Workgroup: Internet Engineering Task Force
Internet-Draft:
draft-dnsop-dnssec-extension-pkix-01
Published: 8 March 2023
Intended Status: Standards Track
Expires: 9 September 2023
Authors: H. Lee
Seoul National University
T. Kwon
Seoul National University

DNSSEC Extension by Using PKIX Certificates

Abstract

The Domain Name System Security Extensions (DNSSEC) were standardized a couple of decades ago but it has not been widely deployed. Thus, a vast majority of DNS messages in the real world are still vulnerable to various kinds of integrity attacks like cache poisoning attacks. This document describes a mechanism that extends the current DNSSEC protocol in such a way that guarantees the integrity of DNS messages using PKIX certificates without any dependencies on other entities in the DNS infrastructure.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 9 September 2023.

document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction

The DNSSEC [RFC4033] was standardized to provide integrity and authentication of DNS records. After twenty years of its introduction, DNSSEC is widely deployed for top-level domains. However, its deployment rates in second-level domains are low and most DNS messages in the real world are still vulnerable to tampering like cache poisoning attacks.

In addition, to support DNSSEC, it is necessary that a domain publishes DS records to its upper zone (i.e., parent zone) to establish a chain of trust. Usually, this process requires manual intervention of domain owners which often results in errors. Thus, it is reported that a large number of domains do not have their corresponding DS records in their upper zones [DNSSEC-Deployment], which results in DNSSEC validation failures. Thus, we need a more practical and deployable mechanism to guarantee the integrity and authenticity of DNS records.

This document specifies DNSSEC-PKI to provide the integrity and authentication of DNS Resource Record sets (RRsets) by leveraging PKIX [RFC5280] certificates. DNSSEC-PKI offers domain owners another option to provide the DNS integrity without requiring the cooperation (or dependencies) of other entities in the DNS infrastructure (e.g., establishing a chain of trust across zones). For this purpose, DNSSEC-PKI exploits PKIX certificates widely used by most domains. Thus, DNSSEC-PKI allows a domain (or a zone) to use a public/private key pair from its certificate rather than generating a distinct key for DNS. This document defines the requirements and operations of domains and validators (e.g., resolvers) to support DNSSEC-PKI.

1.1. Requirements Language

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.

2. DNSSEC-PKI Overview

DNSSEC-PKI is designed to provide a practical and deployable way that guarantees the integrity and authenticity of DNSSEC RRsets. To achieve this, we need a mechanism that does not have dependencies on other entities in the DNS infrastructure.

2.1. Design Requirements

We consider two requirements when designing DNSSEC-PKI.

1. Minimum or no changes of other entities in the DNS infrastructure:

   * Requiring changes (or cooperations) from other entities causes dependency on those entities in the DNSSEC operations. Thus, DNSSEC-PKI should minimally require a change of other entities in the DNS infrastructure (e.g. parent zones and DNS registrars).

2. Maximum reuse of the current DNSSEC standards and DNS infrastructure:

   * DNSSEC-PKI should work with the existing DNS in a way that maximally reuses the current infrastructure. That is, DNSSEC-PKI seeks to achieve compatibility with the current DNSSEC standards.

2.2. The Operations of DNSSEC-PKI

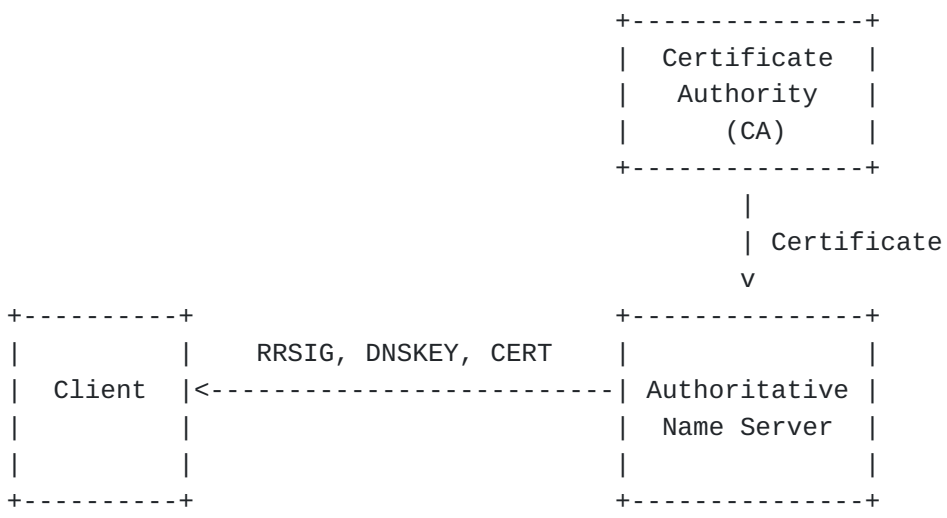

Figure 1: Entities in DNSSEC-PKI

To satisfy the design requirements, we leverage PKIX certificates issued by CAs. A domain generates a private/public key pair. Also, it asks a CA to issue a certificate that contains its domain name and the public key. It then uses the private key to generate the signatures of its DNS RRsets. Each signature (for each RRset) is published as a DNS record. Here, we reuse an RRSIG record [RFC4034], which is used to store the signature of an RRset. The domain publishes the public key in the certificate as a DNSKEY record [RFC4034]. Also the certificates in the certificate chain are published as CERT records [RFC4398]. The signature (i.e., RRSIG record) guarantees the integrity and authenticity of the corresponding DNS RRset and the certificate guarantees that the signature is generated by the domain owner's private key.

A validator (e.g., a client or a resolver) can check the integrity of a given RRset by fetching the corresponding signature (RRSIG), a public key (DNSKEY), and a certificate chain (CERTs) and verifying them.

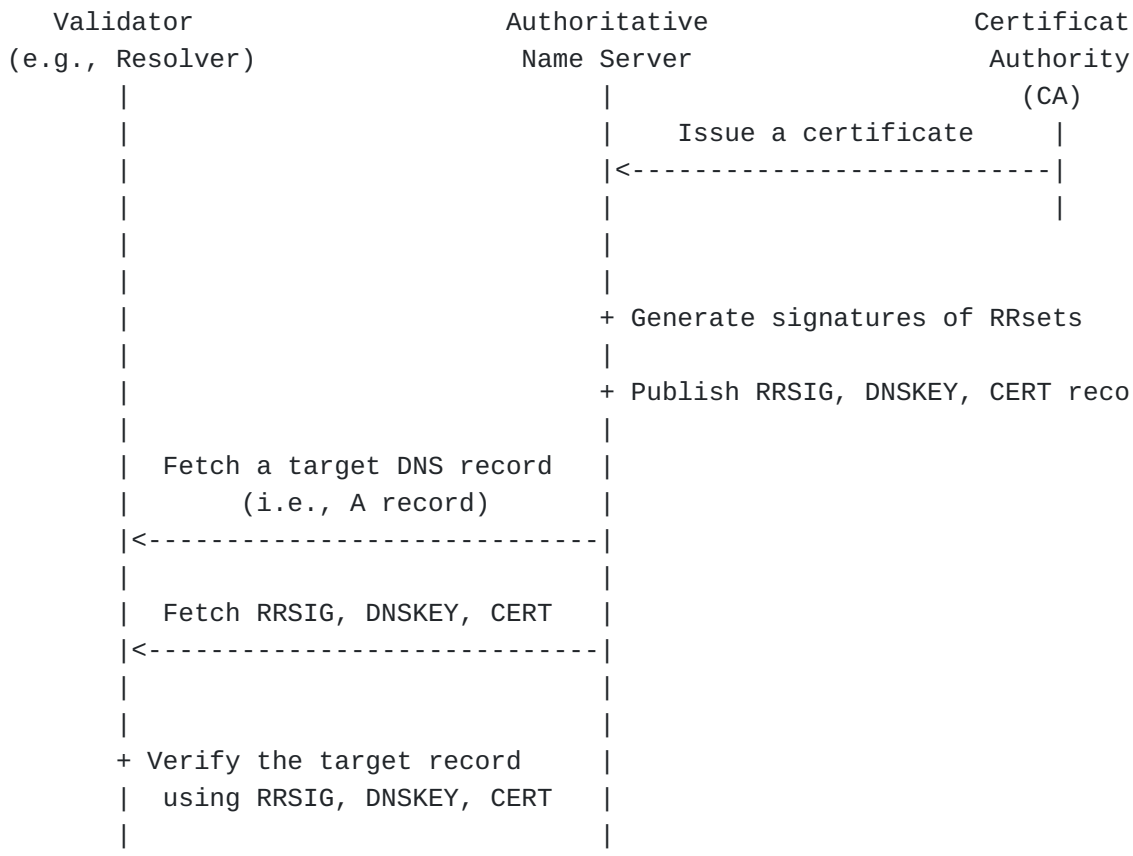

Figure 2: The operations of DNSSEC-PKI

3. Authoritative DNS Server-side Deployment

Deploying DNSSEC-PKI on the authoritative DNS server (i.e., a domain or a zone) requires the following steps:

1. A domain is issued a PKIX certificate (from a CA).

2. The domain generates the signatures of its DNS RRsets using its private key in the certificate.

3. The domain publishes the signatures as RRSIG records.

4. The domain publishes the corresponding public key as a DNSKEY record.

5. Finally, the domain publishes the certificates (of its certificate chain) as CERT records.

3.1. Generating Signatures of RRsets

For each RRset, the domain generates its signature and publishes it as an RRSIG record.

3.1.1. PKIX Certificates

ost domains use PKIX [RFC5280] certificates for HTTPS (or TLS). Thus, they are familiar with how to handle such certificates (issuance, storage, and so on). We also rely on PKIX certificates issued by CAs for DNSSEC-PKI.

3.1.2. RRSIG Records

We use RRSIG records for digital signatures of RRsets, which are generated by a private key in a PKIX certificate. The description of the RRSIG RR format is specified in Section 3 of [RFC4034]. The Key Tag field should be calculated as explained in Appendix B of [RFC4034].

A validator can fetch and validate RRSIG records to check the integrity and authenticity of DNS RRs.

3.2. Publishing Public Keys and Certificate Chains

Like DNSSEC, DNSSEC-PKI uses the public key cryptography to generate signatures of DNS RRsets. Thus, a domain should publish the public key as a DNS record. Also, the domain should publish a certificate that contains the public key and its certificate chain to allow a DNS client to validate the certificate and the signature. For this purpose, we use two existing DNS resource record (RR) types in DNSSEC.

3.2.1. DNSKEY Records

A domain (or zone) generates the signatures of DNS RRsets using its private key. The corresponding public key is stored in a DNSKEY RR which is described in Section 2 of [RFC4034]. As specified in Section 2 of [RFC4034], a DNSKEY RR MUST have a signature algorithm and a public key.

Other fields except the Flags field have the same meaning as those specified in Section 2 of [RFC4034].

3.2.1.1. The Flags Field extension

[RFC4034] specifies the use of two bits in the Flags field. Bit 7 is used to specify the Zone Key flag. If bit 7 is set to 1, the DNSKEY RR contains a DNS zone key. Otherwise, the DNSKEY MUST NOT be used to verify RRSIGs. Thus, to use DNSSEC-PKI, bit 7 MUST be set to 1.

Bit 15 is used to specify the Secure Entry Point flag which is described in [RFC3757]. If bit 7 is set to 1, the DNSKEY RR contains a key-signing key (KSK). Otherwise, the DNSKEY RR contains a zone-signing key (ZSK). In DNSSEC-PKI, a private key corresponding to the

public key in the DNSKEY RR is used to sign a zone's RRsets, and hence its role is similar to that of a ZSK. Thus, for DNSSEC-PKI, bit 7 MUST be set to 0.

We propose to use bit 3 to specify whether the DNSSEC-PKI extension is supported flag (P). If bit 3 is set to 1, the public key in DNSKEY RR MUST be validated using the certificates in CERT RRs (specified in Section 3.2.2).

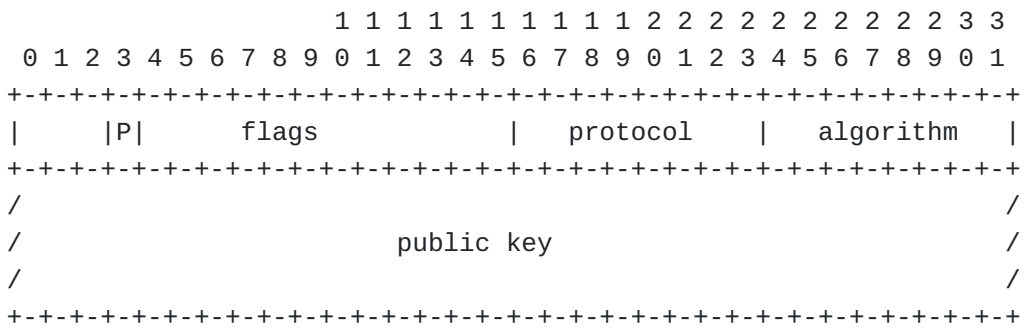

Figure 3: DNSSEC-PKI (P) Flag

3.2.2. CERT Records

A CERT RR [RFC4398] contains a PKIX certificate. The certificate includes a public key which is also included in a DNSKEY RR.

A Key Tag should be calculated using the public key in the certificate as specified in Appendix B of [RFC4034].

We assume that a DNS client already stores the certificate of a root CA. Thus, CERT RRs include the certificates of the domain, issuing CA and intermediate CAs (excluding the root CA).

4. Validator's Behavior

4.1. Verifying DNS RR

DNSSEC-PKI guarantees the integrity and authenticity of DNS RRs based on public key cryptography. Validators should verify the signatures of DNS RRsets generated by domains.

To check the integrity and authenticity of DNS RRs, a validator:

1. Fetches a target DNS RRset and its RRSIG, DNSKEY, and CERT RRs.

    *If there are cached DNS records whose TTLs are valid, the validator can use them directly.

    *The fetched DNS records can be cached.

2. Extracts a public key from the DNSKEY RR and another from a leaf certificate in the CERT RR.

   *If the extracted public keys are not matched, the verification of the signature fails.

   *Otherwise, it proceeds to the next step.

3. Validates the chain of certificates in the CERT RRs.

   *A certificate chain can be verified according to Section 6 of [RFC5280].

   *If the certificate chain verification fails, the verification of the signature fails.

4. Verifies the signature in the RRSIG RR using the public key.

4.2. Determine Verification Results

The signature verification will result in one of the following:

*Mismatch of public keys in DNSKEY and Certificate: A public key in a DNSKEY RR and another in a certificate are different.

*Certificate chain validation failure: The public keys in a DNSKEY RR and a certificate are matched but a certificate chain validation failed. (e.g., a certificate of a CA is expired).

*Signature validation failure: A chain of certificates is validated but the validation of RRSIGs fails (e.g., missing signatures or incorrect signatures).

*Authenticated: the signature is verified.

If a client trusts a (local or public) resolver, the resolver should validate the signature of DNS RRs and return the result to the client. A resolver sets the Authenticated Data (AD) bit in the message header of the DNS response message if and only if the queried RRset is authenticated.

If a client does not trust the resolver, it should verify the signature (RRSIG) by itself.

5. IANA Considerations

We ask to update the “DNSKEY FLAGS” registry [RFC3755][RFC4034] to assign 3rd bit in the DNSKEY Flags as the DNSSEC-PKI (P) bit.

Number	Description	Reference
3	DNSSEC-PKI (P)	Section 3.2.1.1 of this document

Table 1: DNSSEC-PKI (P) bit

6. Security Considerations

6.1. Authenticated Denial of Existence

We recommend using the NSEC [RFC4034] or NSEC3 [RFC5155] mechanism to provide the authenticated denial of existence.

6.2. Leveraging Certificate Authorities

DNSSEC-PKI leverages PKIX certificates issued by CAs. However, Certificate Authorities (CAs) have been criticized since there are no limits in a namespace in terms of certificate issuance. Protocols such as DNS CAA [RFC8659] and Certificates Transparency [RFC9162] can mitigate the problem.

7. References

7.1. Normative References

[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>.

[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>.

[RFC3755] Weiler, S., "Legacy Resolver Compatibility for Delegation Signer (DS)", RFC 3755, DOI 10.17487/RFC3755, May 2004, <https://www.rfc-editor.org/info/rfc3755>.

[RFC3757] Kolkman, O., Schlyter, J., and E. Lewis, "Domain Name System KEY (DNSKEY) Resource Record (RR) Secure Entry Point (SEP) Flag", RFC 3757, DOI 10.17487/RFC3757, April 2004, <https://www.rfc-editor.org/info/rfc3757>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <https://www.rfc-editor.org/info/rfc4033>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <https://www.rfc-editor.org/info/rfc4034>.

[RFC4398] Josefsson, S., "Storing Certificates in the Domain Name System (DNS)", RFC 4398, DOI 10.17487/RFC4398, March 2006, <https://www.rfc-editor.org/info/rfc4398>.

[RFC5155] Laurie, B., Sisson, G., Arends, R., and D. Blacka, "DNS Security (DNSSEC) Hashed Authenticated Denial of Existence", RFC 5155, DOI 10.17487/RFC5155, March 2008, <https://www.rfc-editor.org/info/rfc5155>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[RFC8659] Hallam-Baker, P., Stradling, R., and J. Hoffman-Andrews, "DNS Certification Authority Authorization (CAA) Resource Record", RFC 8659, DOI 10.17487/RFC8659, November 2019, <https://www.rfc-editor.org/info/rfc8659>.

[RFC9162] Laurie, B., Messeri, E., and R. Stradling, "Certificate Transparency Version 2.0", RFC 9162, DOI 10.17487/RFC9162, December 2021, <https://www.rfc-editor.org/info/rfc9162>.

7.2. Informative References

[DNSSEC-Deployment] Chung, T., van Rijswijk-Deij, R., Chandrasekaran, B., Choffnes, D., Levin, D., Maggs, B. M., Misloven, A., and C. Wilson, "A Longitudinal, End-to-End View of the DNSSEC Ecosystem", Proceedings of the 26th USENIX Security Symposium, Vancouver, August 2017, <https://www.usenix.org/system/files/conference/usenixsecurity17/sec17-chung.pdf>.

Authors' Addresses

Hyeonmin Lee
Seoul National University
Korea, Republic of

Email: min0921110@snu.ac.kr

Taekyoung Kwon
Seoul National University
Korea, Republic of

Email: tkkwon@snu.ac.kr