

DNSOP WG  
Internet-Draft  
Intended status: Standards Track  
Expires: December 6, 2021

T. Reddy  
McAfee  
N. Cook  
Open-Xchange  
D. Wing  
Citrix  
M. Boucadair  
Orange  
June 4, 2021

DNS Access Denied Error Page  
draft-reddy-dnsop-error-page-08

## Abstract

When a DNS server filters a query, the response to such query conveys no detailed explanation that elaborates why that query was blocked, leading thus to end-user confusion. A solution to this problem is needed in order to enhance the user experience.

This document defines a method to return an URI that explains the reason why a DNS query was filtered by a DNS server.

## 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 December 6, 2021.

## Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents

(<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">2</a>
<a href="#">2.</a>	Terminology . . . . .	<a href="#">6</a>
<a href="#">3.</a>	Error Page URI EDNS0 Option Format . . . . .	<a href="#">6</a>
<a href="#">4.</a>	Error Page URI Processing . . . . .	<a href="#">7</a>
<a href="#">4.1.</a>	Mitigating EDNS0 Forgery . . . . .	<a href="#">8</a>
<a href="#">5.</a>	Error Page . . . . .	<a href="#">9</a>
<a href="#">6.</a>	Usability Considerations . . . . .	<a href="#">10</a>
<a href="#">7.</a>	Security Considerations . . . . .	<a href="#">10</a>
<a href="#">8.</a>	IANA Considerations . . . . .	<a href="#">11</a>
<a href="#">8.1.</a>	A New Error Page URI EDNS Option . . . . .	<a href="#">11</a>
<a href="#">9.</a>	Acknowledgements . . . . .	<a href="#">11</a>
<a href="#">10.</a>	References . . . . .	<a href="#">11</a>
<a href="#">10.1.</a>	Normative References . . . . .	<a href="#">11</a>
<a href="#">10.2.</a>	Informative References . . . . .	<a href="#">13</a>
	Authors' Addresses . . . . .	<a href="#">14</a>

## [1.](#) Introduction

DNS filters are deployed for a variety of reasons, including endpoint security, parental filtering, and filtering required by law enforcement. Some of these reasons are discussed in more detail below:

- o Various network security services are provided by Enterprise networks to protect endpoints (e.g., Hosts including IoT devices). Network-based security solutions such as firewalls and Intrusion Prevention Systems (IPS) rely upon network traffic inspection to implement perimeter-based security policies. The network security services may, for example, prevent malware download, block known malicious domains, block phishing sites, etc.

These network security services act on DNS queries originating from endpoints. For example, DNS firewalls, a method of

expressing DNS response policy information inside specially constructed DNS zones, known as Response Policy Zones (RPZs) allows DNS servers to modify their DNS responses in real time in order to stop access to malware and phishing domains. Note that

some of the commonly known types of malware are viruses, worms, trojans, bots, ransomware, backdoors, spyware, and adware.

- o Network devices in a home network offer network security to protect the devices within the home network by performing DNS-based content filtering. The network security service may, for example, block access to specific domains to enforce parental control, block access to malware sites, etc.
- o Internet Service Providers (ISPs) typically block access to some DNS domains due to a requirement imposed by an external entity (e.g., Law Enforcement Agency). Such blocking is performed using DNS-based content filtering.

DNS responses can be filtered by sending a bogus (also called, "forged") A or AAAA response, NXDOMAIN error or empty answer, or an extended DNS error (EDE) code defined in [[RFC8914](#)]. Each of these methods have advantages and disadvantages that are discussed below:

1. The DNS response is forged to provide a list of IP addresses that points to an HTTP(S) server alerting the end user about the reason for blocking access to the requested domain (e.g., malware). When an HTTP(S) enabled domain name is blocked, the network security device (e.g., CPE, firewall) presents a block page instead of the HTTP response from the content provider hosting that domain. If an HTTP enabled domain name is blocked, the network security device intercepts the HTTP request and returns a block page over HTTP. If an HTTPS enabled domain is blocked, the block page is also served over HTTPS. In order to return a block page over HTTPS, man in the middle (MITM) is enabled on endpoints by generating a local root certificate and an accompanying (local) public/private key pair. The local root certificate is installed on the endpoint while the network security device(s) stores a copy of the private key. During the TLS handshake, the network security device modifies the certificate provided by the server and (re)signs it using the

private key from the local root certificate.

- \* However, configuring the local root certificate on endpoints is not a viable option in several deployments like home networks, schools, Small Office/Home Office (SOHO), and Small/Medium Enterprise (SME). In these cases, the typical behavior is that the forged DNS response directs the user towards a server hosted to display the block page which breaks the TLS connection. For web-browsing this then results in an HTTPS certificate error message indicating that a secure connection could not be established, which gives no information to the end-user about the reason for the error.

The typical errors are "The security certificate presented by this website was not issued by a trusted certificate authority" (Internet Explorer/Edge), "The site's security certificate is not trusted" (Chrome), "This Connection is Untrusted" (Firefox), "Safari can't verify the identity of the website..." (Safari on MacOS)".

- \* Enterprise networks do not assume that all the connected devices are managed by the IT team or Mobile Device Management (MDM) devices, especially in the quite common Bring Your Own Device (BYOD) scenario. In addition, the local root certificate cannot be installed on IoT devices without a device management tool.
- \* An end user does not know why the connection was reset and, consequently, may repeatedly try to reach the domain but with no success. Frustrated, the end user may switch to an alternate network that offers no DNS-level protection against malware and phishing, potentially compromising both security and privacy. Furthermore, certificate errors train users to click through certificate errors, which is a bad security practice. To eliminate the need for an end user to click through certificate errors, an end user may manually install a local root certificate on a host device (e.g. [\[Chrome-Install-Cert\]](#)). Doing so, however, is also a bad security practice as it creates a security vulnerability that may be exploited by a MITM attack. When a manually installed local root certificate expires, the user has to (again) manually install the new local root certificate.

2. The DNS response is forged to provide a NXDOMAIN response to cause the DNS lookup to terminate in failure. In this case, an end user does not know why the domain cannot be reached and may repeatedly try to reach the domain but with no success. Frustrated, the end user may use insecure connections to reach the domain, potentially compromising both security and privacy.
3. The extended error codes Blocked, Censored, and Filtered defined in [Section 4 of \[RFC8914\]](#) can be returned by a DNS server to provide additional information about the cause of a DNS error. If the extended error code "Forged Answer" defined in [Section 4.5 of \[RFC8914\]](#) is returned by the DNS server, the client can identify the DNS response is forged together with the reason for HTTPS certificate error.

These extended error codes do not suffer from the limitations discussed in bullets (1) and (2), but the user still does not know the exact reason nor he/she is aware of the exact entity

blocking the access to the domain. For example, a DNS server may block access to a domain based on the content category such as "Adult Content" to enforce parental control, "Violence & Terrorism" due to an external requirement imposed by an external entity (e.g., Law Enforcement Agency), etc. These content categories cannot be standardized because the classification of domains into content categories is vendor specific, typically ranges from 40 to 100 types of categories depending on the vendor and the categories keep evolving. Furthermore, the threat data used to categorize domains may sometimes misclassify domains (e.g., domains wrongly classified as Domain Generation Algorithm (DGA) by deep learning techniques, domain wrongly classified as phishing due to crowd sourcing, new domains not categorized by the threat data). A user needs to know the contact details of the IT/InfoSec team to raise a complaint.

4. The EXTRA-TEXT field of the EDE option defined in [Section 2 of \[RFC8914\]](#) can include additional textual information about the cause of the error, but the information could be provided in a language that is not understood by the user. When a resolver or forwarder forwards the received EDE option, the EXTRA-TEXT field only conveys the source of the error ([Section 3 of \[RFC8914\]](#)) and

does not provide additional textual information about the cause of the error. Most importantly, EDE option does not offer authenticated information; it can thus be spoofed by an attacker. In addition, the additional textual information may not be able to convey all of the required information about the cause of the DNS error because lengthy EXTRA-TEXT content would be truncated to prevent fragmentation ([Section 3 of \[RFC8914\]](#)).

No matter which type of response is generated (forged IP address(es), NXDOMAIN or empty answer, or an extended error code), the user who triggered the DNS query has little chance to understand which entity filtered the query, how to report a mistake in the filter, or why the entity filtered it at all. This document describes a mechanism to provide an URI which, when accessed, provides such information to the user.

One of the other benefits of this approach is to eliminate the need to "spoof" block pages for HTTPS resources. This is achieved as the block page no longer needs to create a signed certificate when blocking a destination. This approach avoids the need to install a local root certificate authority on those IT-managed devices.

## [2.](#) 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.

This document makes use of the terms defined in [\[RFC8499\]](#).

'Encrypted DNS' refers to any encrypted scheme to convey DNS messages, for example, DNS-over-HTTPS [\[RFC8484\]](#), DNS-over-TLS [\[RFC7858\]](#), or DNS-over-QUIC [\[I-D.ietf-dprive-dnsquic\]](#).

## [3.](#) Error Page URI EDNS0 Option Format

This document uses an EDNS0 [RFC6891] option to include the URI that provides additional information in a DNS response about the cause of blocking access to a requested domain. This option is structured as depicted in Figure 1.

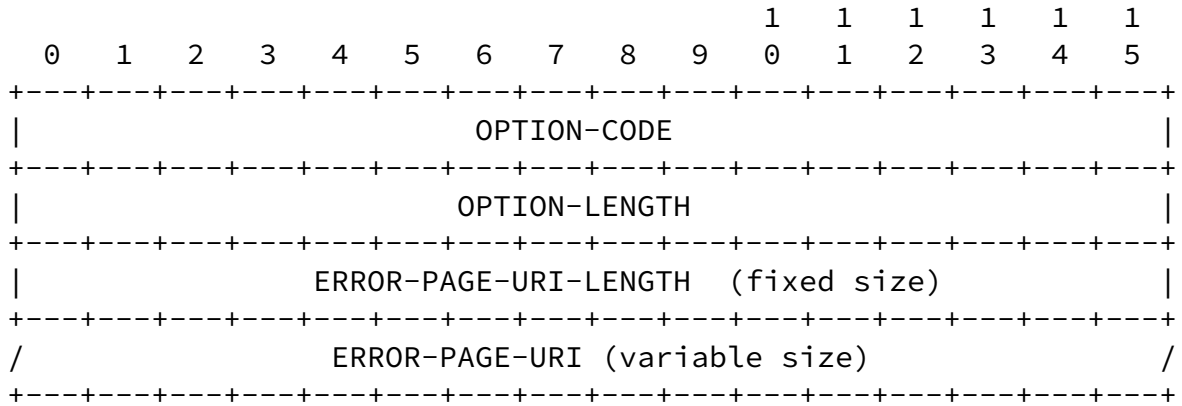


Figure 1: Error Page URI EDNS0 Option Format

The description of the fields is as follows:

- o OPTION-CODE: TBD, indicates the code assigned for Error Page URI (Section 6.1.2 of [RFC6891]). [RFC Editor: change TBD to the proper code once assigned by IANA.]
- o OPTION-LENGTH: See Section 6.1.2 of [RFC6891]. This field contains the length of the payload (everything after OPTION-LENGTH) in octets. The variability of the option length stems from the variable-length ERROR-PAGE-URI field.
- o ERROR-PAGE-URI-LENGTH: This 16-bit field indicates the length of ERROR-PAGE-URI. It MUST NOT be set to 0.

- o ERROR-PAGE-URI: A variable length UTF-8 encoded [RFC5198] text field containing the URI Template [RFC6570] that gives additional information about the cause of blocking access to a domain. The ERROR-PAGE-URI field MUST NOT be zero octets in length.

The Error Page URI option can be included in any response (SERVFAIL, NXDOMAIN, REFUSED, and even NOERROR, etc.) to a query that includes OPT Pseudo-RR [RFC6891].

The URI Template defined in ERROR-PAGE-URI describes how to construct the URL to fetch the error page. The agent acting as the HTTPS client on the endpoint encodes an FQDN to which access is denied into an HTTP GET request to retrieve the error page. The HTTPS server returning the error page defines the URI used by the HTTP GET request through the use of a URI Template. The URI Template is processed with a defined variable "target-domain" whose value is set to the FQDN to which access is denied.

The FQDN is provided as the variable value for "target-domain" to expand the URI Template into an URI reference in the HTTP GET request.

An example is illustrated below:

If the URI Template is "https://resolver.example.net/block-page{?target-domain}" for the HTTPS server returning the error page and access to the target domain "example.com" is blocked by the encrypted DNS server, the variable "target-domain" has the value "example.com" sent in an HTTP GET request. In the above example, the expansion of the above URI Template is "https://resolver.example.net/block-page?target-domain=example.com".

HTTP/2 [[RFC7540](#)] is the minimum RECOMMENDED HTTP version to use to retrieve the error page. The HTTPS client retrieving the error page MUST verify the entire certification path as per [[RFC5280](#)]. The HTTPS client additionally uses validation techniques described in [[RFC6125](#)] to compare the domain name in the error page URI to the server certificate provided in TLS handshake. See [[RFC7525](#)] for additional TLS recommendations.

#### 4. Error Page URI Processing

The DNS client MUST follow the rules below to process the Error Page URI EDNS0 option:

- o If the DNS response contains more than one Error Page URI EDNS0



option, the DNS client MUST discard all Error Page URI EDNS0 options in the DNS response.

- o The Error Page URI EDNS0 option MUST be processed by the DNS client for a "Censored", "Blocked", "Filtered" or "Forged" extended error codes and MUST be ignored for any other type of extended DNS error code. When "Censored", "Blocked", "Filtered" or "Forged" extended error code is returned in conjunction with an Error Page URI EDNS0 option, any other resource records in the answer MUST be ignored by clients supporting this specification.
- o The DNS client MUST reject the error page URI if the scheme is not "https".

#### 4.1. Mitigating EDNS0 Forgery

The Error Page URI EDNS0 option is susceptible to forgery. An attacker (e.g., a man in the middle (MITM)) could insert an extended Error Page URI EDNS0 option into the DNS response causing a client to attempt to visit that URI. For instance, the attacker can be located between the stub resolver and DNS recursive server or between the DNS proxy and the upstream resolver. To mitigate that attack, the following measures are enforced:

- o The DNS client MUST NOT process the DNS response with Error Page URI EDNS0 option unless DNS messages exchanged are cryptographically protected using encrypted DNS.
- o If a DNS client has enabled opportunistic privacy profile ([Section 5 of \[RFC8310\]](#)) for DoT, the DNS client will either fallback to an encrypted connection without authenticating the DNS server provided by the local network or fallback to clear text DNS, and cannot exchange encrypted DNS messages. Both of these fallback mechanisms adversely impacts security and privacy. If the DNS client has enabled opportunistic privacy profile for DoT, the DNS client MUST ignore Error Page URI EDNS0 option in responses, but SHOULD process other parts of the response.
- o If a DNS client has enabled strict privacy profile ([Section 5 of \[RFC8310\]](#)) for DoT, the DNS client requires an encrypted connection and successful authentication of the DNS server; this mitigates both passive eavesdropping and client redirection (at the expense of providing no DNS service if an encrypted, authenticated connection is not available). If the DNS client has enabled strict privacy profile for DoT, the client can process the DNS response with Error Page URI EDNS0 option. Note that the strict and opportunistic privacy profiles as defined in [\[RFC8310\]](#)

---

only applies to DoT protocol, there has been no such distinction made for DoH protocol.

- o If the DNS client determines that the encrypted DNS server does not offer DNS filtering service, it MUST reject the Error Page URI EDNS0 option. For example, the DNS client can learn whether the encrypted DNS resolver performs DNS-based content filtering or not by retrieving resolver information using the method defined in [[I-D.reddy-add-resolver-info](#)].
- o DNS forwarders (or DNS proxies) are supposed to propagate unknown EDNS0 options (Sections [4.1](#) and [4.4.1](#) of [[RFC5625](#)]), which means the Error Page URI EDNS0 option may get propagated by such a DNS server. To detect this scenario, the DNS client MUST verify the domain name in the Error Page URI matches the domain name of the encrypted DNS resolver. If this match fails, the DNS client MUST ignore Error Page URI EDNS0 option in the response, but SHOULD process other parts of the response.

## [5.](#) Error Page

The following outlines the RECOMMENDED contents of an error page to assist the operator developing the error page:

- o The exact reason for blocking access to the domain. If the domain is blocked based on some threat data, the threat type associated with the blocked domain can be provided/displayed to the end user. For example, the reason can indicate the type of malware blocked like spyware and the damage it can do the security and privacy of the user.
- o The domain name blocked.
- o If query was blocked by regulation, a pointer to a regulatory text that mandates this query block.
- o The entity (or organization) blocking the access to the domain and contact details of the IT/InfoSec team to raise a complaint.
- o The blocked error page to not include Ads and dynamic content.

The content of the error page discussed above is non-normative, the above text only provides the guidelines and template for the error page and:

- o does not attempt to offer an exhaustive list for the contents of

an error page.

- o it is not intended to form the basis of any legal/compliance for developing the error page.

## 6. Usability Considerations

The error page SHOULD be returned in the user's preferred language as expressed by the Accept-Language HTTP header.

## 7. Security Considerations

Security considerations in [Section 6 of \[RFC8914\]](#) and [\[RFC8624\]](#) need to be taken into consideration.

The Error Page URI EDNS0 option causes an HTTPS retrieval by the client. To prevent forgery of the Error Page URI EDNS0 option, this specification requires it only be sent only over an encrypted DNS channel with an authorized DNS server.

The client knows it is connecting to a HTTPS server returning the error page. To reduce threat surface the client can retrieve the Error Page URL using, for example, an isolated environment and take other precautions such as clearly labeling the page as untrusted or prevent user interaction with the page. Such isolation should prevent transmitting cookies, block JavaScript, block auto-fill of credentials or personal information, and be isolated from the user's normal environment.

Browsers perform some of the above restrictions when accessing captive portals ([Section 5 of \[RFC8910\]](#) or [\[Safari-Cookie\]](#)), during private browsing, or using containerization [\[Facebook-Container\]](#).

Note that the means to use a sandbox environment and a user interface presenting the error page are not covered in this document. By its nature, these aspects are implementation specific and best left to the application and user interface designers.

The encrypted DNS session provides transport security for the interaction between the DNS client and server, but DNSSEC signing and validation is not possible for the Error Page URI EDNS0 option

returning the Error Page URI Template. However, this specification mandates the DNS client to not process DNS response with Error Page URI EDNS0 option if domain name in the Error Page URI does not match the domain name of the encrypted DNS server. The validation ensures both the servers are operated by the same entity and have the same origin (similar to the Same Origin Policy (SOP)).

By design, the object referenced by the error page URL potentially exposes additional information about the DNS resolution process that

may leak information. An example of this is the reason for blocking the access to the domain name and the entity blocking access to the domain.

## [8.](#) IANA Considerations

### [8.1.](#) A New Error Page URI EDNS Option

This document defines a new EDNS(0) option, entitled "Error Page URI", assigned a value of TBD from the "DNS EDNS0 Option Codes (OPT)" registry [to be removed upon publication:

[<http://www.iana.org/assignments/dns-parameters/dns-parameters.xhtml#dns-parameters-11>]

Value	Name	Status	Reference
TBD	Error Page URI	Standard	[ This document ]

## [9.](#) Acknowledgements

Thanks to Vittorio Bertola, Wes Hardaker, Ben Schwartz, Erid Orth, Viktor Dukhovni, Warren Kumari and Bob Harold for the comments.

## [10.](#) References

### [10.1.](#) 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>>.

- [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", [RFC 4648](#), DOI 10.17487/RFC4648, October 2006, <<https://www.rfc-editor.org/info/rfc4648>>.
- [RFC5198] Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", [RFC 5198](#), DOI 10.17487/RFC5198, March 2008, <<https://www.rfc-editor.org/info/rfc5198>>.
- [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>>.

Reddy, et al.

Expires December 6, 2021

[Page 11]

---

Internet-Draft

DNS Access Denied Error Page

June 2021

- [RFC5625] Bellis, R., "DNS Proxy Implementation Guidelines", [BCP 152](#), [RFC 5625](#), DOI 10.17487/RFC5625, August 2009, <<https://www.rfc-editor.org/info/rfc5625>>.
- [RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", [RFC 6125](#), DOI 10.17487/RFC6125, March 2011, <<https://www.rfc-editor.org/info/rfc6125>>.
- [RFC6570] Gregorio, J., Fielding, R., Hadley, M., Nottingham, M., and D. Orchard, "URI Template", [RFC 6570](#), DOI 10.17487/RFC6570, March 2012, <<https://www.rfc-editor.org/info/rfc6570>>.
- [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>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", [BCP 195](#), [RFC 7525](#), DOI 10.17487/RFC7525, May

2015, <<https://www.rfc-editor.org/info/rfc7525>>.

- [RFC7540] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext Transfer Protocol Version 2 (HTTP/2)", [RFC 7540](#), DOI 10.17487/RFC7540, May 2015, <<https://www.rfc-editor.org/info/rfc7540>>.
- [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>>.
- [RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", [RFC 8310](#), DOI 10.17487/RFC8310, March 2018, <<https://www.rfc-editor.org/info/rfc8310>>.
- [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>>.

Reddy, et al.

Expires December 6, 2021

[Page 12]

---

Internet-Draft

DNS Access Denied Error Page

June 2021

- [RFC8914] Kumari, W., Hunt, E., Arends, R., Hardaker, W., and D. Lawrence, "Extended DNS Errors", [RFC 8914](#), DOI 10.17487/RFC8914, October 2020, <<https://www.rfc-editor.org/info/rfc8914>>.

## [10.2](#). Informative References

[Chrome-Install-Cert]

"How to manually install the Securlly SSL certificate in Chrome", <[support.securlly.com/hc/en-us/articles/206081828-How-to-manually-install-the-Securlly-SSL-certificate-in-Chrome](https://support.securlly.com/hc/en-us/articles/206081828-How-to-manually-install-the-Securlly-SSL-certificate-in-Chrome)>.

[Facebook-Container]

"Facebook container for Firefox", <<https://www.mozilla.org/en-US/firefox/facebookcontainer/>>.

- [I-D.ietf-dprive-dnsquic]  
Huitema, C., Mankin, A., and S. Dickinson, "Specification of DNS over Dedicated QUIC Connections", [draft-ietf-dprive-dnsquic-02](#) (work in progress), February 2021.
- [I-D.reddy-add-resolver-info]  
Reddy, T. and M. Boucadair, "DNS Resolver Information", [draft-reddy-add-resolver-info-03](#) (work in progress), April 2021.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", [RFC 7858](#), DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [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>>.
- [RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", [BCP 219](#), [RFC 8499](#), DOI 10.17487/RFC8499, January 2019, <<https://www.rfc-editor.org/info/rfc8499>>.
- [RFC8910] Kumari, W. and E. Kline, "Captive-Portal Identification in DHCP and Router Advertisements (RAs)", [RFC 8910](#), DOI 10.17487/RFC8910, September 2020, <<https://www.rfc-editor.org/info/rfc8910>>.

Reddy, et al.

Expires December 6, 2021

[Page 13]

---

Internet-Draft

DNS Access Denied Error Page

June 2021

[Safari-Cookie]  
"Isolated cookie store (CVE-2016-1730)",  
<<https://support.apple.com/en-us/HT205732>>.

#### Authors' Addresses

Tirumaleswar Reddy  
McAfee, Inc.  
Embassy Golf Link Business Park  
Bangalore, Karnataka 560071  
India

Email: kondtir@gmail.com

Neil Cook  
Open-Xchange  
UK

Email: neil.cook@noware.co.uk

Dan Wing  
Citrix Systems, Inc.  
USA

Email: dwing-ietf@fuggles.com

Mohamed Boucadair  
Orange  
Rennes 35000  
France

Email: mohamed.boucadair@orange.com