Network Working Group   E. Foudil
Internet-Draft   December 03, 2017
Intended status: Informational
Expires: June 6, 2018

A Method for Web Security Policies
draft-foudil-securitytxt-01

Abstract

When security risks in web services are discovered by independent security researchers who understand the severity of the risk, they often lack the channels to properly disclose them. As a result, security issues may be left unreported. security.txt defines a standard to help organizations define the process for security researchers to securely disclose security vulnerabilities.

Status of This Memo

This Internet-Draft will expire on June 6, 2018.

Copyright Notice

Table of Contents

1. Introduction

1.1. Motivation

Many security researchers encounter situations where they are unable to responsibly disclose security issues to companies because there is no course of action laid out. security.txt is designed to help assist in this process by making it easier for companies to designate the preferred steps for researchers to take when trying to reach out.

As per section 4 of [RFC2142], there is an existing convention of using the SECURITY@domain [1] email address for communications regarding security issues. That convention provides only a single, email-based channel of communication for security issues per domain,

and does not provide a way for domain owners to publish information about their security disclosure policies.

In this document, we propose a richer, machine-parsable and more extensible way for companies to communicate information about their security disclosure policies, which is not limited to email and also allows for additional features such as encryption.

1.2. Terminology

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

2. The Specification

security.txt is a text file that should be located under the /.well-known/ path ("/.well-known/security.txt") [RFC5785] for web properties. For file systems and version control repositories a .security.txt file should be placed in the root directory. This text file contains 4 directives with different values. The "directive" is the first part of a field all the way up to the colon ("Contact:"). Directives are case-insensitive. The "value" comes after the directive ("https://example.com/security"). A "field" always consists of a directive and a value ("Contact: https://example.com/security"). A security.txt file can have an unlimited number of fields. It is important to note that you need a separate line for every field. One MUST NOT chain multiple values for a single directive. Everything MUST be in a separate field.

A security.txt file only applies to the domain, subdomain, IPv4 or IPv6 address it is located in.

```
# The following only applies to example.com.
https://example.com/.well-known/security.txt

# This only applies to subdomain.example.com.
https://subdomain.example.com/.well-known/security.txt

# This security.txt file applies to 192.0.2.0.
http://192.0.2.0/.well-known/security.txt
```

2.1. Comments

Comments can be added using the # symbol:

```
<CODE BEGINS>
# This is a comment.
<CODE ENDS>
```

You MAY use one or more comments as descriptive text immediately before the field. Parsers can then associate the comments with the respective field.

2.2. Separate Fields

A separate line is required for every new value and field. You MUST NOT chain everything in to a single field. Every line must end with a line feed character (%x0A).

2.3. Contact:

Add an address that researchers MAY use for reporting security issues. The value can be an email address, a phone number and/or a security page with more information. The "Contact:" directive MUST always be present in a security.txt file. URIs SHOULD be loaded over HTTPS. Security email addresses SHOULD use the conventions defined in section 4 of [RFC2142], but there is no requirement for this directive to be an email address.

The precedence is in listed order. The first field is the preferred method of contact. In the example below, the e-mail address is the preferred method of contact.

```
<CODE BEGINS>
Contact: security@example.com
Contact: +1-201-555-0123
Contact: https://example.com/security
<CODE ENDS>
```

2.4. Encryption:

This directive allows you to add your key for encrypted communication. You MUST NOT directly add your key. The value MUST be a link to a page which contains your key. Keys SHOULD be loaded over HTTPS.

```
<CODE BEGINS>
Encryption: https://example.com/pgp-key.txt
<CODE ENDS>
```

2.5. Signature:

In order to ensure the authenticty of the security.txt file one SHOULD use the "Signature:" directive, which allows you to link to an external signature or to directly include the signature in the file. External signature files should be named "security.txt.sig" and also be placed under the /.well-known/ path.

Here is an example of an external signature file.

```
<CODE BEGINS>
Signature: https://example.com/.well-known/security.txt.sig
<CODE ENDS>
```

Here is an example inline signature.

```
<CODE BEGINS>
Signature:
-----BEGIN PGP SIGNATURE-----

...
-----END PGP SIGNATURE-----
<CODE ENDS>
```

2.6. Acknowledgement:

This directive allows you to link to a page where security researchers are recognized for their reports. The page should list individuals or companies that disclosed security vulnerabilities and worked with you to remediate the issue.

```
<CODE BEGINS>
Acknowledgement: https://example.com/hall-of-fame.html
<CODE ENDS>
```

Example security acknowledgements page:

We would like to thank the following researchers:

```
(2017-04-15) Frank Denis - Reflected cross-site scripting
(2017-01-02) Alice Quinn  - SQL injection
(2016-12-24) John Buchner - Stored cross-site scripting
(2016-06-10) Anna Richmond - A server configuration issue
```

2.7. Example

```
<CODE BEGINS>
# Our security address
Contact: security@example.com

# Our PGP key
Encryption: https://example.com/pgp-key.txt

# Our security acknowledgements page
Acknowledgement: https://example.com/hall-of-fame.html

# Verify this security.txt file
Signature: https://example.com/.well-known/security.txt.sig
<CODE ENDS>
```

3. Location of the security.txt file

3.1. Web-based services

Web-based services SHOULD place the security.txt file under the /.well-known/ path; e.g. https://example.com/.well-known/security.txt.

3.2. File systems

File systems SHOULD place the security.txt file under the root directory; e.g. /.security.txt, C:.security.txt.

```
<CODE BEGINS>
.
&#9500;&#9472;&#9472; .security.txt
&#9500;&#9472;&#9472; example-directory-1
&#9500;&#9472;&#9472; example-directory-2
&#9500;&#9472;&#9472; example-directory-3
&#9492;&#9472;&#9472; example-file
<CODE ENDS>
```

3.3. Internal hosts

A .security.txt file SHOULD be placed in the root directory of an internal host to trigger incident response.

3.4. Extensibility

Like many other formats and protocols, this format may need to be extended over time to fit the ever-changing landscape of the Internet. Therefore, extensibility is provided via an IANA registry

for headers fields as defined in Section 6.2. Any fields registered via that process MUST be considered optional. In order to encourage extensibility and interoperability, implementors MUST ignore any fields they do not explicitly support.

4. File Format Description

The expected file format of the security.txt file is plain text as defined in section 4.1.3 of [RFC2046] and encoded in UTF-8.

The following is an ABNF definition of the security.txt format, using the conventions defined in [RFC5234].

```
body = *line (contact-field eol) *line

line = *1(field / comment) eol

eol = *WSP [CR] LF

field = contact-field / encryption-field / acknowledgement-field /
ext-field

fs = ":"

comment = "#" *(WSP / VCHAR / %xA0-E007F)

contact-field = "Contact" fs SP (email / uri / phone)

email = <Email address as per [RFC5322]>

phone = "+" *1(DIGIT / "-" / "(" / ")" / SP)

uri = <URI as per [RFC3986]>

encryption-field = "Encryption" fs SP uri

acknowledgement-field = "Acknowledgement" fs SP uri

ext-field = field-name fs SP unstructured

field-name = <as per section 3.6.8 of [RFC5322]>

unstructured = <as per section 3.2.5 of [RFC5322]>
```

"ext-field" refers to extension fields, which are discussed in Section 3.4

5. Security considerations

Companies creating security.txt files will need to take several security-related issues into consideration. These include exposure of sensitive information and attacks where limited access to a server could grant the ability to modify the contents of the security.txt file or affect how it is served.

As stated in Section 2.4, keys specified using the "Encryption:" directive SHOULD be loaded over HTTPS.

To ensure the authenticity of the security.txt file one should sign the file and include the signature using the "Signature:" directive.

6. IANA Considerations

example.com is used in this document following the uses indicated in [RFC2606].

192.0.2.0 is used in this document following the uses indicated in [RFC5735].

6.1. Well-Known URIs registry

The "Well-Known URIs" registry should be updated with the following additional value (using the template from [RFC5785]):

URI suffix: security.txt

Change controller: IETF

Specification document(s): this document

6.2. Registry for security.txt Header Fields

IANA is requested to create the "security.txt Header Fields" registry in accordance with [RFC8126]. This registry will contain header fields for use in security.txt files, defined by this specification.

New registrations or updates MUST be published in accordance with the "Specification Required" guidelines as described in section 4.6 of [RFC8126]. Any new field thus registered is considered optional by this specification unless a new version of this specification is published.

New registrations and updates MUST contain the following information:

1. Name of the field being registered or updated

2. Short description of the field

3. Whether the field can appear more than once

4. The document in which the specification of the field is published

5. New or updated status, which MUST be one of: current: The field is in current use deprecated: The field is in current use but its use is discouraged historic: The field is no longer in current use

An update may make a notation on an existing registration indicating that a registered field is historic or deprecated if appropriate.

The initial registry contains these values:

Field Name: Acknowledgment
Description: link to page where security researchers are recognized
Multiple Appearances: Yes
Published in: this document
Status: current

Field Name: Contact
Description: contact information to use for reporting security issues
Multiple Appearances: Yes
Published in: this document
Status: current

Field Name: Encryption
Description: link to a key to be used for encrypted communication
Multiple Appearances: Yes
Published in: this document
Status: current

Field Name: Signature
Description: signature used to verify the authenticity of the file
Multiple Appearances: No
Published in: this document
Status: current

7. Contributors

The editor would like to acknowledge the help provided during the development of this document by the following individuals:

- Tom Hudson helped writing the "File Format Description" and wrote several security.txt parsers.

- Joel Margolis was a big help when it came to wording this document appropriately.

- Jobert Abma for raising issues and concerns that might arise when using certain directives.

- Gerben Janssen van Doorn for reviewing this document multiple times.

- Austin Heap for helping improve the Internet drafts.

- Justin Calmus was always there to answer questions related to writing this document.

- Casey Ellis had several ideas related to security.txt that helped shape security.txt itself.

8. References

8.1. Normative References

[RFC2046] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, DOI 10.17487/RFC2046, November 1996, <https://www.rfc-editor.org/info/rfc2046>.

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

[RFC2142] Crocker, D., "Mailbox Names for Common Services, Roles and Functions", RFC 2142, DOI 10.17487/RFC2142, May 1997, <https://www.rfc-editor.org/info/rfc2142>.

[RFC2606] Eastlake 3rd, D. and A. Panitz, "Reserved Top Level DNS Names", BCP 32, RFC 2606, DOI 10.17487/RFC2606, June 1999, <https://www.rfc-editor.org/info/rfc2606>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

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

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

[RFC5735] Cotton, M. and L. Vegoda, "Special Use IPv4 Addresses", RFC 5735, DOI 10.17487/RFC5735, January 2010, <https://www.rfc-editor.org/info/rfc5735>.

[RFC5785] Nottingham, M. and E. Hammer-Lahav, "Defining Well-Known Uniform Resource Identifiers (URIs)", RFC 5785, DOI 10.17487/RFC5785, April 2010, <https://www.rfc-editor.org/info/rfc5785>.

8.2. Informative References

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

8.3. URIs

[1] mailto:SECURITY@domain

Author's Address

Edwin Foudil

Email: contact@edoverflow.com