Cache Poisoning Protection Deployment Experience

Tianhao Chi, Puneet Sood
Google Public DNS

Presented at OARC 40

Agenda

Recap from OARC 38 Presentation

Case Randomization Deployment

DNS Cookies Deployment

DNS Cookies Deployment on Name servers

DNS-over-TLS to Authoritative (ADoT)

Concluding remarks

RECAP from OARC 38

- Cache poisoning is a risk for UDP queries
- Proposed mitigations reduce risk (port+query ID randomization, case randomization, DNS cookies)
- Most effective approach (DNS cookies) not deployed widely
- Additional mechanisms for protection exist
  - Nonce prepending (for delegation only zones - root and TLDs)
  - DNS-over-TLS

(for details see OARC 38 presentation)

RECAP: Google Public DNS Countermeasures

Resolvers replicated across metros with multiple servers in each metro.

Our countermeasures (Security Benefits):

- Randomize source ports, query ID, choice of name servers
- Prepending nonce label
- **Case Randomization (0x20)**
- **DNS Cookies**
- **DNS-over-TLS to Authoritative (ADoT)**

Google Public DNS Deployment Updates

Case Randomization: Deployment Experience

- Enabled in multiple metros (not all) around the world
  - Covers > 90% UDP traffic in each enabled metro
- Problem name servers added to actively maintained disable-list
  - 2000 name server IPs + a few subnets (total NS count: 1.5–2 million)
  - At least one large operator in the list

Observed problems

- Response case mismatch
- Error responses to mixed case queries
- No response (timeout) to mixed case queries
- Case for PTR record type sometimes not preserved
- [NEW] Occasional response case mismatch - discovered during deployment
  - Only observed with higher QPS

(Deployment announcement)

Case Randomization: Failure Mitigations

- disable-list ***will*** miss some broken name servers
- Primary fallback: mismatched response results in retry over TCP
- Server regression could generate TCP flood
- Additional fallback for consistent failures with a name server (in progress)
    - Disable case randomization with confirmation from other signals.
- Case randomization is disabled for PTR record type

DNS Cookies (RFC 7873): Deployment Experience

- Expanded manual configuration: primarily more TLDs
- Enabled In-line probing with production service: probe top ~400K IPs
- No probing based enablement for user queries yet
    - expected to cover < 12% of user queries
    - analyzing probe results to make a decision
- Probe results from LAX (% of nameservers)
    - Valid response with Server Cookie: 20.96%
    - Valid response with Client Cookie echo[1]: 1.32% [we consider as supported]
    - Valid response without Cookie: 75.81%
    - Failures: < 3%
    - Nameservers change from supported to unsupported

[1] RFC 7873 section 5.3 describes response processing. BIND, Knot do not consider "client cookie echo" as indicating server support.

DNS Cookies: Failure Mitigation

- Responses without valid cookies
  - attack or implementation issues? Latter seen, former hard to observe during testing
- Mitigation: Fall back to TCP
- Additional mitigation: Disable cookies if server completely drops support

DNS Cookies Deployment for Name Servers

Small increase since Oct 2022

Support Level	Nameserver July 2022	Nameserver Feb 2023	Query July 2022	Query Feb 2023
Full: Server Cookies	40.4%	42.20%	2.0%	2.38%
Echo: Client Cookies	0.8%	0.76%	10.0%	9.24%

DNS Cookies Deployment for Name Servers

Note: not specific to Google Public DNS

- Why is deployment among large operators low?
- Open-source name servers have compliant support
- RFC 7873 section 7 covers topic of incremental deployment
- Deployments using anycast IPs and server farms behind load-balancers
  - Need careful deployment to minimize resolvers seeing different behavior from same IP over a short period of time
- Avoid resolution failures during deployment
  - Clients dropping good responses without cookies should failover to other IPs for DNS zone
  - Client cookie echo could be an intermediate step?
- Experience from operators who have deployed or considering deploying DNS cookies?

DNS-over-TLS (ADoT): Deployment Experience

- Unilateral probing for DoT on by default
- TLS 1.3 session resumption not supported
- ADoT in use for ~700 nameserver IPs for 4.5% of egress traffic
  - ADoT is down as percent of total name server queries since Oct 2022.
- For name servers supporting DoT and UDP
  - Success rate: DoT (99.8%) slightly better than UDP
  - Average Latency: DoT (85 ms) vs UDP (93 ms)
- Top authoritative servers by traffic
  - Facebook, CDN77, one.com, Wikimedia
- Issues experienced with TLS connection management

ADoT: Operational Issues

- Servers closing TLS connections even if not idle
    - Closed after 10s in reaction to the next query (no response)
    - Unconditional close after 4s
    - Queries in flight over the connection fail
    - Requires repeating connection setup
- Google Public DNS: Mid-size operator connection count high for query volume
    - Google egress servers highly replicated per metro
    - Outbound load balancing results in many connections with low QPS per connection.
    - In progress: Optimizing outbound connection management for low volume servers.

Concluding Remarks

Google Public DNS Plans

- DNS Cookies
  - Enable based on probing if all misbehaviors can be mitigated sufficiently
  - Name servers: investigate safe rollout steps
  - Clarify behavior for client cookie echo response
  - Can we get significant name server traffic adoption?
    - Eliminates need for less elegant mechanisms
- ADoT scaling improvements
  - Optimize connection management (reduce count, resumption, metrics)
  - Share best practices?
- Case Randomization
  - Is it worth reviving draft-vixie-dnsext-dns0x20?

Thank you