DNS Operations (dnsop)
Internet-Draft
Intended status: Standards Track
Expires: 20 January 2026

Y.Q. Qiu
X.L. Li
Nankai University

July 2025

Clarifying DNS Resolver Behavior for the Recursion Desired (RD) Flag
draft-qiu-dnsop-rd-flag-clarification-00

Abstract

This document addresses inconsistencies observed in the handling of the Recursion Desired (RD) flag in DNS queries by various DNS resolver implementations, particularly when the RD flag is cleared (set to 0). Such inconsistencies have been shown to be exploitable, leading to potent Denial of Service (DoS) amplification attacks. This document provides clear guidance and recommendations for DNS resolver (including forwarding and recursive resolvers) behavior when processing queries with different RD flag settings. The goal is to enhance DNS security, mitigate specific amplification attack vectors, and ensure more predictable and robust DNS operations.

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 2 January 2026.

Copyright Notice

Table of Contents

1. Introduction

1.1. The Recursion Desired (RD) Flag

The Domain Name System (DNS) is a hierarchical and distributed naming system for computers, services, or other resources connected to the Internet or a private network. DNS queries contain a one-bit field known as the Recursion Desired (RD) flag. As defined in [RFC1034] and [RFC1035], this flag directs the name server receiving the query:

* If RD is set (1), and the queried name server supports recursive queries, it is directed to pursue the query recursively. That is, the name server should take responsibility for resolving the query and returning the final answer.

* If RD is cleared (0), and the name server does not support recursive queries, or recursion is not desired for this specific query, the name server should return an answer based on its own data (e.g., from its cache or authoritative zones) without

contacting other name servers. If the name server does not have the information locally, it should return a response indicating that, which might include referrals to other name servers.

1.2. Observed Vulnerabilities

Recent research, such as the "TsuKing" attack [XU2023], has highlighted that a significant number of DNS resolvers, including open resolvers and forwarding devices, do not strictly adhere to the intended semantics of the RD flag, particularly when it is cleared (RD=0). Specific problematic behaviors include:

* Resolvers performing recursive lookups or forwarding queries upstream even when the incoming query has RD=0.

* DNS forwarders modifying the RD flag, for example, by changing an incoming RD=0 to RD=1 in the query forwarded upstream.

These deviations from standard behavior create vulnerabilities that can be exploited for various purposes, most notably for constructing Denial of Service (DoS) amplification attacks. In such attacks, like the DNSCHAIN and DNSLOOP variants described in [XU2023], attackers leverage these incorrect RD flag handling behaviors to coordinate multiple resolvers into amplifying query traffic towards a victim. The vulnerability of performing recursion when RD=0 was identified as a key factor (termed "V1" in [XU2023]) enabling these attacks. The study also noted that some widely-used DNS software and public DNS services exhibited these behaviors under certain configurations.

1.3. Purpose of this Document

The purpose of this document is to:

* Reiterate and clarify the expected behavior of DNS resolvers (including recursive resolvers and DNS forwarders) concerning the RD flag.

* Provide explicit operational guidance to promote consistent implementation and deployment of RD flag handling.

* Mitigate the security risks associated with incorrect RD flag processing, thereby reducing the attack surface for certain types of DNS amplification attacks.

This document aims to update or clarify existing guidance in [RFC1034] and [RFC1035] by providing more explicit behavioral requirements for resolvers.

2. Conventions and Definitions

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 following terms:

DNS Resolver:
: A program that acts as a client to the DNS system, issuing queries and processing responses. Resolvers can be stub resolvers, recursive resolvers, or forwarders.

Recursive Resolver:
: A resolver that, upon receiving a query for which it does not have an answer, will itself query other name servers to find the answer.

DNS Forwarder:
: A resolver that, upon receiving a query it cannot answer from its cache, forwards the query to a configured upstream recursive resolver.

Stub Resolver:
: A minimal resolver that relies on an upstream recursive resolver to perform the bulk of the resolution work.

Authoritative Server:
: A name server that holds the definitive data for a particular zone.

Iterative Query:
: A query mode where a name server, if it does not have the answer, returns a referral to other name servers that are more authoritative for the queried name. The querier then issues queries to these referred servers.

Recursive Query:
: A query mode where the querier requests the name server to perform the full resolution process and return the final answer.

RD Flag:
: The "Recursion Desired" bit in the DNS query header.

3. Problem Description: Inconsistent RD Flag Handling

The core issue addressed by this document is the inconsistent and often incorrect handling of the RD flag by DNS resolvers, especially when RD=0. This inconsistency deviates from the original intent of the DNS protocol and introduces security vulnerabilities.

3.1. Recursive Behavior for RD=0 Queries

As highlighted in [XU2023], a notable number of DNS resolvers, including both recursive resolvers and forwarders, exhibit recursive behavior even when presented with a query where the RD flag is cleared (RD=0). Instead of responding solely from local data (cache or authoritative zones), these resolvers proceed to query other name servers or forward the query upstream, effectively ignoring the RD=0 directive. This behavior is problematic because:

* It violates the expectation of the querying client, which may have cleared the RD flag specifically to prevent recursion (e.g., for diagnostic purposes or to query a local cache).

* It allows these resolvers to be unwillingly drafted into attack architectures, as demonstrated by the DNSCHAIN and DNSLOOP attacks [XU2023], where chains or loops of resolvers are formed by exploiting this RD=0 recursion.

3.2. RD Flag Modification by Forwarders

Another problematic behavior observed, particularly in some DNS forwarding devices [XU2023] (e.g., certain configurations of RouterOS), is the modification of the RD flag. Specifically, a forwarder might receive a query with RD=0, but when it forwards this query to its upstream recursive resolver, it sets the RD flag to 1. This action fundamentally changes the nature of the query as perceived by the upstream resolver and contributes to the vulnerabilities described above. It forces recursion where none was requested by the original querier.

4. Recommended Behavior for RD Flag Processing

4.1. General Principles

All DNS resolvers (recursive resolvers, forwarders, and stub resolvers that might implement caching or forwarding logic) MUST inspect the RD flag in incoming queries. The decision to perform recursion, forward a query, or perform iterative queries MUST be influenced by the state of the RD flag in the incoming query and the resolver’s configured role and policy.

4.2. Queries with RD=1 (Recursion Desired)

This section describes behavior for queries where the RD flag is set (RD=1).

* Recursive Resolvers: If a resolver is configured to provide recursive service and receives a query with RD=1, it SHOULD attempt to resolve the query fully. This typically involves performing iterative queries to authoritative name servers as needed, or consulting its cache. If the resolver is not configured to provide recursive service to the querier, or if a local policy prevents recursion for the specific query or querier, it MAY return a response with RCODE=REFUSED.

* DNS Forwarders: If a DNS forwarder receives a query with RD=1 that it cannot answer from its local cache, it SHOULD forward the query to one of its configured upstream recursive resolvers. The forwarded query MUST also have the RD flag set to 1.

* Authoritative Servers: An authoritative server receiving a query with RD=1 MAY ignore the RD flag and respond based on its authoritative data for the zone. It is not obligated to perform recursion. If it is not authoritative for the queried name and does not support recursion, it typically returns a referral or an error.

4.3. Queries with RD=0 (Recursion Not Desired)

This section describes the REQUIRED behavior for queries where the RD flag is cleared (RD=0). This is the critical area for mitigating the vulnerabilities discussed.

4.3.1. Recursive Resolvers

When a recursive resolver (i.e., a resolver configured and capable of performing recursion, but not acting as a simple forwarder for this specific query) receives a query with RD=0:

1. The resolver MUST attempt to answer the query using only locally available information. Locally available information includes:

    * Its cache.

    * Local zone data if the resolver is also authoritative for the queried name or has been configured with local zone data.

2. The resolver MUST NOT forward the query to other recursive resolvers (unless it is acting as a designated forwarder as per Section 4.3.2, and even then, specific rules apply).

3. The resolver MUST NOT perform iterative queries to external authoritative name servers for this query.

4. If the answer is not available from locally available information:

   * If the name is known to not exist (e.g., from a cached NXDOMAIN or a negative cache entry compliant with [RFC2308]), the resolver SHOULD return a response with RCODE=NXDOMAIN.

   * Otherwise, the resolver SHOULD return a response with RCODE=0 (NoError) and an empty answer section.

   * Under certain policy conditions, it MAY return RCODE=REFUSED.

The intent is that an RD=0 query to a recursive resolver probes its local knowledge without causing external network activity for resolution.

4.3.2. DNS Forwarders

When a DNS forwarder receives a query with RD=0:

1. The forwarder SHOULD first attempt to answer the query from its local cache, if one exists and is consulted. If a satisfactory answer is found in the cache, it SHOULD be returned to the querier.

2. If the query cannot be answered from its local cache:

   * The forwarder MUST NOT forward the query to its configured upstream recursive resolver(s) with the RD flag set to 1. This directly addresses the problematic behavior of some forwarders modifying the RD flag from 0 to 1.

   * Ideally, a forwarder receiving an RD=0 query that misses its cache SHOULD behave like a recursive resolver as described in Section 4.3.1 (i.e., return an answer based purely on its local state, typically an empty NoError response or NXDOMAIN if known, and not forward the query at all). This is the RECOMMENDED behavior.

* If, due to a specific and explicit local configuration policy (e.g., a transparent proxying setup where the forwarder's role is strictly to pass queries to a specific upstream without local resolution capabilities beyond caching), a forwarder *does* forward a query that it originally received with RD=0, the forwarded query sent to the upstream resolver MUST also have the RD flag set to 0. The forwarder MUST NOT change the RD flag from 0 to 1.

The primary goal for forwarders handling RD=0 queries is to prevent them from initiating or propagating recursion that was not requested by the original querier.

4.3.3. Authoritative Servers

Authoritative name servers typically answer queries based on their zone data, regardless of the RD flag's setting. An authoritative server receiving a query with RD=0 SHOULD respond with data from its authoritative zones if the queried name falls within them. It MAY ignore the RD=0 flag in the sense that its primary function is to provide authoritative answers, not to perform recursion. If it is not authoritative for the queried name, it should respond as it normally would (e.g., with a referral or an error like NXDOMAIN if the domain exists but the name does not).

5. Security Considerations

The recommendations in this document are crucial for mitigating certain DNS-based amplification attacks, such as those described in [XU2023] (DNSCHAIN and DNSLOOP). Strict adherence to the specified handling of RD=0 queries by recursive resolvers and forwarders breaks the chains or loops that these attacks rely on, as resolvers will no longer perform or request recursion when explicitly told not to.

Incorrect handling of the RD=0 flag, specifically:

* Recursive resolvers performing iterative queries or forwarding when RD=0.

* Forwarders forwarding RD=0 queries as RD=1 queries.

can lead to resolvers being unwillingly co-opted into distributed amplification attacks. By following the normative requirements in Section 4.3, implementers and operators can significantly reduce the attack surface of the DNS infrastructure related to these vectors.

Failure to implement negative caching ([RFC2308]) appropriately can exacerbate issues when RD=0 queries are mishandled, as resolvers might repeatedly attempt to resolve unresolvable names. While not directly about RD flag handling, robust negative caching is a complementary mechanism for overall DNS health and can indirectly limit the impact of some misbehaviors. The TsuKing paper [XU2023] also identified "no negative caching" (V2) as a contributing factor in the effectiveness of their attacks.

Operators of DNS resolvers should ensure their software is compliant with the behaviors described herein and configure their resolvers according to these best practices.

6. IANA Considerations

This document makes no requests of IANA.

7. References

7.1. Normative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <https://www.rfc-editor.org/info/rfc1034>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[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>.

7.2. Informative References

[RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", RFC 2181, DOI 10.17487/RFC2181, July 1997, <https://www.rfc-editor.org/info/rfc2181>.

[RFC2308] Andrews, M., "Negative Caching of DNS Queries (DNS NCACHE)", RFC 2308, DOI 10.17487/RFC2308, March 1998, <https://www.rfc-editor.org/info/rfc2308>.

[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>.

[RFC5452] Hubert, A. and R. van Mook, "Measures for Making DNS More Resilient against Forged Answers", RFC 5452, DOI 10.17487/RFC5452, January 2009, <https://www.rfc-editor.org/info/rfc5452>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <https://www.rfc-editor.org/info/rfc6891>.

[RFC7873] Eastlake 3rd, D. and M. Andrews, "Domain Name System (DNS) Cookies", RFC 7873, DOI 10.17487/RFC7873, May 2016, <https://www.rfc-editor.org/info/rfc7873>.

[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>.

[AFEK2020] Afek, Y., Bremler-Barr, A., and L. Shafir, "NXNSAttack: Recursive DNS Inefficiencies and Vulnerabilities", In Proceedings of USENIX Security 2020, August 2020, <https://www.usenix.org/conference/usenixsecurity20/presentation/afek>.

[MOURA2021] Moura, G. C. M., Castro, S., Heidemann, J. S., and W. Hardaker, "TsuNAME: Exploiting Misconfiguration and Vulnerability to DDoS DNS", In Proceedings of the ACM Internet Measurement Conference (IMC ’21) , November 2021, <https://doi.org/10.1145/3487552.3487824>.

[ROSSOW2014] Rossow, C., "Amplification Hell: Revisiting Network Protocols for DDoS Abuse", In Proceedings of the Network and Distributed System Security Symposium (NDSS ’14) , February 2014, <https://www.ndss-symposium.org/ndss2014/programme/amplification-hell-revisiting-network-protocols-ddos-abuse/>.

[XU2023] Xu, W., Li, X., Lu, C., Liu, B., Duan, H., Zhang, J., Chen, J., and T. Wan, "TsuKing: Coordinating DNS Resolvers and Queries into Potent DoS Amplifiers", In Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS '23) , November 2023, <https://doi.org/10.1145/3576915.3616668>.

[ZHENG2020] Zheng, X., Lu, C., Peng, J., Yang, Q., Zhou, D., Liu, B., Man, K., Hao, S., Duan, H., and Z. Qian, "Poison Over Troubled Forwarders: A Cache Poisoning Attack Targeting DNS Forwarding Devices", In Proceedings of USENIX Security 2020, August 2020, <https://www.usenix.org/conference/usenixsecurity20/presentation/zheng-xiaofeng>.

Authors' Addresses

Yuqi Qiu
Nankai University
38 Tongyan Road
Tianjin
Tianjin, 300355
China
Email: norahqiu@163.com

Xiang Li
Nankai University
38 Tongyan Road
Tianjin
Tianjin, 300355
China
Email: lixiang@nankai.edu.cn