Workgroup: Network Working Group
Internet-Draft:
draft-ietf-dnsop-rfc5933-bis-09
Obsoletes: 5933 (if approved)
Updates: 8624 (if approved)
Published: 28 July 2022
Intended Status: Informational
Expires: 29 January 2023
Authors: D. Belyavskiy V. Dolmatov, Ed.
TCINET JSC "NPK Kryptonite"
B. Makarenko, Ed.
The Technical center of Internet, LLC

Use of GOST 2012 Signature Algorithms in DNSKEY and RRSIG Resource Records for DNSSEC

Abstract

This document describes how to produce digital signatures and hash functions using the GOST R 34.10-2012 and GOST R 34.11-2012 algorithms for DNSKEY, RRSIG, and DS resource records, for use in the Domain Name System Security Extensions (DNSSEC).

This document obsoletes RFC 5933 and updates RFC 8624.

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 29 January 2023.

Table of Contents

1. Introduction

The Domain Name System (DNS) is the global hierarchical distributed database for Internet Naming. The DNS has been extended to use cryptographic keys and digital signatures for the verification of the authenticity and integrity of its data. RFC 4033 [RFC4033], RFC 4034 [RFC4034], and RFC 4035 [RFC4035] describe these DNS Security Extensions, called DNSSEC.

RFC 4034 describes how to store DNSKEY and RRSIG resource records, and specifies a list of cryptographic algorithms to use. This document extends that list with the signature and hash algorithms GOST R 34.10-2012 ([RFC7091]) and GOST R 34.11-2012 ([RFC6986]), and specifies how to store DNSKEY data and how to produce RRSIG resource records with these algorithms.

This document obsoletes RFC5933 [RFC5933]. This document also marks the DNS Security Algorithm GOST R 34.10-2001 as obsolete.

Familiarity with DNSSEC and with GOST signature and hash algorithms is assumed in this document.

1.1. 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.

2. DNSKEY Resource Records

The format of the DNSKEY RR can be found in RFC 4034 [RFC4034].

GOST R 34.10-2012 public keys are stored with the algorithm number TBA1.

According to RFC 7091 [RFC7091], a public key is a point on the elliptic curve Q = (x,y). The wire representation of a public key MUST contain 64 octets, where the first 32 octets contain the little-endian representation of x and the second 32 octets contain the little-endian representation of y.

As RFC 6986 and RFC 7091 allows 2 variants of length of the output hash and signature and many variants of parameters of the digital signature, for the purpose of this document we use 256-bit variant of the digital signature algorithm, corresponding 256-bit variant of the digest algorithm. We select the parameters for the digital signature algorithm to be id-tc26-gost-3410-2012-256-paramSetA in RFC 7836 [RFC7836].

2.1. Using a Public Key with Existing Cryptographic Libraries

At the time of this writing, existing GOST-aware cryptographic libraries are capable of reading GOST public keys via a generic X509 API if the key is encoded according to RFC 7091 [RFC7091], Section 2.3.2.

To make this encoding from the wire format of a GOST public key with the parameters used in this document, prepend the 64 octets of key data with the following 32-byte sequence:

```
0x30 0x5e 0x30 0x17 0x06 0x08 0x2a 0x85 0x03 0x07 0x01 0x01 0x01
0x01 0x30 0x0b 0x06 0x09 0x2a 0x85 0x03 0x07 0x01 0x02 0x01 0x01
0x01 0x03 0x43 0x00 0x04 0x40
```

These bytes provide the following ASN.1 structure suitable for parsing by cryptographic toolkits:

```
  0  62: SEQUENCE {
  2   1:   INTEGER 0
  5  23:   SEQUENCE {
  7   8:     OBJECT IDENTIFIER '1 2 643 7 1 1 1 1'
 17  11:     SEQUENCE {
 19   9:       OBJECT IDENTIFIER '1 2 643 7 1 2 1 1 1'
       :       }
       :     }
 30  32:   OCTET STRING
```

The OIDs in the structure above represent GOsudarstvennyy STandart (GOST) R 34.10-2012 public keys with 256 bits private key length algorithm with Parameter set A for Keyed-Hash Message Authentication Code (HMAC) transformation based on the GOsudarstvennyy STandart (GOST) R 34.11-2012 hash function with 256-bit output according to RFC 7836 [RFC7836] and RFC 9125 [RFC9125].

2.2. GOST DNSKEY RR Example

Given a private key with the following value (the value of the Gost12Asn1 field is split here into two lines to simplify reading; in the private key file, it must be in one line):

```
Private-key-format: v1.2
Algorithm: 23 (ECC-GOST12)
Gost12Asn1: MD4CAQAwFwYIKoUDBwEBAQEwCwYJKoUDBwECAQEBBCA0
            zvTDpCSjdRCERkd6WDA2TF/ABQLp9MPZRl7hMXCVGg==
```

The following DNSKEY RR stores a DNS zone key for example:

```
example.  600  IN  DNSKEY  256 3 23 (
            XkZ6T+qQ9teOMsA/YK+kTzELhuMPTsYggdy2b+sfzJ6t
            H9eniziMX3gjMnUZIyrnSIchLjup8xpy+UU5l1Eyjw==
    ) ;{id = 13439 (zsk), size = 512b}
```

Public key can be calculated from the private key using algorithm described in RFC 7091 [RFC7091].

3. RRSIG Resource Records

The value of the signature field in the RRSIG RR follows RFC 7091 [RFC7091] and is calculated as follows. The values for the RDATA fields that precede the signature data are specified in RFC 4034 [RFC4034].

```
hash = GOSTR3411-2012(data)
```

where "data" is the wire format data of the resource record set that is signed, as specified in RFC 4034 [RFC4034].

The signature is calculated from the hash according to the GOST R 34.10-2012 standard, and its wire format is compatible with RFC 7091 [RFC7091].

3.1. RRSIG RR Example

With the private key from this document, consisting of one MX record:

```
example. 600 IN MX 10 mail.example.
```

Setting the inception date to 2020-01-04 17:25:26 UTC and the expiration date to 2020-02-01 17:25:26 UTC, the following signature RR will be valid:

```
example.  600 IN  RRSIG MX 23 1 600 20200201172526 (
                     20200104172526 13439 example.
                     EtrsAEGsNRf12HKjwNTg8U2HZ5JOSo34UaTcshoE1kwd
                     5Ror4I7zltmWAgd4b9OBn80tsajtL0Vuf45u8kEAgA==
)
```

Note: The ECC-GOST12 signature algorithm uses random data, so the actual computed signature value will differ between signature calculations.

4. DS Resource Records

The GOST R 34.11-2012 digest algorithm is denoted in DS RRs by the digest type TBA2. The wire format of a digest value is compatible with RFC 6986 [RFC6986].

4.1. DS RR Example

For Key Signing Key (KSK):

```
example.  IN  DNSKEY  257 3 23 (
                      hP3ISWPT8ehEEut8ozbqPcmbTAQK0jce7MHmK0geOiRo
                      kFALGwsMrBf0H0AK2qrVJCWCJL+50v9UNZAS5mE70g==
                      ) ;{id = 7574 (ksk), size = 512b}
```

The DS RR will be:

```
example.  IN  DS  7574 23 5 (
                     990f40dc548a4dbcb4b80a0760f194ac
                     0cc18484578834c1ac1f749f70c84103
                     )
```

5. Deployment Considerations

5.1. Key Sizes

The key size of GOST public keys conforming to this specification MUST be 512 bits according to RFC 7091 [RFC7091].

5.2. Signature Sizes

The size of a GOST signature conforming to this specification MUST be 512 bits according to RFC 7091 [RFC7091].

5.3. Digest Sizes

The size of a GOST digest conforming to this specification MUST be 256 bits according to RFC 6986 [RFC6986].

6. Implementation Considerations

The support of this cryptographic suite in DNSSEC-aware systems is OPTIONAL. Systems that do not support these algorithms may ignore the RRSIG, DNSKEY and DS records created with them.

(To be removed in RFC). To check the correctness of the implementation, authors recommend using OpenSSL 1.1.1 or 3.0.x series, a fork of ldns available at https://github.com/beldmit/ldns, and a reference implementation of GOST crypto algorithms available at https://github.com/gost-engine/engine.

7. Changes to RFC 5933

This document specifies the usage of the signature algorithm GOST R 34.10-2012 and hash algorithm GOST R 34.11-2012 instead of the signature algorithm GOST R 34.10-2001 and hash algorithm GOST R 34.11-94, specified in RFC 5933.

8. Update to RFC 8624

This document updates RFC8624 [RFC8624]. The paragraph describing the state of GOST algorithms in section 3.1 of RFC 8624 currently says:

ECC-GOST (GOST R 34.10-2001) has been superseded by GOST R 34.10-2012 in [RFC7091]. GOST R 34.10-2012 hasn't been standardized for use in DNSSEC.

That paragraph is now replaced with the following:

ECC-GOST (GOST R 34.10-2001) has been superseded by GOST R 34.10-2012 in [RFC7091]. GOST R 34.10-2012 has been standardized for use in DNSSEC in RFC TBC.

9. Security Considerations

Currently, the cryptographic resistance of the GOST R 34.10-2012 digital signature algorithm is estimated as 2**128 operations of multiple elliptic curve point computations on prime modulus of order 2**256.

Currently, the cryptographic collision resistance of the GOST R 34.11-2012 hash algorithm is estimated as 2**128 operations of computations of a step hash function.

10. IANA Considerations

This document updates the IANA registry "DNS Security Algorithm Numbers". The following entries have been added to the registry:

Value	Algorithm	Mnemonic	Zone Signing	Trans. Sec.	References	Status
TBA1	GOST R 34.10-2012	ECC-GOST12	Y	*	RFC TBA	OPTIONAL

The entry for the Algorithm "GOST R 34.10-2001", number 12 should be updated as such: Description field should be changed to "GOST R 34.10-2001 (deprecated, see TBA1" and Zone Signing field should be changed to "N".

This document updates the RFC IANA registry "Delegation Signer (DS) Resource Record (RR) Type Digest Algorithms" by adding an entry for the GOST R 34.11-2012 algorithm:

Value	Algorithm	Status
TBA2	GOST R 34.11-2012	OPTIONAL

The entry for Value 3, GOST R 34.11-94 should be updated to have its Status changed to '-'.

This paragraph shoud be removed before the publication of RFC: For the purpose of example computations, the following values were used: TBA1 = 23, TBA2 = 5.

11. Acknowledgments

This document is a minor extension to RFC 4034 [RFC4034]. Also, we tried to follow the documents RFC 3110 [RFC3110], RFC 4509 [RFC4509], and RFC 5933 [RFC5933] for consistency. The authors of and contributors to these documents are gratefully acknowledged for their hard work.

The following people provided additional feedback, text, and valuable assistance: Alexander Venedyukhin, Michael StJohns, Valery Smyslov, Tim Wicinski, Stephane Bortzmeyer.

12. References

12.1. Normative References

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

[RFC3110] Eastlake 3rd, D., "RSA/SHA-1 SIGs and RSA KEYs in the Domain Name System (DNS)", DOI 10.17487/RFC3110, RFC 3110, May 2001, <https://www.rfc-editor.org/info/rfc3110>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", DOI 10.17487/RFC4033, RFC 4033, 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>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", DOI 10.17487/RFC4035, RFC 4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC6986] Dolmatov, V., Ed. and A. Degtyarev, "GOST R 34.11-2012: Hash Function", RFC 6986, DOI 10.17487/RFC6986, August 2013, <https://www.rfc-editor.org/info/rfc6986>.

[RFC7091] Dolmatov, V., Ed. and A. Degtyarev, "GOST R 34.10-2012: Digital Signature Algorithm", RFC 7091, DOI 10.17487/RFC7091, December 2013, <https://www.rfc-editor.org/info/rfc7091>.

[RFC7836] Smyshlyaev, S., Ed., Alekseev, E., Oshkin, I., Popov, V., Leontiev, S., Podobaev, V., and D. Belyavsky, "Guidelines on the Cryptographic Algorithms to Accompany the Usage of Standards GOST R 34.10-2012 and GOST R 34.11-2012", DOI

10.17487/RFC7836, RFC 7836, March 2016, <https://www.rfc-editor.org/info/rfc7836>.

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

[RFC8624] Wouters, P. and O. Sury, "Algorithm Implementation Requirements and Usage Guidance for DNSSEC", RFC 8624, DOI 10.17487/RFC8624, June 2019, <https://www.rfc-editor.org/info/rfc8624>.

12.2. Informative References

[RFC4509] Hardaker, W., "Use of SHA-256 in DNSSEC Delegation Signer (DS) Resource Records (RRs)", RFC 4509, DOI 10.17487/RFC4509, May 2006, <https://www.rfc-editor.org/info/rfc4509>.

[RFC5933] Dolmatov, V., Ed., Chuprina, A., and I. Ustinov, "Use of GOST Signature Algorithms in DNSKEY and RRSIG Resource Records for DNSSEC", DOI 10.17487/RFC5933, RFC 5933, July 2010, <https://www.rfc-editor.org/info/rfc5933>.

[RFC9125] Farrel, A., Drake, J., Rosen, E., Patel, K., and L. Jalil, "Gateway Auto-Discovery and Route Advertisement for Site Interconnection Using Segment Routing", DOI 10.17487/RFC9125, RFC 9125, August 2021, <https://www.rfc-editor.org/info/rfc9125>.

Authors' Addresses

Dmitry Belyavskiy
TCINET
8 marta st
Moscow
Russian Federation

Phone: +7 916 262 5593
Email: beldmit@gmail.com

Vasily Dolmatov (editor)
JSC "NPK Kryptonite"
Spartakovskaya sq., 14, bld 2, JSC "NPK Kryptonite"
Moscow
105082
Russian Federation

Email: vdolmatov@gmail.com

Boris Makarenko (editor)
The Technical center of Internet, LLC
8 marta str., 1, bld 12
Moscow
127083
Russian Federation

Email: bmakarenko@tcinet.ru