Workgroup: ipsecme
Internet-Draft: draft-ietf-ipsecme-add-ike-10
Published: 26 March 2023
Intended Status: Standards Track
Expires: 27 September 2023
Authors: M. Boucadair (Orange), T. Reddy (Nokia), D. Wing (Citrix), V. Smyslov (ELVIS-PLUS)

Internet Key Exchange Protocol Version 2 (IKEv2) Configuration for Encrypted DNS

Abstract

This document specifies new Internet Key Exchange Protocol Version 2 (IKEv2) Configuration Payload Attribute Types to assign DNS resolvers that support encrypted DNS protocols, such as DNS-over-HTTPS (DoH), DNS-over-TLS (DoT), and DNS-over-QUIC (DoQ).

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document specifies a mechanism to assign encrypted DNS configurations to an Internet Key Exchange Protocol Version 2 (IKEv2) [RFC7296] initiator. Specifically, it assigns one or more Authentication Domain Names (ADNs) of DNS resolvers that support encrypted DNS protocols. The specific protocols supported are described using the Service Parameters format defined in [I-D.ietf-dnsop-svcb-https]; supported protocols include DNS-over-HTTPS (DoH) [RFC8484], DNS-over-TLS (DoT) [RFC7858], and DNS-over-QUIC (DoQ) [RFC9250].

This document introduces three new IKEv2 Configuration Payload Attribute Types (Section 3) to add support for encrypted DNS resolvers. The ENCDNS_IP4 and ENCDNS_IP6 attribute types (Section 3.1) are used to provision ADNs, a list of IP addresses, and a set of service parameters. The ENCDNS_DIGEST_INFO attribute (Section 3.2) additionally allows a specific resolver certificate to be indicated by the IKEv2 responder.

Sample use cases are described in Appendix A. The Configuration Payload Attribute Types defined in this document are not specific to these deployments, but can also be used in other deployment contexts. It is out of the scope of this document to provide a comprehensive list of deployment contexts.

The encrypted DNS resolver hosted by a VPN provider can get a domain-validate certificate from a public Certificate Authority (CA). The VPN client does not need to be provisioned with the root certificate of a private CA to authenticate the certificate of the encrypted DNS resolvers. The encrypted DNS resolver can run on private IP addresses and its access can be restricted to clients connected to the VPN.

Note that, for many years, typical designs have often considered that the DNS resolver was usually located inside the protected domain, but could be located outside of it. With encrypted DNS, the latter option becomes plausible.

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.

This document uses the terms defined in [RFC8499].

Also, this document uses the terms defined in [RFC7296]. In particular, readers should be familiar with "initiator" and "responder" terms used in that document.

This document makes use of the following terms:

Do53: refers to unencrypted DNS.

Encrypted DNS: refers to a scheme where DNS messages are sent over an encrypted channel. Examples of encrypted DNS are DoT, DoH, and DoQ.

ENCDNS_IP*: refers to any IKEv2 Configuration Payload Attribute Types defined in Section 3.1.

3. IKEv2 Configuration Payload Attribute Types for Encrypted DNS

3.1. ENCDNS_IP* Configuration Payload Attributes

The ENCDNS_IP* IKEv2 Configuration Payload Attribute Types, ENCDNS_IP4 and ENCDNS_IP6, are used to configure encrypted DNS resolvers to an initiator. Both attribute types share the format that is shown in Figure 1. The information included in these attributes adheres to the recommendation in Section 3.1.9 of [I-D.ietf-add-dnr].

```
                     1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-----------------------------+-------------------------------+
|R|         Attribute Type      |            Length             |
+-+-----------------------------+---------------+---------------+
|       Service Priority        | Num Addresses |  ADN Length   |
+-------------------------------+---------------+---------------+
~                         IP Addresses                          ~
+---------------------------------------------------------------+
~                  Authentication Domain Name                   ~
+---------------------------------------------------------------+
~                Service Parameters (SvcParams)                 ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: Attributes Format

The description of the fields of the attribute shown in Figure 1 is as follows:

* R (Reserved, 1 bit) - This bit MUST be set to zero and MUST be ignored on receipt (see Section 3.15.1 of [RFC7296] for details).

* Attribute Type (15 bits) - Identifier for Configuration Attribute Type. This is set to TBA1 for ENCDNS_IP4 or TBA2 for ENCDNS_IP6, as registered in Section 8.

* Length (2 octets, unsigned integer) - Length of the enclosed data in octets. In particular, this field is set to:

  - 0 if the Configuration payload has types CFG_REQUEST (if no specific DNS resolver is requested) or CFG_ACK. If the 'Length' field is set to 0, then later fields shown in Figure 1 are not present.

  - (4 + Length of the ADN + N * 4 + Length of SvcParams) for ENCDNS_IP4 attributes if the Configuration payload has types CFG_REQUEST or CFG_REPLY or CFG_SET; N being the number of included IPv4 addresses ('Num addresses').

  - (4 + Length of the ADN + N * 16 + Length of SvcParams) for ENCDNS_IP6 attributes if the Configuration payload has types CFG_REQUEST or CFG_REPLY or CFG_SET; N being the number of included IPv6 addresses ('Num addresses').

* Service Priority (2 octets) - The priority of this attribute compared to other ENCDNS_IP* instances. This 16-bit unsigned integer is interpreted following the rules specified in Section 2.4.1 of [I-D.ietf-dnsop-svcb-https].

AliasMode (Section 2.4.2 of [I-D.ietf-dnsop-svcb-https]) is not supported because such a mode will trigger additional Do53 queries while the data can be supplied directly in the IKE response. As such, this field MUST NOT be set to 0.

* Num Addresses (1 octet) - Indicates the number of enclosed IPv4 (for ENCDNS_IP4) or IPv6 (for ENCDNS_IP6) addresses. This value MUST NOT be set to 0 if the Configuration payload is of type CFG_REPLY or CFG_SET.

* ADN Length (1 octet) - Indicates the length of the "Authentication Domain Name" field in octets. When set to '0', this means that no ADN is enclosed in the attribute.

* IP Address(es) (variable) - Includes one or more IP addresses that can be used to reach the encrypted DNS resolver identified by the Authentication Domain Name. For ENCDNS_IP4 this field contains one or more 4-octet IPv4 addresses, and for ENCDNS_IP6 this field contains one or more 16-octet IPv6 addresses.

* Authentication Domain Name (variable) - A fully qualified domain name of the encrypted DNS resolver, in DNS presentation format and using an Internationalized Domain Names for Applications (IDNA) A-label [RFC5890]. The name MUST NOT contain any terminators (e.g., NULL, CR).

  An example of a valid ADN for DoH server is "doh1.example.com".

* Service Parameters (SvcParams) (variable) - Specifies a set of service parameters that are encoded following the rules in Section 2.1 of [I-D.ietf-dnsop-svcb-https]. Section 3.1.5 of [I-D.ietf-add-dnr] lists a set of service parameters that are recommended to be supported by implementations.

  The service parameters MUST NOT include "ipv4hint" or "ipv6hint" SvcParams as they are superseded by the included IP addresses.

  If no port service parameter is included, this indicates that default port numbers should be used. As a reminder, the default port number is 853 for DoT (Section 6 of [RFC7858]), 443 for DoH (Section 8.1 of [RFC8484]), and 853 for DoQ (Section 8 of [RFC9250]).

  The service parameters apply to all IP addresses in the ENCDNS_IP* Configuration Payload Attribute.

3.2. ENCDNS_DIGEST_INFO Configuration Payload Attribute

The ENCDNS_DIGEST_INFO configuration payload attribute allows IKEv2 responders to specify a certificate digest that initiators can use when validating TLS connections to encrypted resolvers. This attribute can also be sent by the initiator to request specific hash algorithms for such digests. The format of ENCDNS_DIGEST_INFO attribute if the Configuration payload has type CFG_REQUEST is shown in Figure 2.

```
                     1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-----------------------------+-------------------------------+
|R|         Attribute Type      |            Length             |
+-+-------------+---------------+-------------------------------+
| Num Hash Algs |  ADN Length   |                               |
+---------------+---------------+                               +
~                List of Hash Algorithm Identifiers             ~
+---------------------------------------------------------------+
```

Figure 2: ENCDNS_DIGEST_INFO Attribute Format in CFG_REQUEST

The description of the fields of the attribute shown in Figure 2 is as follows:

- *R (Reserved, 1 bit) - This bit MUST be set to zero and MUST be ignored on receipt (see Section 3.15.1 of [RFC7296] for details).
- *Attribute Type (15 bits) - Identifier for Configuration Attribute Type; is set to TBA3 value listed in Section 8.
- *Length (2 octets, unsigned integer) - Length of the enclosed data in octets. This field MUST be set to "2 + 2 * number of included hash algorithm identifiers".
- *Num Hash Algs (1 octet) - Indicates the number of included hash algorithm identifiers. This field MUST be set to "(Length - 2)/ 2".
- *ADN Length (1 octet) - MUST be set to 0.
- *List of Hash Algorithm Identifiers (variable) - Specifies a list of 16-bit hash algorithm identifiers that are supported by the encrypted DNS client.

  The values of this field are taken from the Hash Algorithm Identifiers of IANA's "Internet Key Exchange Version 2 (IKEv2) Parameters" registry [IANA-IKE-HASH].

  There is no padding between the hash algorithm identifiers.

  Note that SHA2-256 is mandatory to implement (see Section 5).

The format of ENCDNS_DIGEST_INFO attribute if the Configuration payload has types CFG_REPLY or CFG_SET is shown in Figure 3.

```
                     1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-----------------------------+-------------------------------+
|R|         Attribute Type      |            Length             |
+-+-----------------------------+---------------+---------------+
| Num Hash Algs |  ADN Length   |                               |
+---------------+---------------+                               +
~                 Authentication Domain Name                    ~
+-------------------------------+-------------------------------+
| Hash Algorithm Identifier     |                               ~
+-------------------------------+                               +
~                      Certificate Digest                       ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: ENCDNS_DIGEST_INFO Attribute Format in CFG_REPLY or CFG_SET

The description of the fields of the attribute shown in Figure 2 is as follows:

- *R (Reserved, 1 bit) - This bit MUST be set to zero and MUST be ignored on receipt (see Section 3.15.1 of [RFC7296] for details).

- *Attribute Type (15 bits) - Identifier for Configuration Attribute Type; is set to TBA3 value listed in Section 8.

- *Length (2 octets, unsigned integer) - Length of the data in octets.

- *Num Hash Algs (1 octet) - MUST be set to 1.

- *ADN Length (1 octet) - Indicates the length of the "Authentication Domain Name" field in octets. When set to '0', this means that the digest applies on the ADN conveyed in the ENCDNS_IP* Configuration Payload Attribute(s).

- *Authentication Domain Name (variable) - A fully qualified domain name of the encrypted DNS resolver following the syntax defined in [RFC5890]. The name MUST NOT contain any terminators (e.g., NULL, CR). A name is included only when multiple ADNs are included in the ENCDNS_IP* Configuration Payload Attributes.

- *Hash Algorithm Identifier (2 octets) - Specifies the 16-bit hash algorithm identifier selected by the DNS resolver to generate the digest of its certificate.

* Certificate Digest (variable) - This field includes the Subject Public Key Info (SPKI) hash (Section 5) of the encrypted DNS resolver certificate using the algorithm identified in the 'Hash Algorithm Identifier' field. The length of this field is "Length - 4 - ADN Length".

The ENCDNS_DIGEST_INFO attribute may be present in the Configuration payload of CFG_ACK. In such a case, the ENCDNS_DIGEST_INFO MUST be returned with zero-length data.

As discussed in Section 3.15.1 of [RFC7296], there are no defined uses for the CFG_SET/CFG_ACK exchange. The use of the ENCDNS_DIGEST_INFO attribute for these messages is provided for completeness.

4. IKEv2 Protocol Exchange

This section describes how the attributes defined in Section 3 are used to configure an IKEv2 initiator with one or more encrypted DNS resolvers. As a reminder, badly formatted attributes or unacceptable fields are handled as per Section 2.21 of [RFC7296].

Initiators first indicate support for encrypted DNS by including ENCDNS_IP* attributes in their CFG_REQUEST payloads. Responders supply encrypted DNS configuration by including ENCDNS_IP* attributes in their CFG_REPLY payloads. Concretely:

If the initiator supports encrypted DNS, it includes either or both of the ENCDNS_IP4 and ENCDNS_IP6 attributes in its CFG_REQUEST. If the initiator does not want to request specific DNS resolvers, it sets the Length field to 0 for the attribute. For a given attribute type, the initiator MAY send either an empty attribute or a list of distinct suggested resolvers. The initiator MAY also include the ENCDNS_DIGEST_INFO attribute with a list of hash algorithms that are supported by the encrypted DNS client.

If the request includes multiple bitwise identical attributes, only the first occurrence is processed, and the rest SHOULD be ignored by the responder. The responder MAY discard the full request if the count of repeated attributes exceeds an (implementation specific) threshold.

For each ENCDNS_IP* attribute from the CFG_REQUEST, if the responder supports the corresponding address family, and absent any policy restrictions, the responder sends back ENCDNS_IP* attribute(s) in the CFG_REPLY with an appropriate list of IP addresses, service parameters, and an ADN. The list of IP addresses MUST include at least one IP address. The service parameters MUST include at least the "alpn" service parameter.

The responder MAY ignore suggested values from the initiator (if any). Multiple instances of the same ENCDNS_IP* attribute MAY be returned if distinct ADNs or service parameters need to be assigned to the initiator. In such instances, the different attributes can have matching or distinct IP addresses. These instances MUST be presented to a local DNS client following their service priority (i.e., smaller service priority values indicates a higher preference).

In addition, the responder MAY return the ENCDNS_DIGEST_INFO attribute to convey a digest of the certificate of the encrypted DNS and the identifier of the hash algorithm that is used to generate the digest.

If the CFG_REQUEST includes an ENCDNS_IP* attribute but the CFG_REPLY does not include an ENCDNS_IP* matching the requested address family, this is an indication that requested address family is not supported by the responder or the responder is not configured to provide corresponding resolver addresses.

If the initiator receives both ENCDNS_IP* and INTERNAL_IP6_DNS (or INTERNAL_IP4_DNS) attributes, it is RECOMMENDED that the initiator uses the encrypted DNS resolvers.

The DNS client establishes an encrypted DNS session (e.g., DoT, DoH, DoQ) with the address(es) conveyed in ENCDNS_IP* and uses the mechanism discussed in Section 8 of [RFC8310] to authenticate the DNS resolver certificate using the authentication domain name conveyed in ENCDNS_IP*.

If the CFG_REPLY includes an ENCDNS_DIGEST_INFO attribute, the client has to create an SPKI hash (Section 5) of the DNS resolver certificate received in the TLS handshake using the negotiated hash algorithm in the ENCDNS_DIGEST_INFO attribute. If the computed digest for an ADN matches the one sent in the ENCDNS_DIGEST_INFO attribute, the encrypted DNS resolver certificate is successfully validated. If so, the client continues with the TLS connection as normal. Otherwise, the client MUST treat the resolver certificate validation failure as a non-recoverable error. This approach is similar to certificate usage PKIX-EE(1) with selector SPKI(1) defined in [RFC7671] but without PKIX validation.

If the IPsec connection is a split-tunnel configuration and the initiator negotiated INTERNAL_DNS_DOMAIN as per [RFC8598], the DNS client resolves the internal names using ENCDNS_IP* DNS resolvers.

Note: [RFC8598] requires INTERNAL_IP6_DNS (or INTERNAL_IP4_DNS) attribute to be mandatory present when INTERNAL_DNS_DOMAIN is included. This specification relaxes that constraint in the presence of ENCDNS_IP* attributes. That is, if ENCDNS_IP* attributes are supplied, it is allowed for responders to include INTERNAL_DNS_DOMAIN even in the absence of INTERNAL_IP6_DNS (or INTERNAL_IP4_DNS) attributes.

5. Subject Public Key Info (SPKI) Hash

The SPKI hash of the encrypted DNS resolver certificate is the output of a cryptographic hash algorithm whose input is the DER-encoded ASN.1 representation of the SPKI.

Implementations MUST support SHA2-256 [RFC6234].

6. Security Considerations

This document adheres to the security considerations defined in [RFC7296]. In particular, this document does not alter the trust on the DNS configuration provided by a responder.

Networks are susceptible to internal attacks as discussed in Section 3.2 of [I-D.arkko-farrell-arch-model-t]. Hosting encrypted DNS resolvers even in case of split-VPN configuration minimizes the attack vector (e.g., a compromised network device cannot monitor/ modify DNS traffic). This specification describes a mechanism to restrict access to the DNS messages to only the parties that need to know.

The initiator may trust the encrypted DNS resolvers supplied by means of IKEv2 from a trusted responder more than the locally provided DNS resolvers, especially in the case of connecting to unknown or untrusted networks (e.g., coffee shops or hotel networks).

If the IKEv2 responder has used NULL Authentication method [RFC7619] to authenticate itself, the initiator MUST NOT use returned ENCDNS_IP* resolvers configuration unless it is pre-configured, e.g., in the operating system or the application.

This specification does not extend the scope of accepting DNSSEC trust anchors beyond the usage guidelines defined in Section 6 of [RFC8598].

7. Privacy Considerations

As discussed in [RFC9076], the use of encrypted DNS does not reduce the data available in the DNS resolver. For example, the reader may refer to Section 8 of [RFC8484] or Section 7 of [RFC9250] for a discussion on specific privacy considerations to encrypted DNS.

8. IANA Considerations

This document requests IANA to assign the following new IKEv2 Configuration Payload Attribute Types from the "IKEv2 Configuration Payload Attribute Types" namespace available at [IANA-IKE-CFG].

Value	Attribute Type	Multi-Valued	Length	Reference
TBA1	ENCDNS_IP4	YES	0 or more	RFC XXXX
TBA2	ENCDNS_IP6	YES	0 or more	RFC XXXX
TBA3	ENCDNS_DIGEST_INFO	YES	0 or more	RFC XXXX

9. Acknowledgements

Many thanks to Yoav Nir, Christian Jacquenet, Paul Wouters, and Tommy Pauly for the review and comments.

Yoav and Paul suggested the use of one single attribute carrying both the name and an IP address instead of depending on the existing INTERNAL_IP6_DNS and INTERNAL_IP4_DNS attributes.

Thanks to Tero Kivinen for the Shepherd review and Roman Danyliw for the AD review.

Thanks to Stewart Bryant for the gen-art review, Dhruv Dhody for the ops-dir review, and Patrick Mevzek for the dns-dir review.

10. References

10.1. Normative References

[I-D.ietf-dnsop-svcb-https] Schwartz, B. M., Bishop, M., and E. Nygren, "Service binding and parameter specification via the DNS (DNS SVCB and HTTPS RRs)", Work in Progress, Internet-Draft, draft-ietf-dnsop-svcb-https-12, 11 March 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-dnsop-svcb-https-12>.

[IANA-IKE-HASH] "IKEv2 Hash Algorithms", <https://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xhtml#hash-algorithms>.

[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>.

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, DOI 10.17487/RFC5890, August 2010, <https://www.rfc-editor.org/info/rfc5890>.

[RFC6234] Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, DOI 10.17487/RFC6234, May 2011, <https://www.rfc-editor.org/info/rfc6234>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[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>.

[RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", RFC 8310, DOI 10.17487/RFC8310, March 2018, <https://www.rfc-editor.org/info/rfc8310>.

10.2. Informative References

[I-D.arkko-farrell-arch-model-t] Arkko, J. and S. Farrell, "Challenges and Changes in the Internet Threat Model", Work in Progress, Internet-Draft, draft-arkko-farrell-arch-model-t-04, 14 July 2020, <https://datatracker.ietf.org/api/v1/doc/document/draft-arkko-farrell-arch-model-t/>.

[I-D.ietf-add-dnr] Boucadair, M., Reddy.K, T., Wing, D., Cook, N., and T. Jensen, "DHCP and Router Advertisement Options for the Discovery of Network-designated Resolvers (DNR)", Work in Progress, Internet-Draft, draft-ietf-add-dnr-14, 13 March 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-add-dnr-14>.

[IANA-IKE-CFG] "IKEv2 Configuration Payload Attribute Types", <https://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xhtml#ikev2-parameters-21>.

[RFC7619] Smyslov, V. and P. Wouters, "The NULL Authentication Method in the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 7619, DOI 10.17487/RFC7619, August 2015, <https://www.rfc-editor.org/info/rfc7619>.

[RFC7671] Dukhovni, V. and W. Hardaker, "The DNS-Based Authentication of Named Entities (DANE) Protocol: Updates and Operational Guidance", RFC 7671, DOI 10.17487/

RFC7671, October 2015, <https://www.rfc-editor.org/info/rfc7671>.

[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>.

[RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <https://www.rfc-editor.org/info/rfc8484>.

[RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", BCP 219, RFC 8499, DOI 10.17487/RFC8499, January 2019, <https://www.rfc-editor.org/info/rfc8499>.

[RFC8598] Pauly, T. and P. Wouters, "Split DNS Configuration for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 8598, DOI 10.17487/RFC8598, May 2019, <https://www.rfc-editor.org/info/rfc8598>.

[RFC9076] Wicinski, T., Ed., "DNS Privacy Considerations", RFC 9076, DOI 10.17487/RFC9076, July 2021, <https://www.rfc-editor.org/info/rfc9076>.

[RFC9250] Huitema, C., Dickinson, S., and A. Mankin, "DNS over Dedicated QUIC Connections", RFC 9250, DOI 10.17487/RFC9250, May 2022, <https://www.rfc-editor.org/info/rfc9250>.

Appendix A. Sample Deployment Scenarios

A.1. Roaming Enterprise Users

In this Enterprise scenario (Section 1.1.3 of [RFC7296]), a roaming user connects to the Enterprise network through an IPsec tunnel. The split-tunnel Virtual Private Network (VPN) configuration allows the endpoint to access hosts that reside in the Enterprise network [RFC8598] using that tunnel; other traffic not destined to the Enterprise does not traverse the tunnel. In contrast, a non-split-tunnel VPN configuration causes all traffic to traverse the tunnel into the Enterprise.

For both split- and non-split-tunnel configurations, the use of encrypted DNS instead of Do53 provides privacy and integrity protection along the entire path (rather than just to the VPN termination device) and can communicate the encrypted DNS resolver policies.

For split-tunnel VPN configurations, the endpoint uses the Enterprise-provided encrypted DNS resolver to resolve internal-only domain names. These names may be configured to the endpoints using Enterprise-specific provisioning mechanisms or the INTERNAL_DNS_DOMAIN attribute.

For non-split-tunnel VPN configurations, the endpoint uses the Enterprise-provided encrypted DNS resolver to resolve both internal and external domain names.

Enterprise networks are susceptible to internal and external attacks. To minimize that risk all enterprise traffic is encrypted (Section 2.1 of [I-D.arkko-farrell-arch-model-t]).

A.2. VPN Service Provider

Legacy VPN service providers usually preserve end-users' data confidentiality by sending all communication traffic through an encrypted tunnel. A VPN service provider can also provide guarantees about the security of the VPN network by filtering malware and phishing domains.

Browsers and operating systems support DoH/DoT; VPN providers may no longer expect DNS clients to fall back to Do53 just because it is a closed network.

The encrypted DNS resolver hosted by the VPN service provider can be securely discovered by the endpoint using the IKEv2 attributes specified in Section 3.1.

A.3. DNS Offload

VPN service providers typically allow split-tunnel VPN configuration in which users can choose applications that can be excluded from the tunnel. For example, users may exclude applications that restrict VPN access.

The encrypted DNS resolver hosted by the VPN service provider can be securely discovered by the endpoint using the IKEv2 attributes specified in Section 3.1.

Appendix B. Examples

Figure 4 depicts an example of a CFG_REQUEST to request the configuration of IPv6 DNS resolvers without providing any suggested values. In this example, the initiator uses the ENCDNS_DIGEST_INFO attribute to indicate that the encrypted DNS client supports SHA2-256 (2), SHA2-384 (3), and SHA2-512 (4) hash algorithms. The label of these algorithms is taken from [IANA-IKE-HASH]. The use of

INTERNAL_IP6_ADDRESS is explained in [RFC7296]; it is thus not reiterated here.

```
CP(CFG_REQUEST) =
  INTERNAL_IP6_ADDRESS()
  INTERNAL_IP6_DNS()
  ENCDNS_IP6()
  ENCDNS_DIGEST_INFO(0, (SHA2-256, SHA2-384, SHA2-512))
```

Figure 4: Example of CFG_REQUEST

Figure 5 depicts an example of a CFG_REPLY that can be sent by a responder as a response the above CFG_REQUEST. This response indicates the following information to identify the encrypted DNS resolver:

- Its IPv6 address (2001:db8:99:88:77:66:55:44)
- Its authentication domain name (doh.example.com)
- Its supported HTTP version (h2)
- The relative form of the URI Template (/dns-query{?dns})
- The SPKI hash of the resolver's certificate using SHA2-256 (8b6e7a5971cc6bb0b4db5a71...)

```
CP(CFG_REPLY) =
  INTERNAL_IP6_ADDRESS(2001:db8:0:1:2:3:4:5/64)
  ENCDNS_IP6(1, 1, 15,
              (2001:db8:99:88:77:66:55:44),
              "doh.example.com",
              (alpn=h2 dohpath=/dns-query{?dns}))
  ENCDNS_DIGEST_INFO(0, SHA2-256,
                      8b6e7a5971cc6bb0b4db5a71...)
```

Figure 5: Example of CFG_REPLY

In this example, no ADN is included in the ENCDNS_DIGEST_INFO attribute because only one ADN is provided in the ENCDNS_IP6 attribute. There is no ambiguity to identify the encrypted resolver associated with the supplied digest.

An initiator may provide suggested values in the CFG_REQUEST when requesting an encrypted DNS resolver. For example, the initiator may:

* Indicate a preferred resolver that is identified by an IPv6 address (see Figure 6).

```
CP(CFG_REQUEST) =
  INTERNAL_IP6_ADDRESS()
  INTERNAL_IP6_DNS()
  ENCDNS_IP6(1, 1, 0,
                (2001:db8:99:88:77:66:55:44))
```

Figure 6: Example of CFG_REQUEST with a Preferred Resolver Identified by Its IP Address

* Indicate a preferred resolver that is identified by an ADN (see Figure 7).

```
CP(CFG_REQUEST) =
  INTERNAL_IP6_ADDRESS()
  INTERNAL_IP6_DNS()
  ENCDNS_IP6(1, 0, 15, "doh.example.com")
```

Figure 7: Example of CFG_REQUEST with a Preferred Resolver Identified by Its ADN

* Indicate a preferred transport protocol (DoT, in the example depicted in Figure 8)

```
CP(CFG_REQUEST) =
  INTERNAL_IP6_ADDRESS()
  INTERNAL_IP6_DNS()
  ENCDNS_IP6(1, 0, 0, (alpn=dot))
```

Figure 8: Example of CFG_REQUEST with a Preferred Transport Protocol

* or any combination thereof.

An initiator may also indicate that it supports Split DNS by including the INTERNAL_DNS_DOMAIN attribute in a CFG_REQUEST as shown in Figure 9. In this example, the initiator does not indicate any preference for the requested encrypted DNS server nor which DNS queries will be forwarded through the IPsec tunnel.

```
CP(CFG_REQUEST) =
  INTERNAL_IP6_ADDRESS()
  INTERNAL_IP6_DNS()
  ENCDNS_IP6()
  INTERNAL_DNS_DOMAIN()
```

Figure 9: Example of CFG_REQUEST with Support of Split DNS

Figure 10 shows an example of a reply of the responder. Absent any prohibited local policy, the initiator uses the encrypted DNS server (doh.example.com) for any subsequent DNS queries for "example.com" and its subdomains.

```
CP(CFG_REPLY) =
  INTERNAL_IP6_ADDRESS(2001:db8:0:1:2:3:4:5/64)
  ENCDNS_IP6(1, 1, 15,
                (2001:db8:99:88:77:66:55:44),
                "doh.example.com",
                (alpn=h2 dohpath=/dns-query{?dns}))
  INTERNAL_DNS_DOMAIN(example.com)
```

Figure 10: Example of CFG_REPLY with INTERNAL_DNS_DOMAIN

Authors' Addresses

Mohamed Boucadair
Orange
35000 Rennes
France

Email: mohamed.boucadair@orange.com

Tirumaleswar Reddy
Nokia
India

Email: kondtir@gmail.com

Dan Wing
Citrix Systems, Inc.
United States of America

Email: dwing-ietf@fuggles.com

Valery Smyslov
ELVIS-PLUS
Russian Federation

Email: svan@elvis.ru