Multi-vendor DNS Cookies

Willem Toorop & Ondřej Surý

IETF 102, Montreal

DNS Cookies 101

- Handshake between Client and Server to get the Cookie
- No Cookie? No Large Answers!
- Cookie? Large Answers!
- Cookie? RRL Disabled!

Why DNS Cookies?

- DNS Native Protection Mechanism against Amplification Attacks
    - No operator asked for this, it’s DNS vendor initiative.
    - Protection in the DNS itself, no traffic engineering needed.
- To be helpful, it needs to be enabled everywhere.
- Multi-vendor cooperation desirable.

Operational Impacts

- Good
  - Improved policies based on Cookies
  - Better responsiveness under attack
- Bad
  - Anycasts
  - State-synchronization

Anycast Deployments

- Multiple implementations deployed at the same anycast node
- The deployed servers should share:
  - Same server cookie secret
  - Same cookie algorithm
- Clients should handle multiple cookies, if compliant, but…

The Real World

- Mix of servers with and without DNS Cookies
  - Different deployment schedule
  - Different software and different state of implementation
  - Different operators
  - Unconfigured server pick server secret at random
  - Different default algorithms
  - Incompatible algorithms
- There are deployments that change the server at anycast node very often (even every request) — like K-Root

The Solution

- Define a mandatory DNS Cookie algorithms
    - Both the crypto functions and how the input data into the function is processed
- Add SipHash — pseudo-random function (PRF)
    - Designed to network traffic authentications
    - Seems like best fit
- Define optional algorithms to implement:
    - HMAC-SHA256+
    - AES
- Remove non-cryptographically secure algorithms (FNV)
- Provide guidance to the DNS operators

Questions?

Photo by Whitney Wright on Unsplash