Network Working Group T. Hardie
Internet-Draft March 27, 2017
Intended status: Informational
Expires: September 28, 2017

Design considerations for Metadata Insertion
draft-hardie-privsec-metadata-insertion-08

Abstract

The IAB has published RFC7624 in response to several revelations of pervasive attack on Internet communications. This document considers the implications of protocol designs which associate metadata with encrypted flows. In particular, it asserts that designs which do so by explicit actions at the host are preferable to designs in which middleboxes insert them.

Table of Contents

1. Introduction

To minimize the risks associated with pervasive surveillance, it is necessary for the Internet technical community to address the vulnerabilities exploited in the attacks document in [RFC7258] and the threats described in [RFC7624]. The goal of this document is to address a common design pattern which emerges from the increase in encryption: explicit association of metadata which would previously have been inferred from the plaintext protocol.

2. Terminology

This document makes extensive use of standard security and privacy terminology; see [RFC4949] and [RFC6973]. Terms used from [RFC6973] include Eavesdropper, Observer, Initiator, Intermediary, Recipient, Attack (in a privacy context), Correlation, Fingerprint, Traffic Analysis, and Identifiability (and related terms). In addition, we use terms that are specific to the attacks discussed in [RFC7624]. Terms introduced terms from there include: Pervasive Attack, Passive Pervasive Attack, Active Pervasive Attack, Observation, Inference, and Collaborator.

3. Design pattern

One of the core mitigations for the loss of confidentiality in the presence of pervasive surveillance is data minimization, which limits the amount of data disclosed to those elements absolutely required to complete the relevant protocol exchange. When data minimization is in effect, some information which was previously available may be removed from specific protocol exchanges. The information may be

removed explicitly (by a browser suppressing cookies during private modes, as an example) or by other means. As noted in [RFC7624], some topologies which aggregate or alter the network path also acted to reduce the ease with which metadata is available to eavesdroppers.

In some cases, other actors within a protocol context will continue to have access to the information which has been thus withdrawn from specific protocol exchanges. If those actors attach the information as metadata to those protocol exchange, the confidentiality effect of data minimization is lost.

The restoration of information is particularly tempting for systems whose primary function is not to provide confidentiality. A proxy providing compression, for example, may wish to restore the identity of the requesting party; similarly a VPN system used to provide channel security may believe that origin IP should be restored. Actors considering restoring metadata may believe that they understand the relevant privacy considerations or believe that, because the primary purpose of the service was not privacy-related, none exist. Examples of this design pattern include [RFC7239] and [RFC7871].

4. Advice

Avoid inserting metada to restore information which would otherwise be unavailable to later participants in a protocol exchange. It contributes to the overall loss of confidentiality for the Internet and trust in the Internet as a medium. Do not add metadata to flows at intermediary devices unless a positive affirmation of approval for restoration has been received from the actor whose data will be added.

Instead, design the protocol so that the actor can add such metadata themselves so that it flows end-to-end, rather than requiring the action of other parties. In addition to improving privacy, this approach ensures consistent availability between the communicating parties, no matter what path is taken. (Note that this document does not attempt to describe how an actor sets policies on providing this metadata, as the range of systems which might be implied is very broad).

As an example, RFC 7871 describes a method that had already been deployed and notes that it is unlikely that a clean-slate design would result in this mechanism. If a clean-slate design were built to follow the advice in this document, that design would likely would not use a core element of RFC 7871: rather than adding metadata at a proxy, it would provide facilities for end systems to add it to their initial queries. In the case of RFC 7871, the relevant metadata is

relatively easy for an end system to derive, as STUN [RFC5389] provides a method for learning the reflexive transport address from which a client subnet could be derived. This would allow clients to populate this data themselves, thus affirming their consent and providing data at a granularity with which they were comfortable. As in RFC 7871, the addition of this data would require confirmation that the upstream DNS resolver understood what to do with it, but the same negotiation mechanism, an EDNS0 option [RFC6891], could be used. Because of this negotiation, there would be a new variability in responses that would change the caching behavior for data supplied by participating servers. This not a major change from the current design, however, as the same considerations set out in section 7.3.2 and 7.5 of RFC 7871 would apply to client-supplied subnets as well as they do for proxy supplied subnets.

From a protocol perspective, in other words, this approach would be a minor change from RFC 7871, would be as fully featured and would provide better privacy properties than the on-path update mechanism RFC 7871 provides. The next section examines why, despite this, deployment considerations have sometimes trumped cleaner designs.

5. Deployment considerations

There are a few common tensions associated with the deployment of systems which restore metadata. The first is the trade-off in speed of deployment for different actors. The Forwarded HTTP Extension in [RFC7239] provides an example of this. When used with a proxy, it restores information related to the original requesting party, thus allowing a responding server to tailor responses according to the original party's region, network, or other characteristics associated with the identity. It would, of course, be possible for the originating client to add this data itself, after using STUN [RFC5389] or a similar mechanism to first determine the information to declare. This would require, however, full specification and adoption of this mechanism by the end systems. It would not be available at all during this period, and would thereafter be limited to those systems which have been upgraded to include it. The long tail of browser deployments indicates that many systems might go without upgrades for a significant period of time. The proxy infrastructure, in contrast, is commonly under more active management and represents a much smaller number of elements; this impacts both the general deployment difficulty and the number of systems which the origin server must trust.

The second common tension is between the metadata minimization and the desire to tailor content responses. For origin servers whose content is common across users, the loss of metadata may have limited impact on the system's functioning. For other systems, which

commonly tailor content by region or network, the loss of metadata may imply a loss of functionality. Where the user desires this functionality, restoration can commonly be achieved by the use of other identifiers or login procedures. Where the user does not desire this functionality, but it is a preference of the server or a third party, adjustment is more difficult. At the extreme, content blocking by network origin may be a regulatory requirement. Trusting a network intermediary to provide accurate data is, of course, fragile in this case, but it may be a part of the regulatory framework.

There are also tensions with latency of operation. For example, where the end system does not initially know the information which would be added by on-path devices, it must engage the protocol mechanisms to determine it. Determining a public IP address to include in a locally supplied header might require a STUN exchange, and the additional latency of this exchange discourages deployment of host-based solutions. To minimize this latency, engaging those mechanisms may need to be done in parallel with or in advance of the core protocol exchanges with which this metadata would be supplied.

These tensions do not change the basic recommendation, but they suggest that the parties who are introducing encryption and data minimization for existing protocols consider carefully whether the work also implies introducing mechanisms for the end-to-end provisioning of metadata when a user has actively consented to provide it.

6. IANA Considerations

This memo makes no request of IANA.

7. Security Considerations

This memorandum describes a design pattern related emerging from responses to the attacks described in [RFC7258]. Continued use of this design pattern, which uses mid-flow devices to restore metadat, lowers the impact of mitigations to that attack.

Note that some emergency service recipients, notably PSAPs (Public Safety Answering Points) may prefer data provided by a network to data provided by end system, because an end system could use false data to attack others or consume resources. While this has the consequence that the data available to the PSAP is often more coarse than that available to the end system, the risk of false data being provided involved a risk to the lives of those targeted.

8. Contributors

This document is derived in part from the work initially done on the Perpass mailing list and at the [STRINT] workshop. It has been discussed with the IAB's Privacy and Security program, whose review and input is gratefully acknowledged. The document also benefited from an extensive review by Mohamed Boucadair.

9. References

9.1. Normative References

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", FYI 36, RFC 4949, DOI 10.17487/RFC4949, August 2007, <http://www.rfc-editor.org/info/rfc4949>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <http://www.rfc-editor.org/info/rfc6973>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <http://www.rfc-editor.org/info/rfc7258>.

[RFC7624] Barnes, R., Schneier, B., Jennings, C., Hardie, T., Trammell, B., Huitema, C., and D. Borkmann, "Confidentiality in the Face of Pervasive Surveillance: A Threat Model and Problem Statement", RFC 7624, DOI 10.17487/RFC7624, August 2015, <http://www.rfc-editor.org/info/rfc7624>.

9.2. Informative References

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, DOI 10.17487/RFC5389, October 2008, <http://www.rfc-editor.org/info/rfc5389>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <http://www.rfc-editor.org/info/rfc6891>.

[RFC7239] Petersson, A. and M. Nilsson, "Forwarded HTTP Extension", RFC 7239, DOI 10.17487/RFC7239, June 2014, <http://www.rfc-editor.org/info/rfc7239>.

[RFC7871] Contavalli, C., van der Gaast, W., Lawrence, D., and W. Kumari, "Client Subnet in DNS Queries", RFC 7871, DOI 10.17487/RFC7871, May 2016, <http://www.rfc-editor.org/info/rfc7871>.

[STRINT] S Farrell, ., "Strint Workshop Report", April 2014, <https://www.w3.org/2014/strint/draft-iab-strint-report.html>.

Author's Address

Ted Hardie

Email: ted.ietf@gmail.com