**ISP Comprehensive DNS Monitoring Solution**

* The proposed solution should ensure the full implementation of the requirements and recommendations set out in RFC: 1034, 1035, 1123, 1183, 1706, 1712, 1886, 2163, 2181, 2230, 2308, 2538, 2671, 2672, 2782, 2845, 2874, 2008, 3123, 3152, 3225, 3226, 3401, 3402, 3403, 3404, 3596, 3597, 3655, 3645, 4033, 4034, 4035, 4343, 4398, 4470, 4509, 4892, 5001, 5011, 5155, 5702, 6174, draft-vandergaast-edns-client-subnet-02
* Support more than 400k QPS per node
* Solution must supoport HA configuration trough BGP with BFD
* The proposed solution should be based on a virtualization system, and the host for the DNS service is deployed as a virtual machine. All things being equal, preference will be given to a solution using the VMware hypervisor.
* Possibility to limit TTL for RR given to the subscriber, while saving the original TTL in the server cache. Configuration-adjustable parameter for resolver/cache.
* Proactive DNS records prefetch
* Multiple Views, not limited by specific number
* Multiple Resolvers, not limited by specific number
* Flexible binding clients CIDR to views
* Binding Views to Resolvers
* Policy engine with selection based on CIDR, View, Domain, Request Size, Query Type, Response Size, Destination address (on server), result (cname, dname, failure, nxdomain, nxrrset), query flags
* Policy engine must support actions to drop, rate-limit, servfail, refuse, set TC bit, answer with data, use another resolver
* Possibility to rate limiting clients based on QPS and Bytes, with client selection based on policy
* The proposed solution should provide protection against DNS Spoofing attacks through the use of randomization ID, randomization source port, the use of 0x20, as well as the transition to the TCP protocol to communicate with an authoritative DNS server in case of an attack.
* Ability to use DoH/DoT for client requests
* Ability to use DoH/DoT for request from resolver to auth DNS servers. Should be configuration-adjustable parameter per CIDR.
* DNS requests/response logging without performance penalty
* Logging should include requests from clients
* Logging should include answers to clients
* Logging should include requests from resolver to auth DNS servers
* Logging should include answers from auth DNS servers to resolver
* Logging should combine request/answer in one record
* The system should provide metrics in prom standard:
  + TOP Clients with duplicate requests
  + TOP Infected Clients (virus domains)
  + TOP Malware domains
  + TOP No-View Clients
  + TOP QPS Clients
  + TOP Formerr Clients
  + TOP high latency auth DNS servers
  + TOP Domains by QPS
  + TOP Domains by response size
  + TOP Core Domains
  + CPU utilization user/system
  + Counters: cache lookups, cache-misses, requests-received, responses-sent, requests-sent, responses-received, tcp-requests-received, tcp-responses-sent, tcp-requests-sent, tcp-responses-received, doh-requests-received, doh-responses-sent, doh-requests-sent, doh-responses-received, dot-requests-received, dot-responses-sent, dot-requests-sent, dot-responses-received , rate-limited-requests, requests-no-view, tcp-clients, responses-by-rcode (table), queries-by-type (table), formerr-loop-dropped, malformed-responses-dropped, dropped-recursions, id-spoofing--queries
* Metrics frequency up to 1 second
* SLA support 24/7 with 15 minutes response and no more than 4 hours recovery