Internet Engineering Task Force (IETF)
Request for Comments: 7376
Category: Informational
ISSN: 2070-1721

T. Reddy
R. Ravindranath
Cisco
M. Perumal
Ericsson
A. Yegin
Samsung

September 2014

Problems with Session Traversal Utilities for NAT (STUN) Long-Term Authentication for Traversal Using Relays around NAT (TURN)

Abstract

This document discusses some of the security problems and practical problems with the current Session Traversal Utilities for NAT (STUN) authentication for Traversal Using Relays around NAT (TURN) messages.

Status of This Memo

Table of Contents

1. Introduction

Traversal Using Relays around NAT (TURN) [RFC5766] is a protocol that is often used to improve the connectivity of Peer-to-Peer (P2P) applications (as defined in Section 2.7 of [RFC5128]). TURN allows a connection to be established when one or both sides are incapable of a direct P2P connection. The TURN server is also a building block to support interactive, real-time communication using audio, video, collaboration, games, etc., between two peer web browsers using the Web Real-Time Communication (WebRTC) [WebRTC-Overview] framework.

A TURN server is also used in the following scenarios:

- For privacy, users of WebRTC-based web applications may use a TURN server to hide host candidate addresses from the remote peer.

- Enterprise networks deploy firewalls that typically block UDP traffic. When SIP user agents or WebRTC endpoints are deployed behind such firewalls, media cannot be sent over UDP across the firewall but must instead be sent using TCP (which causes a different user experience). In such cases, a TURN server deployed in the DeMilitarized Zone (DMZ) might be used to traverse firewalls.

- o The use case explained in Section 3.3.5 of [WebRTC-Use-Cases] ("Simple Video Communication Service, enterprise aspects") refers to deploying a TURN server in the DMZ to audit all media sessions from inside an Enterprise premises to any external peer.

- o A TURN server could also be deployed for RTP Mobility [TURN-Mobility], etc.

- o A TURN server may be used for IPv4-to-IPv6, IPv6-to-IPv6, and IPv6-to-IPv4 relaying [RFC6156].

- o Interactive Connectivity Establishment (ICE) [RFC5245] connectivity checks using server reflexive candidates could fail when the endpoint is behind a NAT [RFC3235] that performs address-dependent mapping as described in Section 4.1 of [RFC4787]. In such cases, a relayed candidate allocated from the TURN server is used for media.

Session Traversal Utilities for NAT (STUN) [RFC5389] specifies an authentication mechanism called the long-term credential mechanism. Section 4 of TURN [RFC5766] specifies that TURN servers and clients must implement this mechanism; Section 4 of TURN [RFC5766] also specifies that the TURN server must demand that all requests from the client be authenticated using this mechanism or that an equally strong or stronger mechanism for client authentication be used.

In the above scenarios, applications would use the ICE protocol for gathering candidates. An ICE agent can use TURN to learn server reflexive and relayed candidates. If the TURN server requires that the TURN request be authenticated, then the ICE agent will use the long-term credential mechanism explained in Section 10 of [RFC5389] for authentication and message integrity. Section 10 of the TURN specification [RFC5766] explains the importance of the long-term credential mechanism to mitigate various attacks. Client authentication is essential to prevent unauthorized users from accessing the TURN server, and misuse of credentials could impose significant cost on the victim TURN server.

This document focuses on listing security problems and practical problems with current STUN authentication for TURN so that it can serve as the basis for stronger authentication mechanisms.

An Allocate request is more likely than a Binding request to be identified by a server administrator as needing client authentication and integrity protection of messages exchanged. Hence, the issues discussed here regarding STUN authentication are applicable mainly in the context of TURN messages.

2. Notational Conventions

This document uses terminology defined in [RFC5389] and [RFC5766].

3. Scope

This document can be used as input for designing solution(s) to address problems with the current STUN authentication for TURN messages.

4. Problems with STUN Long-Term Authentication for TURN

1. As described in [RFC5389], the long-term credential mechanism could provide to users a long-term credential in the form of a traditional "log-in" username and password; this credential would not change for extended periods of time. The key derived from the user credentials would be used to provide message integrity for every TURN request/response. However, an attacker that is capable of eavesdropping on a message exchange between a client and server can determine the password by trying a number of candidate passwords and checking to see if one of them is correct by calculating the message integrity using these candidate passwords and comparing them with the message integrity value in the MESSAGE-INTEGRITY attribute.

2. When a TURN server is deployed in the DMZ and requires that requests be authenticated using the long-term credential mechanism as described in [RFC5389], the TURN server needs to be aware of the username and password to validate the message integrity of the requests and to provide message integrity for responses. This results in management overhead on the TURN server. Long-term credentials (username, realm, and password) need to be stored on the server side, using an MD5 hash over the credentials, which is not considered best current practice. [RFC6151] discusses security vulnerabilities of MD5 and encourages implementers not to use it. It is not possible to use STUN long-term credentials in implementations that are compliant with US FIPS 140-2 [FIPS-140-2], since MD5 isn't an approved algorithm.

3. The long-term credential mechanism discussed in [RFC5389] specifies that the TURN client must include a username value in the USERNAME STUN attribute. An adversary snooping the TURN messages between the TURN client and server can identify the users involved in the call, resulting in privacy leakage. If TURN usernames are linked to real usernames, then privacy leakage will result, but in certain scenarios TURN usernames need not be linked to any real usernames given to users, as the usernames are just provisioned on a per-company basis.

4. STUN authentication relies on HMAC-SHA1 [RFC2104]. There is no mechanism for hash agility in the protocol itself, although Section 16.3 of [RFC5389] does discuss a plan for migrating to a more secure algorithm in case HMAC-SHA1 is found to be compromised.

5. A man-in-the middle attacker posing as a TURN server challenges the client to authenticate, learns the USERNAME of the client, and later snoops the traffic from the client, thereby identifying user activity; this results in privacy leakage.

6. Hosting multiple realms on a single IP address is challenging with TURN. When a TURN server needs to send the REALM attribute in response to an unauthenticated request, it has no useful information for determining which realm it should send in the response, except the source transport address of the TURN request. Note that this is a problem with multi-tenant scenarios only; this may not be a problem when the TURN server is located in enterprise premises.

7. In WebRTC, the JavaScript code needs to know the username and password to use in the W3C RTCPeerConnection API to access the TURN server. This exposes user credentials to the JavaScript code, which could be malicious; the malicious JavaScript code could then misuse or leak the credentials. If the credentials happen to be used for accessing services other than TURN, then the security implications are much larger.

5. Security Considerations

This document lists problems with current STUN authentication for TURN so that it can serve as the basis for stronger authentication mechanisms.

6. References

6.1. Normative References

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008, <http://www.rfc-editor.org/info/rfc5389>.

[RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", RFC 5766, April 2010, <http://www.rfc-editor.org/info/rfc5766>.

[RFC6156] Camarillo, G., Novo, O., and S. Perreault, "Traversal Using Relays around NAT (TURN) Extension for IPv6", RFC 6156, April 2011, <http://www.rfc-editor.org/info/rfc6156>.

6.2. Informative References

[FIPS-140-2] NIST, "Security Requirements for Cryptographic Modules", FIPS PUB 140-2, May 2001, <http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf>.

[RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997, <http://www.rfc-editor.org/info/rfc2104>.

[RFC3235] Senie, D., "Network Address Translator (NAT)-Friendly Application Design Guidelines", RFC 3235, January 2002, <http://www.rfc-editor.org/info/rfc3235>.

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007, <http://www.rfc-editor.org/info/rfc4787>.

[RFC5128] Srisuresh, P., Ford, B., and D. Kegel, "State of Peer-to-Peer (P2P) Communication across Network Address Translators (NATs)", RFC 5128, March 2008, <http://www.rfc-editor.org/info/rfc5128>.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010, <http://www.rfc-editor.org/info/rfc5245>.

[RFC6151] Turner, S. and L. Chen, "Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms", RFC 6151, March 2011, <http://www.rfc-editor.org/info/rfc6151>.

[TURN-Mobility]
Wing, D., Patil, P., Reddy, T., and P. Martinsen, "Mobility with TURN", Work in Progress, draft-wing-tram-turn-mobility-02, September 2014.

[WebRTC-Overview]
Alvestrand, H., "Overview: Real Time Protocols for Browser-based Applications", Work in Progress, draft-ietf-rtcweb-overview-11, August 2014.

[WebRTC-Use-Cases]
Holmberg, C., Hakansson, S., and G. Eriksson, "Web Real-Time Communication Use-cases and Requirements", Work in Progress, draft-ietf-rtcweb-use-cases-and-requirements-14, February 2014.

Acknowledgments

The authors would like to thank Dan Wing, Harald Alvestrand, Sandeep Rao, Prashanth Patil, Pal Martinsen, Marc Petit-Huguenin, Gonzalo Camarillo, Brian E. Carpenter, Spencer Dawkins, Adrian Farrel, and Simon Perreault for their comments and reviews.

Authors' Addresses

Tirumaleswar Reddy
Cisco Systems, Inc.
Cessna Business Park, Varthur Hobli
Sarjapur Marathalli Outer Ring Road
Bangalore, Karnataka 560103
India

EMail: tireddy@cisco.com

Ram Mohan Ravindranath
Cisco Systems, Inc.
Cessna Business Park, Varthur Hobli
Sarjapur Marathalli Outer Ring Road
Bangalore, Karnataka 560103
India

EMail: rmohanr@cisco.com

Muthu Arul Mozhi Perumal
Ericsson
Ferns Icon
Doddanekundi, Mahadevapura
Bangalore, Karnataka 560037
India

EMail: muthu.arul@gmail.com

Alper Yegin
Samsung
Istanbul
Turkey

EMail: alper.yegin@yegin.org