RADIUS Extensions Working Group
Internet-Draft
Updates: 3748 (if approved)
Intended status: Best Current Practice
Expires: January 9, 2017

S. Winter
RESTENA

July 08, 2016

Considerations regarding the correct use of EAP-Response/Identity
draft-ietf-radext-populating-eapidentity-01

Abstract

There are some subtle considerations for an EAP peer regarding the content of the EAP-Response/Identity packet when authenticating with EAP to an EAP server. This document describes two such considerations and suggests workarounds to the associated problems. One of these workarounds is a new requirement for EAP peers that the use of UTF-8 is required for the content of EAP-Response/Identity (which updates RFC3748).

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 January 9, 2017.

Copyright Notice

Table of Contents

1. Introduction

1.1. Problem Statement

An Extensible Authentication Protocol (EAP, [RFC3748]) conversation between an EAP peer and an EAP server starts with an (optional) request for identity information by the EAP server (EAP-Request/ Identity) followed by the peer's response with identity information (EAP-Response/Identity). Only after this identity exchange are EAP types negotiated.

EAP-Response/Identity is sent before EAP type negotiation takes place, but it is not independent of the later-negotiated EAP type. Two entanglements between EAP-Response/Identity and EAP methods' notions of a user identifier are described in this document.

1. The choice of identifier to send in EAP-Response/Identity may have detrimental effects on the subsequent EAP type negotiation.

2. Using identifiers from the preferred EAP type without thoughtful conversion of character encoding may have detrimental effects on the outcome of the authentication.

The following two chapters describe each of these issues in detail. The last chapter contains recommendations for implementers of EAP peers to avoid these issues.

1.2. Taxonomy of identities in EAP

The notion of identity occurs numerous times in the EAP protocol stack (EAP-Response/Identity, Outer identity, method-specific identity, tunneled identity). This document uses the following terminology when discussing EAP identities.

- User Identifier: Each EAP method has a means to identify the user or machine that tries to authenticate. There are no restrictions on the format or encoding of this identifier. The user identifier is often also referred to as "method-specific identity". If an EAP method distinguishes between the user identifier and a realm identifier (see next bullet), then the user identifier is also often referred to as the "inner/true/real identity".

- Realm Identifier: Some EAP methods allow privacy-preserving enhancements where a string is sent which is actually not necessarily related to the user or machine that tries to authenticate. This identifier is often also referred to as "outer identity" or "roaming identity" or "anonymous outer identity". There is often a relationship between the realm identifier and the user identifier (e.g. they often share the same NAI realm suffix); but this is not a requirement. There are no restrictions on the format or encoding of the realm identifier. Realm identifiers are either

  * explicitly configured (e.g. string input UI in EAP peer: "Outer Identity")

  * implicitly configured by copying the actual user identifier

  * implicitly configured by copying the NAI realm of the user identifier and prefixing it non-configurably with a fixed privacy-preserving local username part like "anonymous" or the empty string (see [RFC7542])

* configured in a mixed way, e.g. using a explicit string input UI for the local part of the realm identifier and combining it implicitly with a copy of the NAI realm part of the user identifier

o EAP-Response/Identity: a string representing the user or machine that tries to authenticate, used outside the EAP method-specific context for the entire EAP conversation. There can be only one EAP-Response/Identity per EAP conversation, even if that conversation could negotiate more than one EAP method to authenticate with. As per [RFC3748] there is no encoding requirement on EAP-Response/Identity (which this document changes:

the encoding MUST be UTF-8). In AAA protocol routing contexts, the content of EAP-Response/Identity is often used for request routing purposes. EAP-Response/Identity is chosen from the set:

* all realm identifiers from all configured EAP types supporting the notion of a realm identifier

* all user identifiers from all configured EAP types without the notion of a realm identifier

Several EAP types in a local configuration may share the same user and/or realm identifiers. The set of identifiers for EAP-Response /Identity may thus contain fewer elements than there are configured EAP types in a local configuration. One of the two problems addressed in this document stems from this fact: the set of identifiers may contain more than one element. The resulting EAP-Response/Identity always routes all configured EAP types to only one destination, even if different EAP types would need routing to different destinations.

o User-Name: when using EAP in AAA protocol contexts (e.g. RADIUS [RFC2865], Diameter [RFC6733]), this additional identifier is created outside the EAP peer (typically in a pass-through authenticator) by copying EAP-Response/Identity content to the AAA protocol's User-Name attribute. There is no format requirement on User-Name, but there is an encoding requirement: the string MUST be UTF-8 encoded. One of the two problems addressed in this document stems from this fact: EAP-Response/Identity does not have an encoding requirement, nor does it carry meta-information about

the encoding used - and yet, it needs to be coerced into a UTF-8 encoding.

- Further identifiers: Some EAP methods establish an EAP session inside EAP (e.g. PEAP first establishes a TLS tunnel using a realm identifier, and then starts an EAP exchange inside the tunnel). This being a new, independent EAP session, it contains its own EAP-Response/Identity, can invoke EAP method negotiation with different (inner) EAP types (this happens e.g. with EAP-FAST and its configurable choice of EAP-GTC or EAP-MSCHAPv2 inside the inner EAP session), and those inner EAP methods then have their own user identifiers. Where the inner EAP method itself supports the notion of realm identifiers, another identifier could be configured. For the purposes of this document, none of those details are considered and the process by which the (outer) EAP method selects its user identifier is left entirely to that EAP type. This document does not consider the (inner) EAP-Response/Identity in scope; the recommendations in this document to not apply to such (inner) occurences of EAP-Response/Identity.

1.3. Requirements Language

In this document, several words are used to signify the requirements of the specification. 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. EAP-Response/Identity: Effects on EAP type negotiation

Assuming the EAP peer's EAP type selection is not the trivial case (i.e. it has more than one configured EAP type for a given network or application, and needs to make a decision which one to use), an issue arises when the configured EAP types are not all configured with the same realm identifier (or user identifier for EAP types not supporting the notion of a realm identifier).

Issue: if the identifiers in the set of configured EAP types differ (e.g. have a different [RFC7542] "realm" portion), and the authenticator does not send identity selection hints as per [RFC7542], then EAP type negotiation may be limited to those EAP types which are terminated in the same EAP server. The reason for

that is because the information in the EAP-Response/Identity is used for request routing decisions and thus determines the EAP server - a given realm identifier may be routed to a server which exclusively serves the corresponding EAP types. Negotiating another EAP type from the set of configured EAP types during the running EAP conversation is then not possible.

Example:

Assume an EAP peer is configured to support two EAP types:

- EAP-AKA' [RFC5448] with user identifier imsi@mnc123.mcc123.3gppnetwork.org; the configuration is set up to authenticate only to
  - cellular networks
  - Wi-Fi Passpoint networks which advertise support for the MNC 123 and MCC 123

  The EAP server for this EAP type is in a host under control of the 3GPP consortium

- EAP-TTLS [RFC5281] with user identifier "john@realm.example" and realm identifier "@realm.example"; the configuration is set up to authenticate only to

  - Wi-Fi networks with the SSID "eduroam"
  - Wi-Fi Passpoint networks which advertise support for the roaming consortium 00-1B-C5-04-60 (the eduroam consortium)
  - wired ethernet

  The EAP server for this EAP type is in a host under control of the eduroam consortium

The user approaches a Passpoint Wi-Fi hotspot with SSID "arbitrary" which emits a beacon advertising support for the MNC 123/MCC 123 AND for the consortium identifier 00-1B-C5-04-60. The local configuration thus yields two different EAP type candidates for authentication to the network. Unbeknownst to the user's device, the

credit with the 3G provider is fully depleted and the user will be unable to authenticate with his EAP-AKA' credentials. Using his identifier of the roaming consortium eduroam (see also [RFC7593]), he could authenticate with EAP-TTLS and his john@realm.example user identifier. Identity selection hints are not sent.

Consequence: If the EAP peer consistently chooses the imsi@mnc123.mcc123.3gpp-network.org user identifier as choice for its initial EAP-Response/Identity, requests will always be routed to the 3GPP consortium EAP server, and the user will be consistently and perpetually rejected, even though in possession of a valid credential for the hotspot.

An EAP peer should always try all options to authenticate. As the example above shows, it may not be sufficient to rely on EAP method negotiation alone to iterate through all configured EAP types and come to a conclusive outcome of the authentication attempt. Multiple new EAP authentications, each using an EAP-Response/Identity from a different element of the set of realm identifiers, may be required to fully iterate through the list of usable identities.

3. Character (re-)encoding may be required

The user identifiers as configured in the EAP method configuration are not always suited as realm identifiers to choose as EAP-Response/Identity: EAP methods define the encoding of their method-specific outer identities at their leisure; in particular, the chosen encoding may or may not be UTF-8.

It is not the intention of EAP, as a mere method-agnostic container which simply carries EAP types, to restrict an EAP method's choice of encoding of user identifiers. However, there are restrictions in what should be contained in the EAP-Response/Identity: EAP is very

often carried over a AAA protocol (e.g over RADIUS as per [RFC3579]). The typical use for the contents of EAP-Response/Identity inside AAA protocols like RADIUS [RFC2865] and Diameter [RFC6733] is to copy the content of EAP-Response/Identity into a "User-Name" attribute; the encoding of the User-Name attribute is required to be UTF-8. EAP-Response/Identity does not carry encoding information itself, so a conversion between a non-UTF-8 encoding and UTF-8 is not possible for the AAA entity doing the EAP-Response/Identity to User-Name copying.

Consequence: If an EAP method's user identifier is not encoded in UTF-8, and if the EAP peer verbatimly uses that user identifier for its EAP-Response/Identity field, then the AAA entity is forced to violate its own specification because it has to, but can not use UTF-8 for its own User-Name attribute. If the EAP method supports a separate realm identifier in a non UTF-8 character set, and the EAP peer verbatimly uses that realm identifier for its EAP-Response/ Identity field, then the same violation occurs.

This jeopardizes the subsequent EAP authentication as a whole; request routing may fail, lead to a wrong destination or introduce routing loops due to differing interpretations of the User-Name in EAP pass-through authenticators and AAA proxies.

4. Recommendations for EAP peer implementations

Where realm identifiers or user identifiers between multiple configured EAP types in an EAP peer differ, the EAP peer can not rely on the EAP type negotiation mechanism alone to provide useful results. If an EAP authentication gets rejected, the EAP peer SHOULD re-try the authentication using a different EAP-Response/Identity than before. The EAP peer SHOULD try all possible EAP-Response/ Identity contents from the entire set of configured EAP types before declaring final authentication failure.

EAP peers need to maintain state on the encoding of the configured user identifiers and realm identifiers which are used in their local EAP type configuration. When constructing an EAP-Response/Identity from the set of available identifiers, they MUST (re-)encode the corresponding identifier as UTF-8 and use the resulting value for the EAP-Response/Identity.

Where an EAP method supports privacy-preserving realm identifiers, those SHOULD be configured for user privacy reasons. For deployments of such EAP types, these realm identifiers MUST be in the the format Network Access Identifier (NAI), see [RFC7542] if the realm identifiers are expected to become used beyond the scope of a single, closed enterprise. Even in such closed environments, the NAI format is RECOMMENDED. The RECOMMENDED format for the local part of the

realm identifier is the empty string; where this is not possible the

suggested alternative is the string "anonymous".

5. Privacy Considerations

Because the EAP-Response/Identity content is not encrypted, the backtracking to a new EAP-Response/Identity will systematically reveal all configured identifiers to intermediate passive listeners on the path between the EAP peer and the EAP server (until one authentication round succeeds).

This additional leakage of identity information is not very significant though, because where privacy is considered important, the additional option for separate privacy-preserving realm identifiers which is present in most modern EAP methods can and should be used.

If the EAP peer implementation is certain that all EAP types will be terminated at the same EAP server (e.g. with a corresponding configuration option) then the iteration over all identities can be avoided, because EAP type negotiation is then sufficient.

If a choice of which identity information to disclose needs to be made by the EAP peer, when iterating through the list of identifiers the EAP peer SHOULD

- in first priority honour a manually configured order of preference of EAP types, if any

- in second priority try EAP types in order of less leakage first; that is, EAP types with a privacy-preserving realm identifier that differs from the user identifier should be tried before other EAP types which would reveal the corresponding actual user identifiers.

6. Security Considerations

The security of an EAP conversation is determined by the EAP method which is used to authenticate. This document does not change the actual authentication with an EAP method, and all the security properties of the chosen EAP method remain. The format requirements (character encoding) and operational considerations (re-try EAP with a different EAP-Response/Identity) do not lead to new or different security properties.

7. IANA Considerations

There are no IANA actions in this document.

8. References

8.1. Normative References

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

8.2. Informative References

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC3579] Aboba, B. and P. Calhoun, "RADIUS (Remote Authentication Dial In User Service) Support For Extensible Authentication Protocol (EAP)", RFC 3579, September 2003.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.

[RFC5281] Funk, P. and S. Blake-Wilson, "Extensible Authentication Protocol Tunneled Transport Layer Security Authenticated Protocol Version 0 (EAP-TTLSv0)", RFC 5281, August 2008.

[RFC5448] Arkko, J., Lehtovirta, V., and P. Eronen, "Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA')", RFC 5448, May 2009.

[RFC6733] Fajardo, V., Arkko, J., Loughney, J., and G. Zorn, "Diameter Base Protocol", RFC 6733, October 2012.

[RFC7542] DeKok, A., "The Network Access Identifier", RFC 7542, DOI 10.17487/RFC7542, May 2015, <http://www.rfc-editor.org/info/rfc7542>.

[RFC7593] Wierenga, K., Winter, S., and T. Wolniewicz, "The eduroam Architecture for Network Roaming", RFC 7593, DOI 10.17487/RFC7593, September 2015, <http://www.rfc-editor.org/info/rfc7593>.

Author's Address

Stefan Winter
Fondation RESTENA
6, rue Richard Coudenhove-Kalergi
Luxembourg 1359
LUXEMBOURG

Phone: +352 424409 1
Fax: +352 422473
EMail: stefan.winter@restena.lu
URI: http://www.restena.lu.