

The future of DNSSEC cryptographic recommendations

AKA draft-hardaker-dnsop-rfc8624-~~bis~~

replacement

Wes Hardaker
Warren Kumari

Implementation recommendations today: RFC8624

- All **algorithm definitions** published in **various documents**
- Added to the various IANA tables
- Occasionally **the RFC with the recommendations** gets updated

3. Algorithm Selection

2024-07-rfc8624-bis ☆ 📁 ☁

File Edit View Insert Format Slide Arrange Tools Extensions Help

3.1. DNSKEY Algorithms

The following table lists the implementation recommendations for DNSKEY algorithms [DNSKEY-IANA].

Number	Mnemonics	Recommendations	DNSSEC Signing	DNSSEC4 Validation
1	RSAMD5	• All algorithm definitions published in various documents	MUST NOT	MUST NOT
3	DSA	• Added to the various IANA tables	MUST NOT	MUST NOT
5	RSASHA1	• Occasionally the RFC with the recommendations gets updated	NOT RECOMMENDED	MUST NOT

Problems with the current approach

- Updating all the recommendations is a heavy lift
 - Updating requires discussing ALL the recommendations
 - Thus: consensus is harder
- The IANA table can be out of sync with the recommendations RFC

Proposed New Approach

1. Move the recommendations **into the IANA tables**
 - draft-hardaker-dnsop-rfc8624-bis
 - This will **not change the recommendation levels**
2. New Goal: small documents updating the IANA tables as needed
 - Enable easy changes via narrow-scoped text
 - Discuss each change individually
3. Examples: Two short deprecation documents:
 - draft-hardaker-dnsop-must-not-sha1
 - draft-hardaker-dnsop-must-not(-ece)-gost

draft-hardaker-dnsop-rfc8624-bis

Adds 3 columns to existing tables:

IANA Table	Column added
Domain Security Algorithm Numbers	Recommended for DNSSEC Signing
Domain Security Algorithm Numbers	Recommended for DNSSEC Validation
Digest Algorithms	Recommended

An issue worth discussing today

Today's IANA change requirements:

Adding **new algorithms** to the IANA tables requires: RFC Required

Updating the **recommendations** requires: STD Action

Options going forward:

1. Require **all changes** to be **STD Action**
2. Or.... dual-level changes:
 - a. **RFC Required can add rows with MAYs** for recommended fields
 - b. **STD Action needed for any other changes**

Example acceptable modifications

RFC Type	Task	Previous Value	New Value
Non STD-Action (ISE, etc)	Add algorithm	N/A	MAY
STD Action	Add algorithm	N/A	*ANY*
STD Action	Modify upward	MAY/SHOULD/...	SHOULD/MUST/...
STD Action	Modify downward	MAY/SHOULD/MUST	MUST NOT, etc

Guidance for Implementers ./ Guidance for Signers

[draft-huque-dnsop-multi-alg-rules](#) also suggests IANA registry. Authors observe:

- RFC / registry currently provides guidance targeted at **implementations**
 - Implementation recommendations are **prescriptive** (“MUST”)
- **Signers** need to pick algorithm, based on actual validation support in the wild
 - Documenting when basically all validators support an algorithm is **descriptive** (“DOES”)

Are “prescriptive MUST” and “descriptive DOES” the same?

- If not: Add column “Universal Support”
- Initial value: empty; RFC publication required to change to “yes” / “formerly”

Use cases: document algos suitable for root KSK rollover / DNSSEC multi-signer

Adoptions?

We would like all three documents to be considered for WG adoption:

- draft-hardaker-dnsop-rfc8624-bis
- draft-hardaker-dnsop-must-not-sha1
- draft-hardaker-dnsop-must-not-gost