Transport Layer Security
Internet-Draft
Intended status: Standards Track
Expires: December 8, 2016

T. Lemon
Nominum, Inc.

June 6, 2016

Blocked Site Alerts for TLS
draft-lemon-tls-blocking-alert-00

Abstract

Hosts connecting to the Internet should generally be able to connect to all available services. However, as a matter of policy, need or preference, some services may be blocked by the network. TLS correctly treats attempts to communicate the reason for such blockage to the client as an attack. This memo describes a safe way for hosts to be notified using the TLS alert mechanism that a connection has been blocked by the network.

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 http://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 8, 2016.

Table of Contents

1. Introduction

There are a number of situations in which a connection from a particular host to a particular service on the Internet may be prohibited by policy, or may be blocked in order to redirect the user to a captive portal login. In current practice, such connections, particularly HTTP connections, are usually terminated on some sort of HTTP proxy that presents a web page notifying the user as to what happened, and possibly offering some way to address whatever problem has come up. For instance, with a captive portal, the user may be directed to log in.

Such HTTP proxies are performing what can accurately be described as a man-in-the-middle attack. Whether the purpose is benign or malicious, TLS[1] detects such attacks and, rightly, prevents them.

Unfortunately, TLS's correct behavior in this situation creates a usability problem. There is no way to notify the user as to what went wrong. This is a problem not only with HTTP connections, but also with other TLS-based connections, such as secure IMAP connections: users of captive portals are generally familiar with the phenomenon of having to reset the mail client after logging in to the captive portal, because it has concluded that the network is not usable as a result of detecting an invalid certificate.

One way to address this is to simply tell the user to click through the security warning. This is of course a disastrously bad idea, because it trains the end user to automatically permit genuinely malicious attacks.

There is no reasonable basis for trusting a proxy engaging in a MiTM attack of this sort. It would be very unsafe, for example, to provide a TLS extension that could be used by the proxy to convey any sort of server-generated status message or URL, because these would present a valuable attack surface.

However, the TLS protocol begins as a plaintext communication. A plaintext response to the initial TLS Client Hello message can include an Alert response indicating that the connection is not permitted. Alert response codes contain no information generated by the server: they simply contain a status code and an indication as to whether the alert is fatal or just a warning.

Because they provide no mechanism for a malicious attacker to trick the end user into clicking on a malicious URL, or any way to tell a careful lie to the end user, TLS alerts would seem to be a viable means of providing the client with sufficient information to present a useful error message without compromising the security of the end user.

This document defines a set of TLS alert descriptions to indicate each of the common reasons why a network service provider might block a particular connection.

2. Applicability

Alert descriptions defined in this document are intended to be used in alert messages marked fatal. If a server sends an alert using any of the codes defined in this document which is marked as a warning, the client will detect a MiTM attack once the connection progresses to the point where the server certificate can be checked.

TLS clients receiving any of the alert descriptions documented here may present a message in a user interface describing the result code that was received. TLS clients without user interfaces may log a message indicating that such an alert was received. In either case, clients should limit the rate at which such messages that are presented, to avoid denial of service or resource exhaustion.

Connections not directly initiated by a user should not result in a message being displayed in the user interface (for example, a Javascript XMLHttpRequest that generates such an alert).

Because a TLS proxy interposed between the host and the server will not know the name of the server to which the host is connecting, it may need to depend on the Server Name Indication extension [2] to provide different status codes for different servers.

3. Meanings of Alert Descriptions

3.1. Captive Portal

The 'captive_portal' alert description represents a claim by the server that the host is connected to a network behind a captive portal. This is a curable condition: the end user may be able to register with the captive portal, and subsequently a connection to the same server would not be intercepted, and could succeed. TLS clients receiving this code may choose to retry the connection periodically, frequently enough that authenticating would provide a timely resumption of service.

3.2. Malicious Site

The 'malicious_site' alert description represents a claim by the server that the host has attempted a connection to a service that is known by the network administration to serve malicious content (e.g., malware, phishing, etc.). This condition is assumed to be a permanent failure; although it may be that at some future time the same IP address is no longer marked malicious, the particular transaction that was attempted is not likely to succeed if retried.

3.3. Policy Violation

The 'policy_violation' alert description represents a claim by the server that the host has attempted to connect to a site the use of which is in violation of local policy. For example, connecting to a porn site from an enterprise network might be a policy violation.

3.4. Account Attention Requested

The 'account_attention_requested' alert description represents a claim by the server that the network service provider is requesting that the end user log in to their account. This is a temporary condition, such that an immediate attempt to reconnect can be expected to succeed reaching the correct server.

3.5. Account Attention Required

The 'account_attention_required' alert description represents a claim by the server that the network service provider is insisting that the end user log in to their account. This is not a temporary condition:

until whatever situation has motivated the service provider to place this block has been resolved, any further attempt to connect will result in the same alert description being returned.

4. Acknowledgements

Placeholder

5. IANA Considerations

The IANA is requested to allocate values for the five new TLS Alert descriptions documented here from the TLS Alert Registry. These are:

TBD captive_portal

TBD malicious_site

TBD policy_violation

TBD account_attention_requested

TBD account_attention_required

6. Security Considerations

This document attempts to avoid creating a channel of attack for a malicious attacker. However, any bit of information, no matter how small, can be used as a lever to trick the user into taking some action which will create an opportunity for attack.

The situation prior to introduction of these new alert messages is that attackers wanting to trick an end user into taking such an action can do one of two things: they can simply block the connection, which will result in the user trying to figure out what went wrong, or they can send an invalid cert and hope that the user clicks through the warnings.

A captive_portal alert might be used by the operating system as a means of directing the end user to log in to a captive portal web page. An attacker knowing the expected behavior of the operating system could trigger such an attempt. However, means of triggering such attempts already exist, so this introduces no new opportunity.

A malicious_site alert has no meaningful user mitigation response other than to stop trying to visit that site. An attacker might provide such a response as a way to prevent an end user from accessing that site in the future. To avoid this, TLS clients that receive such alerts should not cache them. An end user might still

remember that such a warning was received, and might take it more seriously than an invalid cert message. There is no means to mitigate this risk; however, the added value of being able to block malicious sites likely outweighs the possibility that a malicious attacker could succeed in tricking the user in this way.

A policy_violation alert might encourage the end user to try to find some way around the policy. The alternative is to block the connection entirely, however; this would likely trigger similar behavior in the end user, so this does not seem to be a substantial additional risk.

The account_attention_requested and account_attention_required alerts could be used to trick the end user into going to a faked version of their provider's site that is not secured using TLS and PKI. The user could then be tricked into providing authentication credentials or other personal information. Existing browser mitigation for such attacks are likely adequate, but it cannot be denied that there is some additional risk in presenting these messages to the end user. The messages could be presented along with some advice to the end user about checking to make sure that the site is secure, or even provide the user with a user interface element to click that brings them to a browser window that prevents non-TLS/non-PKI connections from succeeding.

7. Privacy Considerations

To the extent that HTTP proxies using these alert messages rely on the Server Name Indication TLS extension [2], there could be a concern that the end user's privacy might be violated if the proxy logs the SNI information sent in each request. However, there is no way at present to prevent a passive listener from capturing such information, so this does not create a new privacy risk.

8. References

8.1. Normative References

[1] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <http://www.rfc-editor.org/info/rfc5246>.

8.2. Informative References

[2] Eastlake 3rd, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, DOI 10.17487/RFC6066, January 2011, <http://www.rfc-editor.org/info/rfc6066>.

Author's Address

Ted Lemon
Nominum, Inc.
800 Bridge Parkway
Redwood City, California 94065
United States of America

Phone: +1 650 381 6000
Email: ted.lemon@nominum.com