Network Working Group
Internet-Draft
Intended status: Informational
Expires: February 20, 2018

E. Foudil

September 10, 2017

A Method for Web Security Policies
draft-foudil-securitytxt-00

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 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 February 20, 2018.

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.

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 BCP 14, RFC 2119 [RFC2119].

2. The Specification

Security.txt is a text file located in the website’s top-level 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 ("In-scope:"). Directives are case-insensitive. The "value" comes after the directive ("example.com"). A "field" always consists of a directive and a value ("In-scope: example.com"). 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 application it is located in.

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.

```
<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 PGP 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. Disclosure:

Specify your disclosure policy. This directive MUST be a disclosure type. The "Full" value stands for full disclosure, "Partial" for partial disclosure and "None" means you do not want to disclose reports after the issue has been resolved. The presence of a disclosure field is NOT permission to disclose vulnerabilities and explicit permission MUST be saught where possible.

```
<CODE BEGINS>
Disclosure: Full
<CODE ENDS>
```

2.6. Acknowledgement:

This directive allows you to link to a page where security researchers are recognized for their reports.

```
Acknowledgement: https://example.com/hall-of-fame.html
```

2.7. Example

```
<CODE BEGINS>
# Our security address

Contact: security@example.com
Encryption: https://example.com/pgp-key.txt
Disclosure: Full
<CODE ENDS>
```

3. File Format Description

The expected file format of the security.txt file is plain text 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 /
                        disclosure-field /
                        acknowledgement-field

fs                    = ":"

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

contact-field         = "Contact" fs SP (email / uri / phone)
email                 = <Email address as per RFC 5322>
phone                 = "+" *1(DIGIT / "-" / "(" / ")" / SP)
uri                   = <URI as per RFC 3986>

encryption-field      = "Encryption" fs SP uri

disclosure-field      = "Disclosure" fs SP disclosure
disclosure            = "Full" / "Partial" / "None"

acknowledgement-field = "Acknowledgement" fs SP uri
```

4. 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.

5. IANA Considerations

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

6. 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.

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.

7. Normative References

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

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

[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>.

[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>.

Author’s Address

Edwin Foudil
Email: contact@edoverflow.com