Internet Engineering Task Force   R. Winter
Internet-Draft   M. Faath
Intended status: Informational   F. Weisshaar
Expires: May 4, 2017   University of Applied Sciences Augsburg
                                                        October 31, 2016

Privacy considerations for IP broadcast and multicast protocol designers
draft-ietf-intarea-broadcast-consider-01

Abstract

A number of application-layer protocols make use of IP broadcasts or multicast messages for functions like local service discovery or name resolution. Some of these functions can only be implemented efficiently using such mechanisms. When using broadcasts or multicast messages, a passive observer in the same broadcast/multicast domain can trivially record these messages and analyze their content. Therefore, broadcast/multicast protocol designers need to take special care when designing their protocols.

Table of Contents

1. Introduction

Broadcast and multicast messages have a large (and to the sender unknown) receiver group by design. Because of that, these two mechanisms are vital for a number of basic network functions such as auto-configuration. Application developers use broadcast/multicast messages to implement things like local service or peer discovery and it appears that an increasing number of applications make use of it. And, as RFC 919 [RFC0919] puts it, "The use of broadcasts [...] is a good base for many applications".

Using broadcast/multicast can become problematic if the information that is being distributed can be regarded as sensitive or when the information that is distributed by multiple of these protocols can be correlated in a way that sensitive data can be derived. This is clearly true for any protocol, but broadcast/multicast is special in at least two respects:

(a) The aforementioned large receiver group, consisting of receivers unknown to the sender. This makes eavesdropping without special privileges or a special location in the network trivial for anybody in the broadcast/multicast domain.

(b) Encryption is more difficult when broadcast/multicast messages, leaving content of these messages in the clear and making it easier to spoof and replay them.

Given the above, privacy protection for protocols based on broadcast or multicast communication is significantly more difficult compared to unicast communication and at the same time invading the privacy is much easier.

Privacy considerations of IETF-specified protocols have received some attention in the recent past (e.g. RFC 7721 [RFC7721] or RFC 7919 [RFC7819]). There is also general guidance available for document authors on when and how to include a privacy considerations section in their documents and on how to evaluate the privacy implications of Internet protocols [RFC6973]. RFC6973 also describes potential threats to privacy in great detail and lists terminology that is also used in this document.

In contrast to RFC6973, this document contains a number of privacy considerations especially for broadcast/multicast protocol designers that are intended to reduce the likelihood that a broadcast/multicast protocol can be misused to collect sensitive data about devices, users and groups of users on a broadcast/multicast domain. These considerations particularly apply to protocols designed outside the IETF for two reasons. For one, non-standard protocols will likely not receive operational attention and support in making them more secure such as e.g. DHCP snooping does for DHCP because they typically are not documented. The other reason is that these protocols have been designed in isolation, where a set of considerations to follow is useful in the absence of a larger community providing feedback. In particular, carelessly designed broadcast/multicast protocols can break privacy efforts at different layers of the protocol stack such as MAC address or IP address randomization [RFC4941].

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Privacy considerations

There are a few obvious and a few not necessarily obvious things designers of broadcast/multicast protocols should consider in respect to the privacy implications of their protocol. Most of these items are based on protocol behavior observed as part of experiments on operational networks [TRAC2016].

2.1. Message frequency

Frequent broadcast/multicast traffic caused by an application can give user behavior and online times away. This allows a passive observer to potentially deduce a user's current activity (e.g. a game) and it allows to create an online profile (i.e. times the user is on the network). The higher the frequency of these messages, the more accurate this profile will be. Given that broadcasts/multicasts are only visible in the same broadcast/multicast domain, these messages also give the rough location of the user away (e.g. a campus or building).

This behavior has e.g. been observed by a synchronization mechanism of a popular application, where multiple messages have been sent per minute via broadcast. Given this behavior, it is possible to record a device's time on the network with a sub-minute accuracy given only the traffic of this single application installed on the device. But also services used for local name resolution in modern operating systems utilize broadcast/multicast protocols (e.g. mDNS, LLMNR or NetBIOS) to announce for example their shares regularly and allow a tracking of the online time of a device.

If a protocol relies on frequent or periodic broadcast/multicast messages, the frequency SHOULD be chosen conservatively, in particular if the messages contain persistent identifiers (see next subsection). Also, intelligent message suppression mechanisms such as the ones employed in mDNS [RFC6762] SHOULD be implemented. The lower the frequency of broadcast messages, the harder traffic analysis and surveillance becomes.

2.2. Persistent identifiers

A few broadcast/multicast protocols observed in the wild make use of persistent identifiers. This includes the use of host names or more abstract persistent identifiers such as a UUID or similar. These IDs, which e.g. identify the installation of a certain application might not change across updates of the software and are therefore extremely long lived. This allows a passive observer to track a user precisely if broadcast/multicast messages are frequent. This is even true in case the IP and/or MAC address changes. Such identifiers also allow two different interfaces (e.g. WiFi and Ethernet) to be correlated to the same device. If the application makes use of persistent identifiers for multiple installations of the same application for the same user, this even allows to infer that different devices belong to the same user.

The aforementioned broadcast messages from a synchronization mechanism of a popular application also included a persistent

identifier in every broadcast. This identifier did never change after the application was installaed and allowed to track a device even when it changed its network interface or when it connected to a different network.

If a broadcast/multicast protocol relies on IDs to be transmitted, it SHOULD be considered if frequent ID rotations are possible in order to make user tracking more difficult. Persistent IDs are considered bad practice in general for broadcast and multicast communication as persistent application layer IDs will make efforts on lower layers to randomize identifiers (e.g. [I-D.huitema-6man-random-addresses]) useless or at least much more difficult.

2.3. Anticipate user behavior

A large number of users name their device after themselves, either using their first name, last name or both. Often a host name includes the type, model or maker of a device, its function or includes language specific information. Based on gathered data, this appears currently to be prevalent user behavior [TRAC2016]. For protocols using the host name as part of the messages, this clearly will reveal personally identifiable information to everyone on the local network. This information can also be used to mount more sophisticated attacks, when e.g. the owner of a device is identified (as an interesting target) or properties of the device are known (e.g. known vulnerabilities).

A popular operating system vendor includes the name the user chooses for the user account during the installation process as part of the host name of the device. The name of the operating system is also included, revealing therefore two pieces of information, which can be regarded as private information if the host name is used in broadcast/multicast messages.

Where possible, the use of host names and other user provided information in broadcast/multicast protocols SHOULD be avoided. If only a persistent ID is needed, this can be generated. An application might want to display the information it will broadcast on the LAN at install/config time, so the user is at least aware of the application's behavior. More host name considerations can be found in [I-D.ietf-intarea-hostname-practice]. More information on user participation can be found in RFC 6973 [RFC6973].

2.4. Consider potential correlation

A large number of services and applications make use of the broadcast/multicast mechanism. That means there are various sources of information that are easily accessible by a passive observer. In

isolation, the information these protocols reveal might seem harmless, but given multiple such protocols, it might be possible to correlate this information. E.g. a protocol that uses frequent messages including a UUID to identify the particular installation does not give the identity of the user away. But a single message including the user's host name might just do that and it can be correlated using e.g. the MAC address of the device's interface.

In the experiments described in [TRAC2016], it was possible to correlate frequently sent broadcast messages that included a unique identifier with other broadcast/multicast messages containing usernames (e.g. mDNS, LLMNR or NetBIOS), but also relationships to other users. This allowed to reveal the real identity of the users of many devices but it also gave some information about their social environment away.

A broadcast protocol designer should be aware of the fact that even if - in isolation - the information a protocol leaks seems harmless, there might be ways to correlate that information with other broadcast protocol information to reveal sensitive information about a user.

2.5. Configurability

A lot of applications and services using broadcast/multicast protocols do not include the means to declare "safe" environments (e.g. based on the SSID of a WiFi network and the MAC addresses of the access points). E.g. a device connected to a public WiFi will likely broadcast the same information as when connected to the home network. It would be beneficial if certain behavior could be restricted to "safe" environments.

A popular operating system e.g. allows the user to specify the trust level of the network the device connects to, which for example restricts specific system services (using broadcast/multicast messages for their normal operation) to be used in untrusted networks. Such functionality could implemented as part of an application.

An application developer making use of broadcasts/multicasts as part of the application SHOULD make the broadcast feature, if possible, configurable, so that potentially sensitive information does not leak on public networks, where the thread to privacy is much larger.

3. Operational considerations

Besides changing end-user behavior, choosing sensible defaults as an operating system vendor (e.g. for suggesting host names) and the considerations for protocol designers mentioned in this document, there are things that the network administrators/operators can do to limit the above mentioned problems.

A feature not uncommonly found on access points e.g. is to filter broadcast and multicast traffic. This will potentially break certain applications or some of their functionality but will also protect the users from potentially leaking sensitive information.

4. Summary

Increasingly, applications rely on broadcast and multicast messages. For some, broadcasts/multicasts are the basis of their application logic, others use broadcasts/multicasts to improve certain aspects of the application but are fully functional in case broadcasts/ multicasts fail. Irrespective of the role of broadcast and multicast messages for the application, the designers of protocols that make use of them should be very careful in their protocol design because of the special nature of broad- and multicast.

It is not always possible to implement certain functionality via unicast, but in case a protocol designer chooses to rely on broadcast/multicast, the following should be carefully considered:

- IETF-specified protocols, such as mDNS [RFC6762], should be used if possible as operational support might exist to protect against the leakage of private information

- Avoid using user-specified information inside broadcast/multicast messages as users will often use personal information or other information aiding attackers, in particular if the user is unaware about how that information is being used

- Avoid persistent IDs in messages as this allows user tracking, correlation and potentially has a devastating effect on other privacy protection mechanisms

- If you really must use a broadcast/multicast protocol and cannot use an IETF-specified protocol, then:

  * Be very conservative in how frequently you send messages as an effort in data minimization

* Seek advice from IETF-specifies protocols such as message suppression in mDNS

* Try to design the protocol in a way that the information cannot be correlated with other information in broadcast/multicast messages

* Let the user configure safe environments if possible (e.g. based on the SSID)

[Note: discussions on this document should be take place on the Intarea mailing list of the IETF. Subscription: https://www.ietf.org/mailman/listinfo/int-area, Mailing list archive: https://www.ietf.org/mail-archive/web/int-area/current/maillist.html]

5. Other considerations

Besides the privacy implications of frequent broadcasting, it also represents a performance problem. In particular in certain wireless technologies such as 802.11, broadcast and multicast are transmitted at a much lower rate (the lowest common denominator rate) compared to unicast and therefore have a much bigger impact on the overall available airtime. Further, it will limit the ability for devices to go to sleep if frequent broadcasts are being sent. A similar problem in respect to Router Advertisements is addressed in [I-D.ietf-v6ops-reducing-ra-energy-consumption]. In that respect broadcasts can be used for another class of attacks that not related to privacy. The potential impact on network performance should nevertheless be considered by broadcast protocol designers.

6. Acknowledgments

We would like to thank Eliot Lear and Stephane Bortzmeyer for their input.

This work was partly supported by the European Commission under grant agreement FP7-318627 mPlane. Support does not imply endorsement.

7. IANA Considerations

This memo includes no request to IANA.

8. Security Considerations

This document deals with privacy-related considerations of broadcast- and multicast-based protocols. It contains advice for designers of such protocols to minimize the leakage of privacy-sensitive information. The intent of the advice is to make sure that

identities will remain anonymous and user tracking will be made difficult.

9. Informative References

[I-D.huitema-6man-random-addresses]
Huitema, C., "Implications of Randomized Link Layers Addresses for IPv6 Address Assignment", draft-huitema-6man-random-addresses-03 (work in progress), March 2016.

[I-D.ietf-intarea-hostname-practice]
Huitema, C. and D. Thaler, "Current Hostname Practice Considered Harmful", draft-ietf-intarea-hostname-practice-00 (work in progress), October 2015.

[I-D.ietf-v6ops-reducing-ra-energy-consumption]
Yourtchenko, A. and L. Colitti, "Reducing energy consumption of Router Advertisements", draft-ietf-v6ops-reducing-ra-energy-consumption-03 (work in progress), November 2015.

[RFC0919] Mogul, J., "Broadcasting Internet Datagrams", STD 5, RFC 919, DOI 10.17487/RFC0919, October 1984, <http://www.rfc-editor.org/info/rfc919>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, DOI 10.17487/RFC4941, September 2007, <http://www.rfc-editor.org/info/rfc4941>.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <http://www.rfc-editor.org/info/rfc6762>.

[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>.

[RFC7721] Cooper, A., Gont, F., and D. Thaler, "Security and Privacy Considerations for IPv6 Address Generation Mechanisms", RFC 7721, DOI 10.17487/RFC7721, March 2016, <http://www.rfc-editor.org/info/rfc7721>.

[RFC7819] Jiang, S., Krishnan, S., and T. Mrugalski, "Privacy Considerations for DHCP", RFC 7819, DOI 10.17487/RFC7819, April 2016, <http://www.rfc-editor.org/info/rfc7819>.

[TRAC2016] Faath, M., Weisshaar, F., and R. Winter, "How Broadcast Data Reveals Your Identity and Social Graph", 7th International Workshop on TRaffic Analysis and Characterization IEEE TRAC 2016, September 2016.

Authors' Addresses

Rolf Winter
University of Applied Sciences Augsburg
Augsburg
DE

Email: rolf.winter@hs-augsburg.de

Michael Faath
University of Applied Sciences Augsburg
Augsburg
DE

Email: michael.faath@hs-augsburg.de

Fabian Weisshaar
University of Applied Sciences Augsburg
Augsburg
DE

Email: fabian.weisshaar@hs-augsburg.de