Using TLS in Applications
Internet-Draft
Intended status: Standards Track
Expires: January 9, 2017

D. Margolis
Google, Inc
A. Brotman
Comcast, Inc
B. Ramakrishnan
Yahoo!, Inc
J. Jones
Microsoft, Inc
M. Risher
Google, Inc
July 8, 2016

SMTP TLS Reporting
draft-ietf-uta-smtp-tlsrpt-01

Abstract

A number of protocols exist for establishing encrypted channels between SMTP Mail Transfer Agents, including STARTTLS [RFC3207], DANE [RFC6698], and SMTP MTA STS (TODO: Add ref). These protocols can fail due to misconfiguration or active attack, leading to undelivered messages or delivery over unencrypted or unauthenticated channels. This document describes a reporting mechanism and format by which sending systems can share statistics and specific information about potential failures with recipient domains. Recipient domains can then use this information to both detect potential attackers and diagnose unintentional misconfigurations.

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

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
1.1. Terminology 3
2. Related Technologies 3
3. Reporting Policy 4
3.1. Example Reporting Policy 5
3.1.1. Report using MAILTO 5
3.1.2. Report using HTTPS 5
4. Reporting Schema 5
4.1. Result Types 6
4.1.1. Success Type 6
4.1.2. Routing Failures 6
4.1.3. Negotiation Failures 6
4.1.4. Policy Failures 7
5. Report Delivery 7
6. IANA Considerations 7
7. Security Considerations 7
8. Appendix 1: JSON Report Schema 8
9. Appendix 2: Example JSON Report 10
10. Normative References 10
Authors' Addresses 11

1. Introduction

The STARTTLS extension to SMTP [RFC3207] allows SMTP clients and hosts to establish secure SMTP sessions over TLS. The protocol design is based on "Opportunistic Security" (OS) [RFC7435], which provides interoperability for clients that do not support STARTTLS but means that any attacker who can delete parts of the SMTP session (such as the "250 STARTTLS" response) or redirect the entire SMTP session (perhaps by overwriting the resolved MX record of the delivery domain) can perform a downgrade or interception attack.

Because such "downgrade attacks" are not necessarily apparent to the receiving MTA, this document defines a mechanism for sending domains to report on failures at multiple parts of the MTA-to-MTA conversation.

Recipient domains may also use the mechanisms defined by MTA-STS (TODO: Add ref) or DANE [RFC6698] to publish additional encryption and authentication requirements; this document defines a mechanism for sending domains that are compatible with MTA-STS or DANE to share success and failure statistics with recipient domains.

Specifically, this document defines a reporting schema that covers failures in routing, STARTTLS negotiation, and both DANE [RFC6698] and MTA-STS (TODO: Add ref) policy validation errors, and standard TXT record that recipient domains can use to indicate where reports in this format should be sent.

This document is intended as a companion to the specification for SMTP MTA Strict Transport Security (MTA-STS, TODO: Add ref).

1.1. Terminology

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be interpreted as described in [RFC2119].

We also define the following terms for further use in this document:

- STS Policy: A definition of the expected TLS availability and behavior, as well as the desired actions for a given domain when a sending MTA encounters different results.
- TLSRPT Policy: A policy detailing the endpoint to which sending MTAs should deliver reports.
- Policy Domain: The domain against which an STS Policy is defined.
- Sending MTA: The MTA initiating the delivery of an email message.

2. Related Technologies

- This document is intended as a companion to the specification for SMTP MTA Strict Transport Security (MTA-STS, TODO: Add ref).
- The Public Key Pinning Extension for HTTP [RFC7469] contains a JSON-based definition for reporting individual pin validation failures.

- The Domain-based Message Authentication, Reporting, and Conformance (DMARC) [RFC7489] contains an XML-based reporting format for aggregate and detailed email delivery errors.

3. Reporting Policy

SMTP TLSRPT policies are distributed via DNS from the Policy Domain's zone, as TXT records (similar to DMARC policies) under the name "_smtp_tlsrpt". For example, for the Policy Domain "example.com", the recipient's SMTP STS policy can be retrieved from "_smtp_tlsrpt.example.com".

(Future implementations may move to alternate methods of policy discovery or distribution. See the section _Future_ _Work_ for more discussion.)

Policies consist of the following directives:

- "v": This value MUST be equal to "TLSRPTv1".

- "rua": A URI specifying the endpoint to which aggregate information about policy failures should be sent (see the section _Reporting_ _Schema_ for more information). Two URI schemes are supported: "mailto" and "https".

* In the case of `https`, reports should be submitted via POST ([@!RFC2818]) to the specified URI.
* In the case of `mailto`, reports should be submitted to the specified email address. When sending failure reports via SMTP, sending MTAs MUST NOT honor SMTP STS or DANE TLSA failures.

- "ruf": Future use. (There may also be a need for enabling more detailed "forensic" reporting during initial stages of a deployment. To address this, the authors consider the possibility of an optional additional "forensic reporting mode" in which more details--such as certificate chains and MTA banners--may be reported. See the section _Future_ _Work_ for more details.)

The formal definition of the "_mta_sts" TXT record, defined using [RFC5234], is as follows:

```
sts-text-record = sts-version *WSP %x3B *WSP sts-id

sts-version     = "v" *WSP "=" *WSP %x54 %x4C %x53
                  %x52 %x50 %x54 %x76 %x31

sts-id          = "id" *WSP "=" *WSP 1*32(ALPHA / DIGIT)
```

3.1. Example Reporting Policy

3.1.1. Report using MAILTO

```
_smtp_tlsrpt.mail.example.com. IN TXT \
        "v=TLSRPTv1;rua=mailto:reports@example.com"
```

3.1.2. Report using HTTPS

```
_smtp_tlsrpt.mail.example.com. IN TXT \
        "v=TLSRPTv1; \
        rua=https://reporting.example.com/v1/tlsrpt"
```

4. Reporting Schema

Aggregate reports contain the following fields:

- Report metadata:
  - The organization responsible for the report
    - Contact information for one or more responsible parties for the contents of the report
  - A unique identifier for the report
  - The reporting date range for the report
- Policy, consisting of:
  - One of the following policy types:
    - The SMTP MTA STS policy applied (as a string)
    - The DANE TLSA record applied (as a string) * The literal string "no-policy-found", if neither a TLSA nor

      MTA-STS policy could be found.
  - The domain for which the policy is applied
  - The MX host
  - An identifier for the policy (where applicable)
- Aggregate counts, comprising result type, sending MTA IP, receiving MTA hostname, message count, and an optional additional

information field containing a URI for recipients to review further information on a failure type.

Note that the failure types are non-exclusive; an aggregate report MAY contain overlapping "counts" of failure types where a single send attempt encountered multiple errors.

4.1. Result Types

The list of result types will start with the minimal set below, and is expected to grow over time based on real-world experience. The initial set is:

4.1.1. Success Type

- "success": This indicates that the sending MTA was able to successfully negotiate a policy-compliant TLS connection, and serves to provide a "heartbeat" to receiving domains that reporting is functional and tabulating correctly.

4.1.2. Routing Failures

- "mx-mismatch": This indicates that the MX resolved for the recipient domain did not match the MX constraint specified in the policy.

- "certificate-host-mismatch": This indicates that the certificate presented by the receiving MX did not match the MX hostname.

4.1.3. Negotiation Failures

- "starttls-not-supported": This indicates that the recipient MX did not support STARTTLS.

- "invalid-certificate": This indicates that the certificate presented by the receiving MX did not validate.

- "certificate-host-mismatch": This indicates that the certificate presented did not adhere to the constraints specified in the STS or DANE policy, e.g. if the CN field did not match the hostname of the MX.

- "certificate-name-constraints-not-permitted": The certificate request contains a name that is not listed as permitted in the name constraints extension of the cert issuer.

- "certificate-name-constraints-excluded": The certificate request contains a name that is listed as excluded in the name constraints extension of the issuer.

- "expired-certificate": This indicates that the certificate has expired.

4.1.4. Policy Failures

4.1.4.1. DANE-specific Policy Failures

- "tlsa-invalid": This indicates a validation error in the TLSA record associated with a DANE policy.

- "dnssec-invalid": This indicates a failure to authenticate DNS records for a Policy Domain with a published TLSA record.

4.1.4.2. STS-specific Policy Failures

- "sts-invalid": This indicates a validation error for the overall MTA-STS policy.

- "webpki-invalid": This indicates that the MTA-STS policy could not be authenticated using PKIX validation.

5. Report Delivery

Note that, when sending failure reports via SMTP, sending MTAs MUST NOT honor SMTP STS or DANE TLSA failures.

6. IANA Considerations

There are no IANA considerations at this time.

7. Security Considerations

SMTP TLS Reporting provides transparency into misconfigurations or attempts to intercept or tamper with mail between hosts who support STARTTLS. There are several security risks presented by the existence of this reporting channel:

- Flooding of the Aggregate report URI (rua) endpoint: An attacker could flood the endpoint and prevent the receiving domain from accepting additional reports. This type of Denial-of-Service attack would limit visibility into STARTTLS failures, leaving the receiving domain blind to an ongoing attack.

- Untrusted content: An attacker could inject malicious code into the report, opening a vulnerability in the receiving domain. Implementers are advised to take precautions against evaluating the contents of the report.

- Report snooping: An attacker could create a bogus TLSRPT record to receive statistics about a domain the attacker does not own. Since an attacker able to poison DNS is already able to receive counts of SMTP connections (and, absent DANE or MTA-STS policies, actual SMTP message payloads) today, this does not present a significant new vulnerability.

8. Appendix 1: JSON Report Schema

The JSON schema is derived from the HPKP JSON schema [RFC7469] (cf. Section 3)

```
{
  "organization-name": organization-name,
  "date-range": {
    "start-datetime": date-time,
    "end-datetime": date-time
  },
  "contact-info": email-address,
  "report-id": report-id,
  "policy": {
    "policy-type": policy-type,
    "policy-string": policy-string,
    "policy-domain": domain,
    "mx-host": mx-host-pattern
  },
  "report-items": [
    {
      "result-type": result-type,
      "sending-mta-ip": ip-address,
      "receiving-mx-hostname": receiving-mx-hostname,
      "message-count": message-count,
      "additional-information": additional-info-uri
    }
  ]
}
```

JSON Report Format

- "organization-name": The name of the organization responsible for the report. It is provided as a string.

- "date-time": The date-time indicates the start- and end-times for the report range. It is provided as a string formatted according to Section 5.6, "Internet Date/Time Format", of [RFC3339].

- "email-address": The contact information for a responsible party of the report. It is provided as a string formatted according to Section 3.4.1, "Addr-Spec", of [RFC5322].

- "report-id": A unique identifier for the report. Report authors may use whatever scheme they prefer to generate a unique identifier. It is provided as a string.

- "policy-type": The type of policy that was applied by the sending domain. Presently, the only two valid choices are "tlsa" and "sts". It is provided as a string.

- "policy-string": The string serialization of the policy, whether TLSA record or STS policy. Any linefeeds from the original policy MUST be replaced with [SP]. TODO: Help with specifics.

- "domain": The Policy Domain upon which the policy was applied. For messages sent to "user@example.com" this field would contain "example.com". It is provided as a string.

- "mx-host-pattern": The pattern of MX hostnames from the applied policy. It is provided as a string, and is interpreted in the same manner as the "Checking of Wildcard Certificates" rules in Section 6.4.3 of [RFC6125].

- "result-type": A value from the _Result Types_ section above.

- "ip-address": The IP address of the sending MTA that attempted the STARTTLS connection. It is provided as a string representation of an IPv4 or IPv6 address in dot-decimal or colon-hexadecimal notation.

- "receiving-mx-hostname": The hostname of the receiving MTA MX record with which the sending MTA attempted to negotiate a STARTTLS connection.

- "message-count": The number of (attempted) messages that match the relevant "result-type" for this section.

- "additional-info-uri": An optional URI pointing to additional information around the relevant "result-type". For example, this URI might host the complete certificate chain presented during an attempted STARTTLS session.

9. Appendix 2: Example JSON Report

```
{
        "organization-name": "Company-X",
        "date-range": {
                "start-datetime": "2016-04-01T00:00:00Z",
                "end-datetime": "2016-04-01T23:59:59Z"
        },
        "contact-info": "sts-reporting@company-x.com",
        "report-id": "5065427c-23d3-47ca-b6e0-946ea0e8c4be",
        "policy": {
                "policy-type": "sts",
                "policy-string": "TODO: Add me",
                "policy-domain": "company-y.com",
                "mx-host": "*.mail.company-y.com"
        },
        "report-items": [{
                "result-type": "ExpiredCertificate",
                "sending-mta-ip": "98.136.216.25",
                "receiving-mx-hostname": "mx1.mail.company-y.com",
                "message-count": 100
        }, {
                "result-type": "StarttlsNotSupported",
                "sending-mta-ip": "98.22.33.99",
                "receiving-mx-hostname": "mx2.mail.company-y.com",
                "message-count": 200,
                "additional-information": "hxxps://reports.company-x.com/
                  report_info?id=5065427c-23d3#StarttlsNotSupported"
        }]
}
```

Example JSON report for a messages from Company-X to Company-Y, where 100 messages were attempted to Company Y servers with an expired certificate and 200 messages were attempted to Company Y servers that did not successfully respond to the STARTTLS command.

10. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, DOI 10.17487/RFC3207, February 2002, <http://www.rfc-editor.org/info/rfc3207>.

[RFC3339] Klyne, G. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, DOI 10.17487/RFC3339, July 2002, <http://www.rfc-editor.org/info/rfc3339>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <http://www.rfc-editor.org/info/rfc5234>.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <http://www.rfc-editor.org/info/rfc5322>.

[RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <http://www.rfc-editor.org/info/rfc6125>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <http://www.rfc-editor.org/info/rfc6698>.

[RFC7435] Dukhovni, V., "Opportunistic Security: Some Protection Most of the Time", RFC 7435, DOI 10.17487/RFC7435, December 2014, <http://www.rfc-editor.org/info/rfc7435>.

[RFC7469] Evans, C., Palmer, C., and R. Sleevi, "Public Key Pinning Extension for HTTP", RFC 7469, DOI 10.17487/RFC7469, April 2015, <http://www.rfc-editor.org/info/rfc7469>.

[RFC7489] Kucherawy, M., Ed. and E. Zwicky, Ed., "Domain-based Message Authentication, Reporting, and Conformance (DMARC)", RFC 7489, DOI 10.17487/RFC7489, March 2015, <http://www.rfc-editor.org/info/rfc7489>.

Authors' Addresses

Daniel Margolis
Google, Inc

Email: dmargolis (at) google.com

Alexander Brotman
Comcast, Inc

Email: alexander_brotman (at) cable.comcast (dot com)

Binu Ramakrishnan
Yahoo!, Inc

Email: rbinu (at) yahoo-inc (dot com)

Janet Jones
Microsoft, Inc

Email: janet.jones (at) microsoft (dot com)

Mark Risher
Google, Inc

Email: risher (at) google (dot com)