IPv6 Operations
Internet-Draft
Intended status: Informational
Expires: April 30, 2015

E. Vyncke
Cisco

October 27, 2014

Happy Eyeballs Considerations for HTTP State Management Mechanisms
draft-vyncke-v6ops-happy-eyeballs-cookie-00

Abstract

HTTP servers usually save session states in their persistent storage indexed by session cookies generated by the HTTP servers. It is up to the HTTP user-agent to send this session cookie on each HTTP request. Some HTTP servers check whether the cookie is associated with the HTTP user-agent by the means of the user-agent IP address...

If the Happy Eyeball mechanism is used to select between IPv6 and IPv4, it may happen that while using the same HTTP server, some HTTP requests are done over IPv6 and the others over IPv4, which leads to two different sets of session states in the HTTP server. This has the consequence of inconsistencies at the HTTP server.

The only purpose of this document is to document this issue.

A similar problem arises with the use of non RFC 6888 compliant Large Scale NAT (LSN) devices used to access an IPv4-only HTTP 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 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 April 30, 2015.

Table of Contents

1. HTTP Session Management with HTTP Cookie 2
2. Other Use of Session Cookies 3
3. Happy Eyeballs Issue 3
4. Large Scale NAT Issue 4
5. Potential Mitgation 4
6. IANA Considerations 4
7. Security Considerations 4
8. Acknowledgements 4
9. Informative References 5
Author's Address 5

1. HTTP Session Management with HTTP Cookie

HTTP requests are basically stateless, therefore if a HTTP server requires to have some states associated to a HTTP user-agent (such as user name, login state, history, shopping basket, ...), there is a need to conserve those states. This is usually done by using a HTTP cookie (see also [RFC6265]) identifying the session; also called "session state cookie".

This session state cookie is generated by the HTTP server at the very first HTTP request from a HTTP user-agent. The cookie is usually opaque (often a random number) and has no semantic except as being an index within the persistent storage of the HTTP server. This index is used to access the complete state of the user-agent. This mechanism is secure if the cookie is transferred with confidentiality between the server and the user-agent. If the cookie transfer and storage are not secured, then any hostile user-agent can reuse this cookie to access the full original session states (including shopping basket, payment details, ...).

Some HTTP applications link the user-agent IP address (whether IPv6 or IPv4) to the session state, probably for additional security checks in order to prevent session cookie stealing. This link leads to some issues in a dual-stack world which are described in this document.

The author knows about at least two large web sites having this problem. It was so severe that those sites which were dual-stack had to move back to being IPv4-only... until the application and its security is updated.

2. Other Use of Session Cookies

Beside the use of session cookies by the HTTP server to keep states on the server, the very same cookie is also sometimes used by Server Load Balancing (SLB) mechanism to ensure that all HTTP requests from the same user-agent (even if behind a NAT) are always sent to the same physical HTTP server. This is required if the server persistent storage is local to the server and is not shared by all the physical servers behind the SLB.

3. Happy Eyeballs Issue

When a HTTP user-agent uses the Happy Eyeball [RFC6555] mechanism to access a HTTP server, then, part of the HTTP requests can happen over IPv6 and another part over IPv4 if the latency between IPv4 and IPv6 varies quickly over time. If there is a link between the session cookie and the user-agent IP address, then upon the first change of IP protocol version, the states associated to the cookie will be invalidated and will be deleted. Here is an example:

1. User-agent with IPv4 address, ADDR4, connect to the server by using IPv4 because IPv6 is slower; the first request does not have any HTTP cookie;

2. Server generates a new cookie C4 and stores in its persistent storage that C4 is associated with address ADDR4;

3. User-agent continues his/her session using IPv4, on each new request the HTTP server receives the cookie C4 and checks that the user-agent address is indeed ADDR4;

4. Latency of IPv6 changes and becomes now faster than IPv4;

5. User-agent now uses its IPv6 address, ADDR6, to connect to the same server and continues to use the same cookie C4 as the server name is unchanged;

6. The server receives the HTTP request with the C4 cookie and checks whether C4 is associated with ADDR6 which is not the case... All session states are deleted and a new cookie, C6, is generated and associated to the IPV6 address ADDR6;

7. The end-user becomes frustrated because he/she has to restart his/her complete session from the beginning.

This cookie invalidation may have some security benefit but it actually prevents a host using Happy Eyeballs to have a persistent session with a dual-stack HTTP server; with painful consequences for the user-experience: disconnection, loss of shopping basket, ...

4. Large Scale NAT Issue

[RFC6888] describes the LSN requirements but not all LSN implement them. Some LSN in the real world have a pool of IPv4 addresses and do not always use the same public IPv4 address for all requests from a LSN client. This obviously leads to the same problem as in section Section 3. Whether the LSN is used by IPv4 clients or by IPv6 clients does not make any difference to the problem.

5. Potential Mitgation

A potential mitigation for this issue is NOT to link any HTTP state management (including cookies) to any IP address of the HTTP user-agent.

6. IANA Considerations

This document contains no IANA considerations.

7. Security Considerations

The association of the session cookie with the user-agent IP address has some security value as it effectively prevents "session cookie stealing"; this benefit should be balanced with the lack of persistent session and the remaining vulnerability if the HTTP session can be intercepted by a man-in-the-middle attack.

8. Acknowledgements

The author would like to thank Dan Wing and Andrew Yourtchenko for some discussions on this topic.

9. Informative References

[RFC6265] Barth, A., "HTTP State Management Mechanism", RFC 6265, April 2011.

[RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, April 2012.

[RFC6888] Perreault, S., Yamagata, I., Miyakawa, S., Nakagawa, A., and H. Ashida, "Common Requirements for Carrier-Grade NATs (CGNs)", BCP 127, RFC 6888, April 2013.

Author's Address

Eric Vyncke
Cisco
De Kleetlaan 6a
Diegem 1831
Belgium

Phone: +32 2 778 4677
Email: evyncke@cisco.com