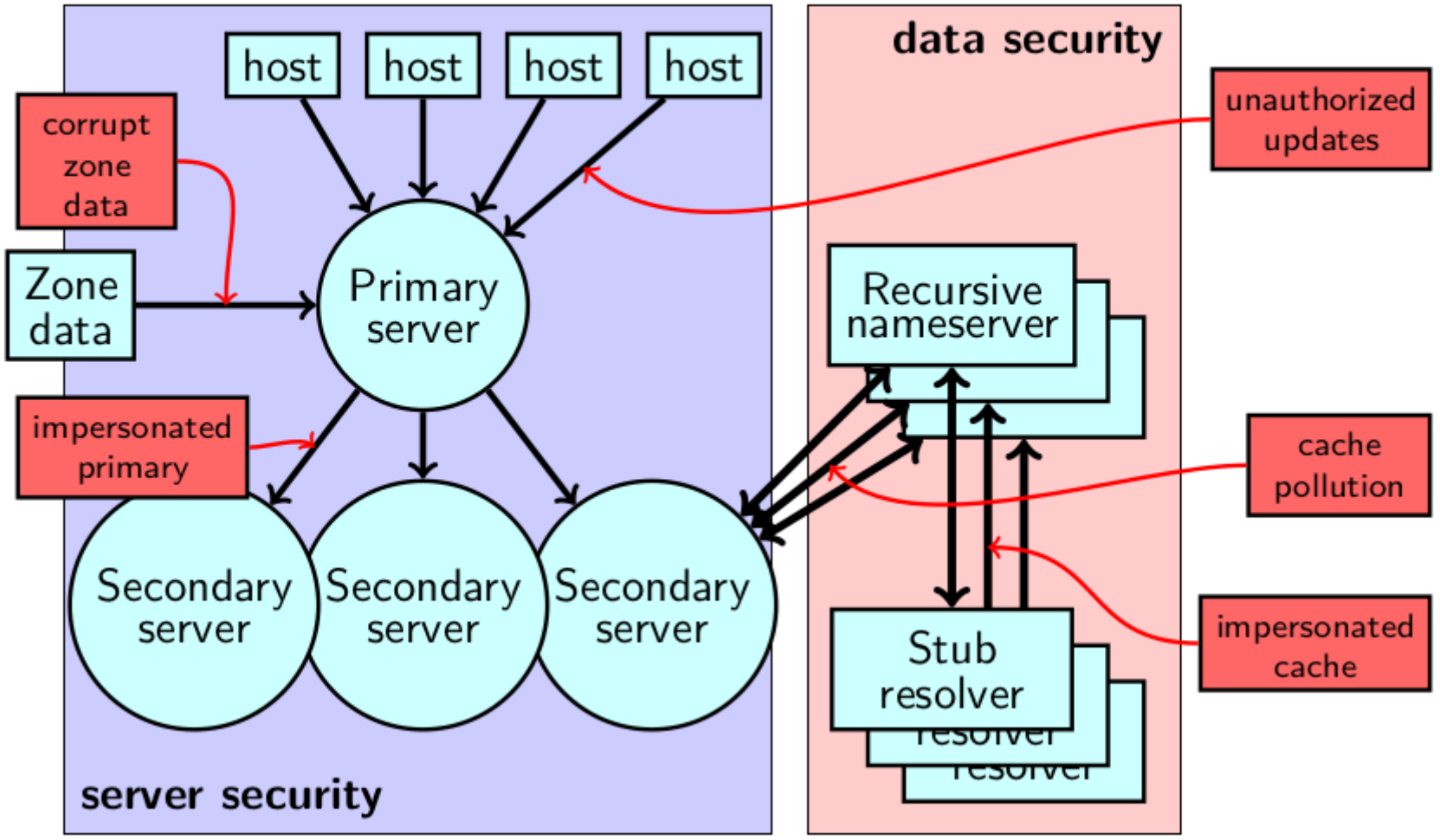

DNS Advanced Course
Protocol
Security
Tools
NSD
Design
Principles
Lab Environment
Dynamic
Update
Security
DNS
Extensions
DNSSEC

KASP: Key And Signing Policy

The OpenDNSSEC project pioneered the concept of a policy for how the owner of a zone wants to keep the zone DNSSEC signed.

- Prior to KASP it was not clearly understood that different zones need different policies and that separating the policy definition from the zone provides a useful abstraction.
- "KASP" is now the standard way of describing **how** a zone should be signed.
- Finally, with version 9.16, also BIND9 has moved to a clear adoption of the KASP model.

DNS Vulnerabilities



The Trust Anchor for Root: BIND9

- BIND9 uses the option "**dnssec-validation**" to determine DNSSEC "semantics":

```
options {
    dnssec-validation yes; # needed by validating
                           # resolvers
};
```

- **dnssec-validation** default to "**yes**" but may be set to "**no**" to explicitly turn off validation functionality
- BIND 9.8+ has an added feature to simplify configuration of DNSSEC validation. Using `dnssec-validation auto;` causes BIND9 to use a compiled in trust anchor for root for validation

Key Algorithm: RFC 8623 Recommendations

- RSA: Slow signing, quick validation
  - **Must** be supported in DNSSEC
  - There are several "secure hash" algorithms for RSA keys but only SHA256, i.e. algorithm 'RSA/SHA256" is relevant today (among other things it is used for the root zone)
- ECDSA (Elliptic Curve): Much smaller signatures and faster signing (but slower validation)
  - The curve ECDSAP256SHA256 **must** be supported, for both signing and validation.
- ED25519 and ED448: Edwards-curve Elliptic Curve signature system (EdDSA). Another set of elliptic curve algorithms. Also small signatures and fast signing.
  - ED25519 is **recommended** and expected to become the future default.

New Record Types: `CDS` ("Child DS")

Earlier we touched upon the question of fully automated `DS` rollovers.

The DNSSEC community is now (2023) migrating towards a fully automated scheme for KSK rollovers. It is not in common use yet, but a few TLDs (including `.SE` and `.NU`) use it in production.

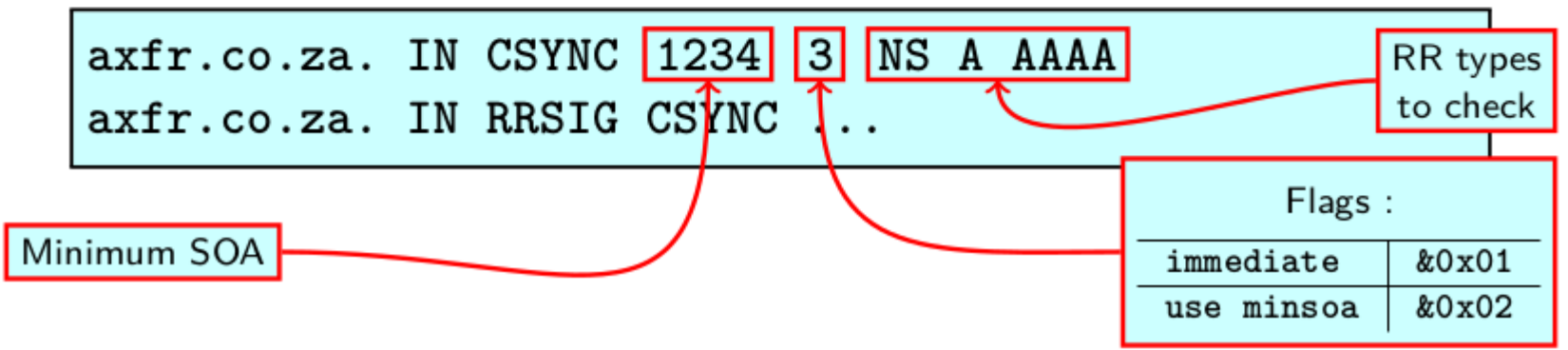
- By publishing a `CDS` record the child enables the parent to notice that it is about to roll the KSK.
- When (or if) the parent detects the presence of a `CDS` record for another KSK than the present `DS` in the parent zone, it may initiate a replacement of the `DS` with the data from the `CDS`

`CDS` is described in RFC 7344 and RFC 8078.

New Record Types: CSYNC

The CSYNC record is published in the child zone and when (or if) the parent notices its presence it may be used to synchronize the delegation information between child and parent.

- contains a minimum SOA serial for which the CSYNC is valid
- contains a flags field, with two defined flags: "immediate" and "use the soamin field"
- contains a list of RR types that should be synchronized



The Answer Is In The Bottle

Drinking For Admins

How DNSSEC Really Works

VODKA PURE PRESS

Brandy G. Tonic

“DoT” (DNS over TLS, DTLS)

To secure the “last mile” (i.e. the transport between stub resolver and recursive server) it is possible to encrypt the traffic and the primary alternative for encryption is to use TLS.

- In theory it should be possible to do TLS either over TCP transport or over UDP (DTLS), but in practice only TCP transport is working today.

Note also that support for DNS over TLS is still lacking in the usual stub resolver implementations.

- Therefore the easiest (and in many cases only) way of utilising this is by using Unbound as a local forwarder
- I.e. unsecure transport from application to local Unbound on the same host, and then TLS secured transport to a remote recursive nameserver

“DoH” (DNS over HTTPS)

Another development is the emergence of DoH which is most typically exemplified by the web browsers from Mozilla (i.e. Firefox and similar) changing

- from using the local DNS resolver (as configured in the resolver configuration, typically via DHCP)
- to instead use a Mozilla configured resolver (typically Cloudflare), accessed over HTTPS.

This has all sorts of both positive and negative consequences and has to a large extent taken the Internet community by surprise.

johani@axfr.net

mje@posix.co.za