PowerDNS

PROXYv2 Protocol for DNS

Passing source information from loadbalancers to DNS backends

Pieter Lexis **Peter van Dijk**

RIPE DNSWG, 25th of November 2021

POWERDNS

The Problem

- DNS set-ups can be fronted by loadbalancers/proxies
- Backend servers require true client IP address for ACL, views, or other purposes
- Proxies may not want to do extensive packet parsing and processing
- Backends may also want to know about transports used, including port numbers

POWERDNS

Existing Solutions

- EDNS Client Subnet
- X-Proxied-For (XPF)
- Private/bespoke EDNS option

POWERDNS

Drawbacks of EDNS0 Client Subnet

- Squatting existing EDNS options is **a bad idea**
- Requires parsing and modifying DNS packets in-flight
- No way to pass ECS from recursor to proxied auth
- No port information
- No transport protocol information

POWERDNS

Drawbacks of XPF and Bespoke Options

XPF

- Draft expired in the IETF
- Requires parsing and modifying DNS packets in-flight
- Requires special handling to not break TSIG

Private EDNS option

- Requires changes to all software in the chain
- Hard to debug with standard tools

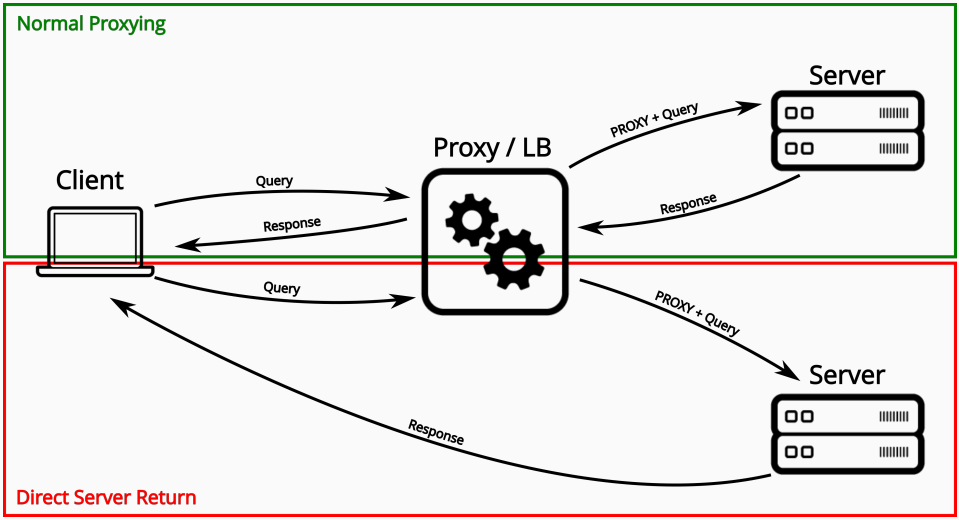
POWERDNS

The PROXYv2 Protocol

- Binary protocol
- Prefixes proxied data
- Passes v4, v6 addresses, ports, and protocol (TCP/UDP)
- Extensible with any number of arbitrary TLV fields
- Already supported by several loadbalancer vendors
- Goes well together with ECS used as intended

POWERDNS

PROXY Header Placement



Implementations

- Loadbalancers (HAProxy, f5) [TCP only]
- Webservers (nginx, Apache) [TCP only]
- dnsdist 1.5.0
- PowerDNS Authoritative Server 4.6.0
- PowerDNS Recursor 4.4.0
- Roadmapped for BIND9

POWERDNS

Further Reading

PROXY protocol specification

dnsdist documentation on using the PROXY protocol

ISC BIND9 issue for implementing PROXY protocol

POWERDNS

Questions?