DNSOP WG
Internet-Draft
Intended status: Standards Track
Expires: January 27, 2021

T. Reddy
McAfee
N. Cook
Open-Xchange
D. Wing
Citrix
M. Boucadair
Orange

July 26, 2020

DNS Access Denied Error page
draft-reddy-dnsop-error-page-01

Abstract

When a DNS server filters a query the response conveys no detailed explanation of why the query was blocked, leading to end-user confusion. This document defines a method to return an URL that explains the reason the DNS query was filtered.

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 January 27, 2021.

Copyright Notice

Table of Contents

1. Introduction

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

- 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 on 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 DNS responses in real time 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.

- Network devices in a home network offer network security to protect the devices connected to 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.

- ISPs typically block access to some domains due to a requirement imposed by an external entity (e.g., Law Enforcement Agency) by performing DNS-based content filtering.

DNS responses can be filtered by sending a bogus ("forged") A or AAAA response, NXDOMAIN error or empty answer, or an extended error code defined in [I-D.ietf-dnsop-extended-error]. Each of these have advantages and disadvantages, discussed below:

1. The DNS response is forged providing IP addresses that points to a HTTP(S) server alerting the end user of the reason for blocking access to the domain (e.g., malware). When a HTTP(S) enabled domain name is blocked, the network security device presents a block page instead of the HTTP response from the content provider. 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, and the network security device(s) store 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 with the private key from the local root certificate.

   * However, configuring the local root certificate on endpoints is not 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 devices connected to their network are managed by the IT team or Mobile Device Management (MDM) devices, especially in the quite common BYOD ("Bring Your Own Device") 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 unsuccessfully reach the domain. Frustrated, the end user may use insecure interfaces to reach the domain, potentially compromising both security and privacy. Furthermore, certificate errors train users to click through certificate errors, which is poor 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 [Chrome-Install-Cert] on a host device. Doing so, however, is also poor security practice as it creates a security vulnerability that may be exploited by a MITM attack. When the 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 unsuccessfully reach the domain. Frustrated, the end user may use insecure interfaces to reach the domain, potentially compromising both security and privacy.

3. The extended error codes Blocked, Censored, and Filtered defined in [I-D.ietf-dnsop-extended-error] can be returned by the DNS server to provide additional information about the cause of an DNS error. If the extended error code "Forged answer" defined in [I-D.ietf-dnsop-extended-error] is returned by the DNS server, the client can identify the DNS response is forged and the reason for HTTPS certificate error. These extended error codes do not suffer from the limitations discussed in (1) and (2) but the user still does not know the exact reason nor the user is aware of the exact entity blocking the access to the domain. For example, a DNS server may block access domain based on the content category like "Adult Content" to enforce parental control, "Violence & Terrorism" due to an external requirement imposed by an external entity (e.g., Law Enforcement Agency), etc. The content categories for domains 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. Further, the threat data used to categorize domains may sometimes mis-classify domains (e.g., Domains wrongly classified as DGA (Domain Generation Algorithm) by deep learning techniques, domain wrongly classified as phishing due to crowd sourcing, new domains not categorized by the threat data, etc.). The end user needs to

know the contact details of the IT/InfoSec team to raise a complaint.

No matter which type of response is generated (forged IP address, NXDOMAIN or empty answer, or an extended error code), the user who generated the 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 a URL which, when accessed, provides such information to the user.

One of the other benefits of this document is eliminating the need to "spoof" block pages for HTTPS resources, as the block page no longer needs to create a signed certificate when blocking a destination. This avoids the need to install an 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] and [I-D.ietf-dnsop-terminology-ter].

'DoH/DoT' refers to DNS-over-HTTPS and/or DNS-over-TLS.

3. Method to return the error page URL

The mechanism for providing additional information about the cause of blocking access to a domain is from the HTTPS DNS record [I-D.ietf-dnsop-svcb-https]. The "HTTPS" DNS resource record type provides more information to the client before it attempts to establish a HTTPS connection. This HTTPS record in the "ServiceMode" (Section 2.4.2 of [I-D.ietf-dnsop-svcb-https]) provides the URL that gives additional information about the cause of blocking access to a domain. In order to convey an error page URL, this HTTPS record SHOULD be returned along with the "Forged Answer" extended error code in Extended DNS Error (EDE) EDNS option and MUST contain the "eut" (Section 7) parameter. The value stored in the parameter is a URL. The SvcParamKey "eut" MUST only be processed by the DNS client for a "Forged Answer" extended error code and MUST be ignored for any other type of DNS response. When the "forged answer" extended error code is returned in conjunction with an HTTPS record containing the "eut" SvcParamKey, any other resource records in the answer MUST be ignored

by clients supporting this specification. The "eut" is a single-valued SvcParamKey and the value MUST NOT be empty.

The following example shows a record containing an error page URL:

```
foo.example.com.  7200  IN HTTPS 1 . (
   eut=https://block.example.net/block-page=ZXhhbXBsZS5jb20 )
```

Figure 1: Example 1

In the above example, if the URI template is "https://block.example.net/block-page={target-domain}" for the server returning the error page and access to the target domain "example.com" is blocked, the DNS server replaces the string "{target-domain}" in the template with the base64url-encoded target domain [RFC4648].

The agent acting as HTTPS client on the endpoint uses the URL as given by the DNS server in a HTTP GET request to retrieve the error page. HTTP/2 [RFC7540] is the minimum RECOMMENDED version of HTTP to use to retrieve the error page.

4. ERROR Page

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

- 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.

- The domain name blocked.

- If query was blocked by regulation, a pointer to a regulatory text that mandates this query block.

- The entity (or organization) blocking the access to the domain and contact details of the IT/InfoSec team to raise a complaint.

- 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.

- Does not attempt to offer an exhaustive list for the contents of an error page.

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

5. Usability Considerations

The error page SHOULD be returned in the user's preferred language as expressed by the Accept-Language header. If the error page is displayed in a language not known to the end user and assuming Internationalization features failed, browser extensions to translate to user's native language can be used. For example, "Google Translate" extension [Chrome-Translate] provided by Google on Chrome can be used by the user to translate the error page. The "Google Translate" extension automatically detects whether the language of a page is different from the language the user has selected. If it is in a different language, a banner appears at the top of the page. The user can click on the Translate button in the banner to have all the text on the page appear in the language selected by the user.

6. Security Considerations

Security considerations in [I-D.ietf-dnsop-extended-error] need to be taken into consideration. Unless the DNS response that conveys the URL that provides additional information about the cause of blocking access to a domain is sent over DNS-over-HTTPS (DoH) [RFC8484] or DNS-over-TLS (DoT) [RFC7858], the DNS response is susceptible to forgery.

The agent acting as the HTTPS client on the endpoint MUST NOT fetch the URL unless DNS messages exchanged are cryptographically protected using DoH/DoT. Bad actors can host DoH/DoT servers, and claim the servers offer privacy and filtering capability to block malware domains but exactly do the opposite to invade the security and privacy of the end user. For example, this attack can be mitigated if the endpoint selects DoH/DoT servers hosted by well-known organizations (e.g., ISPs, organization for which a user works, etc.) or the user selects DoH/DoT server with filtering capability pre-configured in the OS/Browser. The DNS client can learn the filtering capability of a DoH/DoT server using [I-D.reddy-add-server-policy-selection]. [I-D.reddy-add-server-policy-selection] also discusses how a DNS client can authenticate it is connecting to a DoH/DoT server hosted by a specific organization (e.g., ISP). This information is cryptographically signed to attest its authenticity. It is particularly useful when the DoH/DoT server is insecurely discovered

and prevents the client from connecting to an attackers DoH/DoT server.

In order to deal with malicious servers, because the client knows that it is accessing a error page URL, it can know not to send cookies, not to send credentials, disable JavaScript, auto-enable private browsing mode for the error page or load the error page in a container isolated from other web activity, etc. The client MUST reject the URL if the scheme is not "https".

The DoH/DoT session provides transport security for the interaction between the DNS client and server, but DNSSEC signing and validation is not possible for the HTTPS record returning the error page URL along with the "Forged Answer" extended error.

7. IANA Considerations

7.1. Error Page URL DNS Parameter

This document adds a parameter to the "Service Binding (SVCB) Parameter" registry. If present, this parameter indicates the URL that provides additional information about the cause of blocking access to a domain is designated for use with the "Forged answer" extended error code. This is a string encoded as UTF-8 characters.

Name: eut

SvcParamKey: TBD

Meaning: URL that provides additional information about the cause of blocking access to a domain.

Reference: This document.

8. Acknowledgements

Thanks to Vittorio Bertola and Bob Harold for the comments..

9. References

9.1. Normative References

[I-D.ietf-dnsop-extended-error]
Kumari, W., Hunt, E., Arends, R., Hardaker, W., and D. Lawrence, "Extended DNS Errors", draft-ietf-dnsop-extended-error-16 (work in progress), May 2020.

[I-D.ietf-dnsop-svcb-https]
Schwartz, B., Bishop, M., and E. Nygren, "Service binding and parameter specification via the DNS (DNS SVCB and HTTPS RRs)", draft-ietf-dnsop-svcb-https-01 (work in progress), July 2020.

[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>.

[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>.

9.2. Informative References

[Chrome-Install-Cert]
"How to manually install the Securly SSL certificate in Chrome", <support.securly.com/hc/en-us/articles/206081828-How-to-manually-install-the-Securly-SSL-certificate-in-Chrome>.

[Chrome-Translate]
"Google Translate", <https://chrome.google.com/webstore/detail/google-translate/aapbdbdomjkkjkaonfhkkikfgjllcleb/RK%3D2/RS%3DBBFW_pnWkPY0xPMYsAZI5xOgQEE->.

[I-D.ietf-dnsop-terminology-ter]
Hoffman, P., "Terminology for DNS Transports and Location", draft-ietf-dnsop-terminology-ter-01 (work in progress), February 2020.

[I-D.reddy-add-server-policy-selection]
Reddy.K, T., Wing, D., Richardson, M., and M. Boucadair, "DNS Server Selection: DNS Server Information with Assertion Token", draft-reddy-add-server-policy-selection-03 (work in progress), June 2020.

[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>.

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