Nameserver Predelegation Check - Annotated

Table of Contents

DE

1. Introduction

1.1. Background

This documentation specifies the requirements across DNS related nameservers and their managed zones. These requirements must be fulfilled in order to guarantee that an existing DENIC domain can be delegated properly. Therefore, the following sections describe nameserver policies as well as the corresponding predelegation check requirements.

1.2. Theories of Motivation

The Domain Name System (DNS) represents a hierarchical, distributed and highly available database used for any kind of IP address translations within IT networking and infrastructure sector. Therefore, DNS enforces high redundancy and fault resilience. However, due to faulty networking or human miss-configuration invalid domain resolving still might occur as described in RFC46971. In order to guarantee stable zone delegation on any level, TLD administration defines requirements and criteria that must or should be fulfilled within proper DNS operation. Based on that, this document describes a pre-staged procedure named as *Predelegation Check* which is used to validate the compliance in zone delegation of a corresponding user domain effectively and securely.

2. General Checks

2.1. Nameserver Policy

Different issue types (i.e. warning, error) are used to describe violations within check execution. Therefore each check requirement represents a criterion that **MUST** or **SHOULD** be fulfilled. According to this, the relation between check requirement and issues type is defined as follows:

WARNING: This issue covers the violation of a requirement that **SHOULD** be fulfilled. Any occurrence of this type is treated as a recommendation and non-critical. It will not affect the overall result of predelegation checks themselves.

ERROR: This issue covers the violation of a requirement that **MUST** be fulfilled. Hence, any occurrence of this type is treated as critical and results in failure of the overall predelegation check procedure.

2.1.1. Authoritative Nameservers Only

All corresponding nameservers of the requested domain **MUST** be addressable and configured in authoritative mode towards the domain's final delegated zone. Any violation results in **ERROR**.

ERROR:

Code	Message
116	SOA record response must be authoritative
133	Answer must be authoritative

Further explanation: Requesting authoritative nameservers only ensures the agreement towards the final zone delegation of the requested domain. This is necessary because nameservers can be operated by any uncontracted third party (i.e. non-members) of DENIC eG- Additional notes can be found in RFC1034 and RFC1035.

2.1.2. Redundant Connectivity

At least two different nameservers **MUST** be addressable via IPv4 or IPv6 for the requested domain. Furthermore, at least one given nameserver **MUST** be addressable via IPv4. Any violation results in corresponding **ERROR** below.

ERROR:

Code	Message
107	Insufficient diversity of nameserver' s IP addresses
125	Insufficient diversity of nameserver' s IPv4 addresses
127	Insufficient number of nameservers reachable
129	Invalid IPv4 or IPv6 address
132	Could not resolve any IP address for this nameserver

Further explanation: One nameserver differs completely from another by using different IPv4 and IPv6 addresses. Besides, all available IP addresses of every nameserver will be resolved and considered within predelegation check. According to RFC1035, each zone must be delegable by two-levelled redundancy setup by at least two different nameservers that can be addressed by distinguished IP addresses.

Example: Valid set of nameservers

Nameservers	IP addresses
ns1.nic.nast	172.31.1.1', 'fd00:10:10::1:1
ns2.nic.nast	'fd00:10:10::2:2

Example: Invalid set of nameservers

Nameservers	IP addresses
ns1.nic.nast	172.31.1.1', 'fd00:10:10::1:1
ns2.nic.nast	172.31.1.1', 'fd00:10:10::2:2

2.1.3. Glue Records

The predelegation check uses narrow glue policy. Hence, glue records need to be available in .de zone (i.e. 9.4.164.arpa) if the name of any corresponding nameserver is located within the delegated zone only.

Based on that the following requirements can be derived:

2.1.3.1. Nameserver in Zone

At least one IP address (i.e. IPv4 or IPv6) **MUST** be specified in the check request for any nameserver located within the delegated zone. Any violation results in corresponding **ERROR** below.

ERROR:

Code	Message
101	Missing glue record for the nameserver

Further explanation: Considering both addressing cases IPv4 and IPv6 at least one glue record is required.

2.1.3.2. Nameserver NOT in Zone

No IP address (i.e. IPv4 or IPv6) **SHOULD** be specified in the check request for any nameserver NOT located within delegated zone. Otherwise **WARNING**.

WARNING:

Code	Message
102	Provided glue records not applicable

Further explanation: The narrow glue policy is applied for .de as well as for 9.4.e164.arpa. Hence, glue records are only allowed for the limited case that the nameserver is located in the delegated zone. Any additionally provided IP addresses are dispensable. The warning shall point out possible input errors.

2.1.3.3. IP addresses and RRSet Consistency

For each specified IP address of any nameserver in the check request the corresponding A and AAAA RRSet **MUST** be retrievable in authoritative mode and match with the request's specified IP addresses. Otherwise **ERROR**.

ERROR:

Code	Message
106	Inconsistent set of nameserver IP addresses

Further explanation: Since glue records as well as authoritative data coexist in DNS both need to be accessible and consistent. Therefore, the retrieved IP addresses from DNS must match the request's origin IP addresses. Besides, this requirement ensures equivalence between glue records and the A and AAAA RRSet related data (e.g. missing IP addresses in glue records).

2.1.4. SOA Zone Data

According to the SOA record data fields the following value-based requirements are specified:

2.1.4.1. Refresh

The value **SHOULD** be in range of [3600,86400] seconds. Otherwise **WARNING**.

WARNING:

Code	Message
108	Refresh value out of range

Further explanation: This value specifies the refresh interval in data sync between master/slave nameservers. Lower rates will cause higher DNS traffic and load on corresponding systems. On the opposite, high rates will lead to more outdated data. Therefore, any violation results in warning because the setting is defined by the nameserver operators.

2.1.4.2. Retry

The value **SHOULD** be in range of [900,28800] seconds AND **SHOULD** be a fractional part between 1/8 and 1/3 of Refresh. Otherwise a corresponding **WARNING** will be issued.

WARNING:

Code	Message
109, 110	Retry value out of range

Further explanation: This value overrules the refresh value if SOA sync between nameservers fails and will continue until sync is achieved or expiry threshold is reached. Hence, the value needs to be lower than Refresh. Keep in mind that smaller values will increase load. Besides, the requirement ensures balance between Refresh and Retry and guarantees advantage in switchover between those.

2.1.4.3. Expire

The value **SHOULD** be in range of [604800,3600000] seconds. Otherwise **WARNING**.

WARNING:

Code	Message
111	Expire value out of range

Further explanation: This value defines the limit for failed syncs until a corresponding slave will stop delegation of the requested zone. Values less than one week lead to early loss of zone delegation and are marked as critical. Hence, a common value of 3600000 seconds (i.e. 1000 hours) seems to be good tradeoff between sync failure occurrence and trigger of further root cause investigation.

2.1.4.4. NegTTL

The value **SHOULD** be in range of [180,86400] seconds. Otherwise **WARNING**.

WARNING:

Code	Message
112	Minimum TTL out of range

Further explanation: This values specifies the lifetime of any invalid SOA record response. It represents the counterpart of the usual TTL according to RFC2308. Higher values barely reduce DNS traffic because of DNS caches. Values beneath the lower boundary (i.e. 180 seconds) will disable the functionality of NegTTL completely.

2.1.5. Additional Zone Data

2.1.5.1. NS RRSet Consistency

The NS RRSet of the delegated zone **MUST** match the request's nameserver list completely. Otherwise **ERROR**.

ERROR:

Code	Message
118	Inconsistent set of NS RRs

Further explanation: RFC1034 specifies consistency between authoritative nameservers of the delegating and delegated zone.

2.1.5.2. No CNAME RR

The delegated zone **MUST** be free of any CNAME RR. Otherwise **ERROR**.

ERROR:

Code	Message
115	SOA record response must be direct

Further explanation: CNAME RR must be unique on every node within the DNS tree. Hence, any further CNAME RR within the delegated zone violates this specification.

2.1.5.3. Referral Response Size

The referral response size **MUST** fit the max DNS UDP packet size of 512 Bytes (including large QNAMEs, all address entries and glue records). Otherwise **ERROR**.

ERROR:

Code	Message
104	Calculated referral response larger than allowed

Further explanation: All DENIC nameserver query responses cover a referral (i.e. link) towards the corresponding, next-level nameserver in the zone delegation hierarchy. Hence, this requirement is used to avoid high loads of TCP based fallback retries due to truncation of former UDP requests.

2.1.5.4. Primary Nameserver Consistency

The primary nameserver (i.e. MNAME RR) of the delegated zone **SHOULD** be consistent in the SOA RR of any related nameserver. Otherwise **WARNING**.

WARNING:

Code	Message
113	Primary Master (MNAME) inconsistent across SOA records

Further explanation: This requirement co-insures the consistency requirements of former section SOA Zone Data

2.1.6. Miscellaneous

Additional ungrouped requirements are summarized below:

2.1.6.1. IPv6

All IPv6 addresses of any nameserver **MUST** be located within the same global unicast address scope, allocated and routable. Any violation results in the corresponding **ERROR** below.

ERROR:

Code	Message
130	IPv6 address is not allocated
131	IPv6 address is not routable

Further explanation: IPv6 is restricted to different address scopes. In order to ensure common reachability of any nameserver via IPv6 just global scoped addresses are accepted.

2.1.6.2. Recursive Queries not Allowed

The execution of recursive DNS queries **SHOULD** not be allowed. Otherwise **WARNING**.

WARNING:

Code	Message
120	Recursive queries should not be allowed

Further explanation: Separation of authoritative and recursive nameservers is necessary on namespace level and due to security reasons.

2.1.6.3. TCP Reachability

Any nameserver in check request **SHOULD** be reachable via TCP connection. Otherwise the corresponding **WARNING** will be issued.

WARNING:

Code	Message
902	Timeout
908	Connection refused

Further explanation: Within RFC1034 and RFC1035 TCP based DNS requests are supported too, but should be used as a fallback approach towards prior failed UDP requests only. Hence, if a UDP request fails on first level (e.g. due to truncation etc.) a switchover to TCP can be possible as mentioned in RFC123.

3. DNSSEC Checks

3.1. Fundamentals

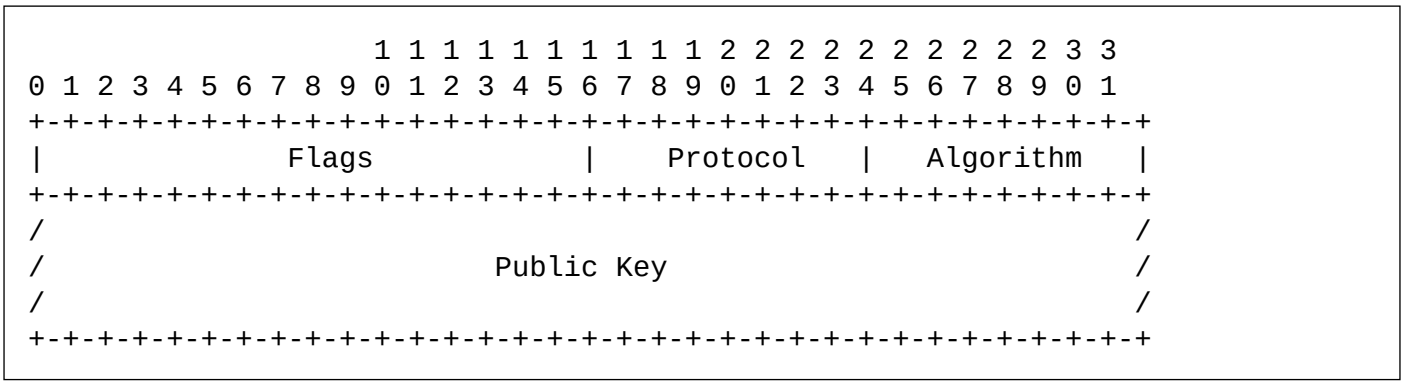
In order to perform DNSSEC based validations in zone delegations, additional security data (i.e. keys) need to be provided within the corresponding zones.

Therefore, the Key Signing Key (KSK) of the delegated zone reflects the major Secure Entry Point (SEP). This key is placed within the DNSKEY RRSet of the delegated zone, which is signed by the key, too. Besides, the key's fingerprint is placed as DS RR (Delegation Signer Resource Record) on higher delegation levels to avoid additional resource consumption.

All public-key-related data is provided as DNSKEY RR in wire-text format. The max number of possible keys in check requests is limited to 5. Within zone signing procedure all signatures and DS RRs are generated for each KSK automatically and are distributed with the delegated zone finally.

3.2. DNSKEY Resource Record

Each DNSKEY RR is provided in wire-text format as described in RFC4034 and shown below:



The "Flags" field covers the bits for DNSKEY related ZONE, REVOKE and SEP settings. The field "Algorithm" covers the keys cryptographic format. The "Public Key" part is kept as last field within RR.

3.3. DNSKEY Types & Signing

Introducing DS RR within RFC3658 recommends paired separation between the Zone Signing Key (ZSK) and the Key Signing Key (KSK). While a ZSK is used to sign any record data within zone (excluding DNSKEY RRSet), the KSK is used to authenticate ZSK by signing its corresponding DNSKEY RRSet. Hence, changing the ZSK implies less effort while changing the KSK leads to further changes on higher-level zones. Therefore, a KSK reflects larger keys than a ZSK to ensure longer periods of usage. This also means less consumption of zone data and finally smaller package sizes in DNS responses as explained in RFC4641. Hence, the separation of keys improves security, resource consumption as well as flexibility in key management but leads to a higher complexity in the DNS protocol, too. For that reason, key separation is not mandatory and usage of a single key instead of key pairs is possible in DNSSEC. Usually common use cases as ZSK+KSK and ZSK=KSK are feasible. Nevertheless, any key used for DNSSEC related zone signing must be covered within DNSKEY RRSet.

3.4. DNSKEY RR Visibility

The notion "visible DNSKEY RR" is defined as follows: A DNSKEY RR given in the check request is visible if it is included in the DNSKEY RRSet of the delegated zone.

3.5. Validation Concept

Within DNSSEC validation *Proof of Possession* and *Chain of Trust* must be satisfied and considered as follows.

Proof of Possession: The check request contains at least one public signing key (i.e. KSK) that validates the signature of the delegated zone's DNSKEY RRSet. This ensures integrity and authenticity of the DNSKEY RRSet itself.

Chain of Trust: The check request or the DNSKEY RRSet contains at least one public signing key (i.e. ZSK) that validates the signature of the delegated zone's SOA RR. Further trust verification on higher zone levels is not considered.

Based on this, corresponding criteria are defined in Requirements.

3.6. Requirements

As a first step in DNSSEC-related predelegation checks, the DNSKEY RRs of request are pre-checked according on the key format correctness (i.e. flags, algorithm, public key etc.). In the second step, further checks across zones and nameservers are carried out.

3.6.1. DNSKEY: Parameters

All DNSKEY RRs of the request **MUST** be distinct. Hence, the key's field parameter **MUST** be unique among all other keys. Besides, a maximum of 5 DNSKEY RR can be handed over in a request. Otherwise a corresponding **ERROR** is issued.

ERROR:

Code	Message
208	Duplicate DNSKEY RR
210	Max 5 DNSKEY RR allowed

3.6.2. DNSKEY: Flags

The "Flags" field is specified as single numeric value in range of [0,65535] and shall match the following requirements.

3.6.2.1. ZONE Bit

Bit 7 (ZONE) **MUST** be SET. Otherwise **ERROR**.

ERROR:

Code	Message
200	DNSKEY RR ZONE flag (bit 7) must be set

Further explanation: This requirement is specified in RFC4034.

3.6.2.2. REVOKE Bit

Bit 8 (REVOKE) **MUST** not be set. Otherwise **ERROR**.

ERROR:

Code	Message
201	DNSKEY RR REVOKE flag (bit 8) must not be set

Further explanation: Revoked keys cannot be used as trust anchors as described in RFC5011.

3.6.2.3. SEP Bit

Bit 15 (SEP) **SHOULD** be set. Otherwise a **WARNING** is returned.

WARNING:

Code	Message
202	DNSKEY RR SEP flag (bit 15) should be set

3.6.2.4. Final Values

According to the former requirements, a numeric value out of [256,257] **MUST** be chosen for ZONE Bit, REVOKE Bit and SEP Bit. All other values result in **ERROR**.

ERROR:

Code	Message
221	Unknown flags in DNSKEY RR are set

3.6.3. DNSKEY: Protocol

The "Protocol" field is considered as an immutable value of 3 as specified in RFC4034. Otherwise an **ERROR** is returned.

ERROR:

Code	Message
209	DNSKEY RR has invalid protocol

3.6.4. DNSKEY: Algorithm

The "Algorithm" field value **MUST** be chosen according to the following subset list of IANA Registry.

Supported algorithms: 3, 5, 6, 7, 8, 10, 12, 13 ,14

Any violation of this requirement results in **ERROR**.

ERROR:

Code	Message
220	DNSKEY RR has invalid algorithm

REMARK: Algorithms 3, 5, 7 and 12 are *deprecated* and future support will end in upcoming releases.

3.6.5. DNSKEY: Public Key

The public key field **MUST** cover the base64 encoded key value. Otherwise **ERROR**.

ERROR:

Code	Message
207	DNSKEY RR public key must be base64 encoded

Furthermore, the internal key format depends on the selected cryptographic algorithm and enforces specific requirements as shown below.

3.6.5.1. RSA

The RSA related algorithms 5,7,8 and 10 cover the requirements below.

3.6.5.1.1. Modulus

The modulu’s bit length **MUST** be in range of [512,4096]. Otherwise **ERROR**.

ERROR:

Code	Message
203	DNSKEY RR RSA key modulus length in bits out of range

3.6.5.1.2. Exponent

The exponent’s max bit length **MUST** be less than 128. Otherwise **ERROR**.

ERROR:

Code	Message
204	DNSKEY RR RSA public key exponent length in bits must not exceed 128 bits

Further explanation: The ranges for Modulus and Exponent are specified in RFC3110.

3.6.5.2. DSA

DSA based algorithms 3 and 6 consider the requirements below.

3.6.5.2.1. T Parameter

The T parameter **MUST** be in range of [0,8]. Otherwise **ERROR**.

ERROR:

Code	Message
205	DNSKEY RR DSA public key parameter T out of range

3.6.5.2.2. Length

The byte length must be equal to (213+T*24). Otherwise **ERROR**.

ERROR:

Code	Message
206	DNSKEY RR DSA public key has invalid size

3.6.5.3. ECDSA

The ECDSA algorithm 13 and 14 differ in key's bit length:

- In ECDSAP256SHA256 (13) the bit length **MUST** be 512. Otherwise **ERROR**.
- In ECDSAP384SHA384 (14) the bit length **MUST** be 768. Otherwise **ERROR**.

ERROR:

Code	Message
226	DNSKEY RR ECDSA public key has invalid size

Further explanation: All length parameter are specified in RFC6605.

3.6.5.4. GOST

The key bit length of algorithm 12 **MUST** be 512. Otherwise **ERROR**.

ERROR:

Code	Message
227	DNSKEY RR GOST public key has invalid size

Further explanation: All length parameter are specified in RFC5933.

3.6.6. DNSKEY RRSet

3.6.6.1. Status

The DNSKEY RRSet of the delegated zone **MUST** be identical on all authoritative nameservers. Otherwise **ERROR**.

ERROR:

Code	Message
211	Inconsistent DNSKEY RR in nameserver response

3.6.6.2. Visibility

At least one DNSKEY RR of request **MUST** be VISIBLE within the DNSKEY RRSet. Otherwise **ERROR**.

ERROR:

Code	Message
213	Did not find any DNSKEY RR from request in all nameserver responses

Besides, for any invisible DNSKEY RR of request a **WARNING** is returned.

WARNING:

Code	Message
212	Did not find DNSKEY RR from request in all nameserver responses

Further explanation: Additional DNSKEY RR in RRSet are neglected and accordance in DNSKEY RRSet signature is assumed but not tested explicitly. This allows online signing for DSA and ECDSA based algorithms.

3.6.7. Validation Proof of Possession

At least one visible DNSKEY RR of request **MUST** validate the signature of the DNSKEY RRSet. Otherwise **ERROR**.

ERROR:

Code	Message
216	No visible DNSKEY found signing the DNSKEY RR obtained in response

Further explanation: This requirement enforces named proof of possession of section Validation Concept.

3.6.8. Validation Chain of Trust

For the SOA RR of the delegated zone a valid chain of trust **MUST** exist. This means at least one visible DNSKEY RR of request or within the DNSKEY RRSet must validate the signature of the SOA RR. Otherwise **ERROR**.

ERROR:

Code	Message
217	No visible DNSKEY found in signing directly or indirectly the SOA RR obtained in response

Further explanation: This requirement enforces chain of trust towards the delegated zone and prevents security lameness. Trust validation is limited to delegated zone level to allow predelegation check for unregistered domains, too.

3.6.9. Cross Checks

According to the grouped DNSSEC requirements of the aforementioned sections further cross requirements can be derived.

3.6.9.1. EDNS0 Support

All authoritative nameservers **MUST** support the EDNS0 protocol. Hence, nameservers **MUST** respond with DNSSEC data (i.e. signatures) to DO-Bit signed queries. Otherwise **ERROR**.

ERROR:

Code	Message
218	Received invalid answer to a DO-Bit query

3.6.9.2. UDP related EDNS0

All authoritative nameservers **SHOULD** support UDP sufficiently according to the EDNS0 extended package size and connectivity. Otherwise a corresponding **WARNING** is returned.

WARNING:

Code	Message
214	Querying some authoritative nameservers via EDNS0 UDP failed

3.6.9.3. TCP connection reuse

All authoritative nameservers **SHOULD** support TCP connection reuse to lower connection setup costs according to RFC7766. Otherwise a corresponding **WARNING** is returned.

WARNING:

Code	Message
229	TCP connection reuse should be allowed

3.6.9.4. Availability of DNSKEY RRSet

The DNSKEY RRSet **MUST** be retrievable from DNS via TCP or UDP with attached DNSSEC signature data (EDNS0). Otherwise **ERROR**.

ERROR:

Code	Message
219	Unable to retrieve DNSKEY RR with TCP or EDNS0
902	Timeout
908	Connection refused

4. Glossary

4.1. Issues

Code	Severity	Message	Section ref.
101	ERROR	Missing glue record for the nameserver	Nameserver in Zone
102	WARNING	Provided glue records not applicable	Nameserver NOT in Zone
104	ERROR	Calculated referral response larger than allowed	Referral Response Size
106	ERROR	Inconsistent set of nameserver IP addresses	IP addresses and RRSet Consistency
107	ERROR	Insufficient diversity of nameserver's IP addresses	Redundant Connectivity
108	WARNING	Refresh value out of range	Refresh
109	WARNING	Retry value out of range	Retry
110	WARNING	Retry value out of range	Retry
111	WARNING	Expire value out of range	Expire
112	WARNING	Minimum TTL out of range	NegTTL

Code	Severity	Message	Section ref.
113	WARNING	Primary Master (MNAME) inconsistent across SOA records	Primary Nameserver Consistency
115	ERROR	SOA record response must be direct	No CNAME RR
116	ERROR	SOA record response must be authoritative	Authoritative Nameservers Only
118	ERROR	Inconsistent set of NS RRs	NS RRSet Consistency
120	WARNING	Recursive queries should not be allowed	Recursive Queries not Allowed
125	ERROR	Insufficient diversity of nameserver' s IPv4 addresses	Redundant Connectivity
127	ERROR	Insufficient number of nameservers reachable	Redundant Connectivity
129	ERROR	Invalid IPv4 or IPv6 address	Redundant Connectivity
130	ERROR	IPv6 address is not allocated	IPv6
131	ERROR	IPv6 address is not routable	IPv6
132	ERROR	Could not resolve any IP address for this nameserver	Redundant Connectivity
133	ERROR	Answer must be authoritative	Authoritative Nameservers Only
200	ERROR	DNSKEY RR ZONE flag (bit 7) must be set	ZONE Bit
201	ERROR	DNSKEY RR REVOKE flag (bit 8) must not be set	REVOKE Bit
202	WARNING	DNSKEY RR SEP flag (bit 15) should be set	SEP Bit
203	ERROR	DNSKEY RR RSA key modulus length in bits out of range	Modulus
204	ERROR	DNSKEY RR RSA public key exponent length in bits must not exceed 128 bits	Exponent
205	ERROR	DNSKEY RR DSA public key parameter T out of range	T Parameter
206	ERROR	DNSKEY RR DSA public key has invalid size	Length
207	ERROR	DNSKEY RR public key must be base64 encoded	DNSKEY: Public Key

Code	Severity	Message	Section ref.
208	ERROR	Duplicate DNSKEY RR	DNSKEY: Parameters
209	ERROR	DNSKEY RR has invalid protocol	DNSKEY: Protocol
210	ERROR	Max 5 DNSKEY RR allowed	DNSKEY: Parameters
211	ERROR	Inconsistent DNSKEY RR in nameserver response	Status
212	WARNING	Did not find DNSKEY RR from request in all nameserver responses	Visibility
213	ERROR	Did not find any DNSKEY RR from request in all nameserver responses	Visibility
214	WARNING	Querying some authoritative nameservers via EDNS0 UDP failed	UDP related EDNS0
216	ERROR	No visible DNSKEY found signing the DNSKEY RR obtained in response	Validation Proof of Possession
217	ERROR	No visible DNSKEY found in signing directly or indirectly the SOA RR obtained in response	Validation Chain of Trust
218	ERROR	Received invalid answer to a DO-Bit query	EDNS0 Support
219	ERROR	Unable to retrieve DNSKEY RR with TCP or EDNS0	Availability of DNSKEY RRSet
220	ERROR	DNSKEY RR has invalid algorithm	DNSKEY: Algorithm
221	ERROR	Unknown flags in DNSKEY RR are set	Final Values
227	ERROR	DNSKEY RR GOST public key has invalid size	GOST
228	ERROR	DNSKEY RR ED public key has invalid size	[req:dnskey-alg-eddsa]
229	WARNING	TCP connection reuse should be allowed	TCP connection reuse
901	ERROR	Unexpected RCODE	
902	ERROR	Timeout	TCP Reachability
903	ERROR	Timeout with recursive resolver	
904	ERROR	Port unreachable	

Code	Severity	Message	Section ref.
908	ERROR	Connection refused	TCP Reachability
909	ERROR	Host unreachable	
910	ERROR	Broken pipe	
911	ERROR	Connection aborted	
999	WARNING	Unexpected exception	

4.1.1. Issue Adaptions

4.1.1.1. Removed Issues

Code	Severity	Message	Section ref.	Remarks
103	WARNING	Nameservers having IPv6 glue records should have IPv4 glue records too (NS, # of IPv4 glues, # of IPv6 glues)" RECURSION_AVAILABLE ¬ "Recursive queries should not be allowed (resolver)	IP addresses and RRSet Consistency	Obsolete
114	ERROR	Inconsistent serial number across SOA records	IP addresses and RRSet Consistency	Obsolete
118	ERROR	NS query response is empty	NS RRSet Consistency	Obsolete
215	ERROR	Timeout after switching from UDP to TCP - switch to TCP due to truncation	Availability of DNSKEY RRSet	Obsolete
222	WARNING	Querying some authoritative nameservers via EDNS0 UDP causes timeout	UDP related EDNS0	Obsolete

Code	Severity	Message	Section ref.	Remarks
223	ERROR	Timeout after switching from UDP to TCP - switch to TCP due to timeout	Availability of DNSKEY RRSet	Obsolete
224	WARNING	Querying some authoritative nameservers via EDNS0 UDP causes unreachable	UDP related EDNS0	Obsolete
225	ERROR	Timeout after switching from UDP to TCP	Availability of DNSKEY RRSet	Obsolete
905	ERROR	Invalid DNSKEY RR public key - conversion problem		Obsolete
906	ERROR	Invalid DNSKEY RR DSA public key - conversion problem		Obsolete
907	ERROR	DNSKEY RR from nameserver response cannot be compared with DNSKEY RR from request - conversion problem		Obsolete

4.1.1.2. Changed Issues

Code	Severity	Message	Section ref.	Change
105	ERROR	All IPv6 Addresses must be Global Unicast dedicated, allocated and routable	IPv6	Split into new issues: 130, 131
107	ERROR	Insufficient diversity of nameserver's IPv4 addresses	Redundant Connectivity	Moved to new issue: 125
107	ERROR	Insufficient number of nameservers reachable via IPv4	Redundant Connectivity	Moved to new issue: 127
107	ERROR	Insufficient number of nameservers reachable	Redundant Connectivity	Moved to new issue: 127

Code	Severity	Message	Section ref.	Change
119	WARNING	Some nameservers not reachable via TCP	TCP Reachability	Split into new issues: 902, 908
121	WARNING	Received a truncated response	UDP related EDNS0	Moved to new issue: 214
209	ERROR	At least one DNSKEY RR must be specified in request	DNSKEY: Protocol	Message: DNSKEY RR has invalid protocol
212	WARNING	Did not find DNSKEY RR from request in nameserver response	Visibility	Message: Did not find DNSKEY RR from request in all nameserver responses
213	ERROR	No DNSKEY RR from request found in nameserver response	Visibility	Message: Did not find any DNSKEY RR from request in all nameserver responses
214	WARNING	Some nameservers not reachable via EDNS0 with sufficient packet size	UDP related EDNS0	Message: Querying some authoritative nameservers via EDNS0 UDP failed
908	ERROR	TCP Connection refused	TCP Reachability	Message: Connection refused
909	ERROR	Socket error		Message: Host unreachable

4.1.1.3. New Issues

Code	Severity	Message	Section ref.
133	ERROR	Answer must be authoritative	Authoritative Nameservers Only
228	ERROR	DNSKEY RR ED public key has invalid size	[req:dnskey-alg-eddsa]
910	ERROR	Broken pipe	
911	ERROR	Connection aborted	