DNSOP Working Group
Internet-Draft
Updates: RFC 2845, RFC 2931 (if approved) (if approved)
Intended status: Standards Track
Expires: September 6, 2018
March 05, 2018

R. Bellis
ISC
P. van Dijk
R. Gacogne
PowerDNS

DNS X-Proxied-For
draft-bellis-dnsop-xpf-04

Abstract

It is becoming more commonplace to install front end proxy devices in front of DNS servers to provide (for example) load balancing or to perform transport layer conversions.

This document defines a meta resource record that allows a DNS server to receive information about the client's original transport protocol parameters when supplied by trusted proxies.

Table of Contents

1. Introduction

It is becoming more commonplace to install front end proxy devices in front of DNS servers [RFC1035] to provide load balancing or to perform transport layer conversions (e.g. to add DNS over TLS [RFC7858] to a DNS server that lacks native support).

This has the unfortunate side effect of hiding the clients' source IP addresses from the server, making it harder to employ server-side technologies that rely on knowing those addresses (e.g. ACLs, DNS Response Rate Limiting, etc).

This document defines the XPF meta resource record (RR) that allows a DNS server to receive information about the client's original transport protocol parameters when supplied by trusted proxies.

Whilst in some circumstances it would be possible to re-use the Client Subnet EDNS Option [RFC7871] to carry a subset of this

information, a new RR is defined to allow both this feature and the Client Subnet Option to co-exist in the same packet.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The XPF RR is analogous to the HTTP "X-Forwarded-For" header, but in DNS the term "forwarder" is usually understood to describe a network component that sits on the outbound query path of a resolver.

Instead we use the term "proxy", which in this document means a network component that sits on the inbound query path in front of a recursive or authoritative DNS server, receiving DNS queries from clients and dispatching them to local servers.

3. Description

The XPF RR contains the entire 6-tuple (IP version, Layer 4 protocol, source address, destination address, source port and destination port) of the packet received from the client by the proxy.

The presence of the source address supports use of ACLs based on the client's IP address.

The source port allows for ACLs to support Carrier Grade NAT whereby different end-users might share a single IP address.

The destination address supports scenarios where the server behaviour depends upon the packet destination (e.g. BIND view's "match-destinations" option)

The protocol and destination port fields allow server behaviour to vary depending on whether DNS over TLS [RFC7858] or DNS over DTLS [RFC8094] are in use.

3.1. Client Handling

Stub resolvers, client-side proxy devices, and recursive resolvers MUST NOT add the XPF RR to DNS requests.

3.2. Request Handling

The rules in this section apply to processing of the XPF RR whether by a proxy device or a DNS server.

If this RR is received from a non-white-listed client the server MUST return a REFUSED response.

If a server finds this RR anywhere other than in the Additional Section of a request it MUST return a REFUSED response.

If the value of the RR's IP version field is not understood by the server it MUST return a REFUSED response.

If the length of the IP addresses contained in the RR are not consistent with that expected for the given IP version then the server MUST return a FORMERR response.

Servers MUST NOT send this RR in DNS responses.

3.3. Proxy Handling

For each request received, proxies MUST generate an XPF RR containing the 6-tuple representing the client's Layer 3 and Layer 4 headers and append it to the Additional Section of the request (updating the ARCOUNT field accordingly) before sending it to the intended DNS server.

If a valid XPF RR is received from a white-listed client the original XPF RR MUST be preserved instead.

3.4. Server Handling

When this RR is received from a white-listed client the DNS server SHOULD use the transport information contained therein in preference to the packet's own transport information for any data processing logic (e.g. ACLs) that would otherwise depend on the latter.

3.5. Wire Format

The XPF RR is formatted like any standard RR, but none of the fields except RDLENGTH and RDATA have any meaning in this specification. All multi-octet fields are transmitted in network order (i.e. big-endian).

The required values of the RR header fields are as follows:

NAME: MUST contain a single 0 octet (i.e. the root domain).

TYPE: MUST contain TBD1 (XPF).

CLASS: MUST contain 1 (IN).

TTL: MUST contain 0 (zero).

RDLENGTH: specifies the length in octets of the RDATA field.

The RDATA of the XPF RR is as follows:

```
                +0 (MSB)                            +1 (LSB)
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
   0: |     Unused    |   IP Version  |          Protocol             |
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
   2: |     Source Address Octet 0    |             ...               |
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
      |              ...             ///                              |
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
      |  Destination Address Octet 0  |             ...               |
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
      |              ...             ///                              |
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
      |                          Source Port                          |
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
      |                        Destination Port                       |
      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

Unused: Currently reserved. These bits MUST be zero unless redefined in a subsequent specification.

IP Version: The IP protocol version number used by the client, as defined in the IANA IP Version Number Registry [IANA-IP]. Implementations MUST support IPv4 (4) and IPv6 (6).

Protocol: The Layer 4 protocol number (e.g. UDP or TCP) as defined in the IANA Protocol Number Registry [IANA-PROTO].

Source Address: The source IP address of the client.

Destination Address: The destination IP address of the request, i.e. the IP address of the proxy on which the request was received.

Source Port: The source port used by the client.

Destination Port: The destination port of the request.

The length of the Source Address and Destination Address fields will be variable depending on the IP Version used by the client.

3.6. Presentation Format

XPF is a meta RR that cannot appear in master format zone files, but a standardised presentation format is defined here for use by debugging utilities that might need to display the contents of an XPF RR.

The Unused bits and the IP Version field are treated as a single octet and presented as an unsigned decimal integer with range 0 .. 255.

The Protocol field is presented as an unsigned decimal integer with range 0 .. 255.

The Source and Destination Address fields are presented either as IPv4 or IPv6 addresses according to the IP Version field. In the case of IPv6 the recommendations from [RFC5952] SHOULD be followed.

The Source and Destination Port fields are presented as unsigned decimal integers with range 0 .. 65535.

3.7. Signed DNS Requests

Any XPF RRs found in a packet MUST be ignored for the purposes of calculating or verifying any signatures used for Secret Key Transaction Authentication for DNS [RFC2845] or DNS Request and Transaction Signatures (SIG(0)) [RFC2931].

Typically it is expected that proxies will append the XPF RR to the packet after any existing TSIG or SIG(0) RRs, and that servers will remove the XPF RR from the packet prior to verification of the original signature, with the ARCOUNT field updated as appropriate.

If either TSIG or SIG(0) are configured between the proxy and server then any XPF RRs MUST be ignored when the proxy calculates the packet signature.

4. Security Considerations

If the white-list of trusted proxies is implemented as a list of IP addresses, the server administrator MUST have the ability to selectively disable this feature for any transport where there is a possibility of the proxy's source address being spoofed.

This does not mean to imply that use over UDP is impossible - if for example the network architecture keeps all proxy-to-server traffic on a dedicated network and clients have no direct access to the servers then the proxies' source addresses can be considered unspoofable.

5. Implementation status

[RFC Editor Note: Please remove this entire section prior to publication as an RFC.]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [RFC7942]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC7942], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

5.1. dnsdist

Support for adding an XPF RR to proxied packets is provided in the git version of dnsdist. The code point is configurable.

5.2. PowerDNS Recursor

Support for extracting the XPF RR from received packets (when coming from a trusted source) is available in the git version of the PowerDNS Recursor. The code point is configurable.

5.3. Wireshark

Support for dissecting XPF RRs is present in Wireshark 2.5.0, using a temporary code point of 65422.

6. Privacy Considerations

Used incorrectly, this RR could expose internal network information, however it is not intended for use on proxy / forwarder devices that sit on the client-side of a DNS request.

This specification is only intended for use on server-side proxy devices that are under the same administrative control as the DNS servers themselves. As such there is no change in the scope within which any private information might be shared.

Use other than as described above would be contrary to the principles of [RFC6973].

7. IANA Considerations

<< a copy of the RFC 6895 IANA RR TYPE application template will appear here >>

8. Acknowledgements

Mark Andrews, Robert Edmonds, Duane Wessels

9. References

9.1. Normative References

[IANA-IP] IANA, "IANA IP Version Registry", n.d., <http://www.iana.org/assignments/version-numbers/>.

[IANA-PROTO] IANA, "IANA Protocol Number Registry", n.d., <http://www.iana.org/assignments/protocol-numbers/>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2845] Vixie, P., Gudmundsson, O., Eastlake 3rd, D., and B. Wellington, "Secret Key Transaction Authentication for DNS (TSIG)", RFC 2845, DOI 10.17487/RFC2845, May 2000, <https://www.rfc-editor.org/info/rfc2845>.

[RFC2931] Eastlake 3rd, D., "DNS Request and Transaction Signatures (SIG(0)s)", RFC 2931, DOI 10.17487/RFC2931, September 2000, <https://www.rfc-editor.org/info/rfc2931>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <https://www.rfc-editor.org/info/rfc6973>.

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <https://www.rfc-editor.org/info/rfc7942>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

9.2. Informative References

[RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", RFC 5952, DOI 10.17487/RFC5952, August 2010, <https://www.rfc-editor.org/info/rfc5952>.

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/info/rfc7858>.

[RFC7871] Contavalli, C., van der Gaast, W., Lawrence, D., and W. Kumari, "Client Subnet in DNS Queries", RFC 7871, DOI 10.17487/RFC7871, May 2016, <https://www.rfc-editor.org/info/rfc7871>.

[RFC8094] Reddy, T., Wing, D., and P. Patil, "DNS over Datagram Transport Layer Security (DTLS)", RFC 8094, DOI 10.17487/RFC8094, February 2017, <https://www.rfc-editor.org/info/rfc8094>.

Authors' Addresses

Ray Bellis
Internet Systems Consortium, Inc.
950 Charter Street
Redwood City CA 94063
USA

Phone: +1 650 423 1200
Email: ray@isc.org

Peter van Dijk
PowerDNS.COM B.V.
Den Haag
The Netherlands

Email: peter.van.dijk@powerdns.com

Remi Gacogne
PowerDNS.COM B.V.
Den Haag
The Netherlands

Email: remi.gacogne@powerdns.com