Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 2, 2018

E. Foudil

January 29, 2018

The Security Policy Specification Standard
draft-foudil-spss-00

Abstract

This document proposes a way of standardising the structure, language, and grammar used in security policies. The goal is to reduce ambiguity and confusion that stems from poorly-worded security policies. Organisations and individuals can refer back to this document if their security policy uses definitions found in this document.

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 August 2, 2018.

Copyright Notice

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. The Specification

1.1. Disclosure Policy

The "Disclosure policy" refers to the section of the security policy where the vendor describes how someone can report a security issue. Vendors SHOULD use the SECURITY@domain email address for communications regarding security issues as per section 4 of [RFC2142] and set up a security.txt file pointing to the security policy as per [draft-foudil-securitytxt] [1]. This section also establishes a safe harbour where the vendor declares that they are ready to investigate legitimate reports and not take legal action against the reporter if the reporter abides by the vendor's security policy.

Example

> You can report security vulnerabilities to security@example.com. We will investigate legitimate reports and make every effort to quickly resolve any vulnerability. To encourage reporting security vulnerabilities directly to us, we will not take legal action against you nor ask law enforcement to investigate you providing you comply with the following guideline:

* Make a good faith effort to avoid privacy violations, destruction of data, and interruption or degradation of my services.

1.2. Service-level agreement (Performance expectations)

Since the disclosure process should be coordinated, the security researcher will want to know what to expect from the vendor, especially when it comes to the duration of the entire disclosure process. A service-level agreement SHOULD include the expected time to:

o First response

o Reward (If the vendor offers any rewards - see "Rewards" section)

o Resolution

Example

We will make a best effort to meet the following expectations for security researchers participating in this security program:

* Time to first response: 2 business days or less.

* Time to triage: 3 business days or less.

* Time to reward: 3 business days or less.

* Time to resolution: 30 business days or less.

1.3. Scope

This section will define the grammar to be used when defining a scope; this becomes especially useful when the vendor wants to encourage security researchers to inspect the security of a product and report by back with any findings.

In scope

The term "in scope" refers to any assets that belong to the vendor and are within the security policy's boundary. Security researchers are instructed to look for security issues in in-scope assets.

Out of scope

"Out of scope" is the opposite of "in scope" and refers to any assets that do not belong to the vendor, and are not within the security policy's boundary. Security researchers SHOULD not look for issues nor report security issues located in out-of-scope assets.

Grammar

To make things very clear to the security researcher it is important to have a clear and well-designed scope; one way of achieving this is to have a standardised and detailed language to describe the asset in question.

Symbol	Definition
"*"	"All", as in any possible value.
"[80, 443]"	The set containing the values 80 and 443.
"[0-10]"	A range. In this example a range from 0 to 10.
"+"	In scope. This symbol can be substituted with simply placing the definitions under an "In scope" section.
"- "	Out of scope. This symbol can be substituted with simply placing the definitions under an "Out of scope" section.
"app:"	This directive refers to a native application. This allows one to link to a digital distribution service and make it clear that the native application is what is being referred to and not the actual page where you can download the native application.
"mailto:"	Refers to an email address as defined in [RFC6068]

Usage

The symbols defined above can be used as follows:

Example	Meaning
"http://example.com/"	The top-level directory of example.com on port

	80.
"*://*.example.com:*/*"	All protocols, subdomains, ports and pages with the example.com basename.
"example.com:[80, 443]/*"	All pages on ports 80 and 443 on the example.com basename.
"example.com:[22-443]/*"	All pages on ports 22 to 443 on the example.com basename.
"http://192.0.2.0/"	The top-level directory of the 192.0.2.0 IPv4 address on port 80.
"192.0.2.0/24"	CIDR notation to describe the IPv4 range from 192.0.2.0 to 192.0.2.255.
"2001:db8::/48"	CIDR notation to describe the IPv6 range from 2001:db8:0:0:0:0:0:0 to 2001:db8:0:ffff:ffff:ffff:ffff.
"192.0.2.*"	All possible values within the last octet, ranging from 192.0.2.0 to 192.0.2.255
"app:com.example.android"	The _com.example.android_ Android application.
"app:https://play.google.com/store/apps/details?id=com.example.android"	The _com.example.android_ Android application and not "https://play.google.com/".

"+ http://example.com/"	The top-level directory of example.com on port 80 is in scope.
"- http://example.com/"	The top-level directory of example.com on port 80 is out of scope.
"- *"	Everything that is not defined in the "In scope" section and not using the "+" symbol, is not in scope. The opposite does not exist ("+ *").
"mailto:contact@example.com"	The "contact@example.com" email address.

Example

```
 + *://*.example.com:*/*
 + test.another-example.com:[80, 443]/*
 + https://*.test.another-example.com/*
 + http://pub.another-example.com:[22-443]/*
 + app:https://play.google.com/store/apps/details?id=com.example.android
 - *
```

1.4. Exclusions

1.4.1. Excluded test types

Excluded test types define what methods for discovering security issues a security researcher is not permitted to use.

Example

> Findings from physical testing such as office access are strictly prohibited.

1.4.2. Excluded issue types

Excluded issue types are types of security issues that the vendor does not want security researchers to report. To prevent confusion the vendor SHOULD structure each exclusion in this section as a description of the issue type followed by the reason why the vendor does not want to receive reports concerning that type of issue.

Example

> We do not want researchers to report Cross-site Request Forgery (CSRF) with minimal security implications such as logout CSRF to us. In order for CSRF to be a valid issue, it must affect some important action such as deleting one's account.

1.5. Proof of concepts

The "Proof of concepts" section describes what the vendor wants the security researcher to demonstrate in their proof of concept. This section should also set boundaries to ensure that the security researchers know how far they have to escalate their finding to demonstrate the issue.

Example

Google's Vulnerability Reward Program states the following [2]:

> [...] while alert(1) is the standard way of confirming that your attempt to inject JavaScript code into a web application succeeded in some way, it does not tell you where that injection happened, exactly. That's particularly important for Google services because of our use of sandboxed domains to safely render some of the content we get from our users or retrieve from the Internet. So, we always recommend our reporters to try alert(document.domain) instead.

Encouraging security researchers to use alert(document.domain) in their proof of concept allows Google and the security researcher to quickly determine if the finding is a valid issue.

1.6. Terminology

The term "severity" is frequently used interchangeably with "impact" or "priority". This section defines some basic terminology in order to prevent any potential confusion.

Severity (Oxford Dictionaries' definition [3])

The fact or condition of being severe.

Overall severity

The overall score determined using a vulnerability scoring system such as the Common Vulnerability Scoring System [4].

Impact (Information Technology Infrastructure Library's definition [5])

A measure of the effect of an incident, problem or change on business processes. Impact is often based on how service levels will be affected. Impact and urgency are used to assign priority.

Priority (Information Technology Infrastructure Library's definition [6])

A category used to identify the relative importance of an incident, problem or change. Priority is based on impact and urgency, and is used to identify required times for actions to be taken.

1.7. Rewards

Some vendors reward security researchers for reporting security issues either in form of a public acknowledgment, prizes -- often referred to as "swag" -- and/or payments, so called "bounties". Financial rewards (bounties) SHOULD be tied to the overall severity of the reported security issue and not the security issue type.

Example

This is an example reward table where the bounty amounts are tied to overall severity scores calculated using the Common Vulnerability Scoring System.

CVSS Score	Bounty Amount in $
5	$50
6	$86
7	$137
8	$205
9	$290
10	$400

2. Security considerations

Organizations creating a security policy will need to consider several security-related issues. These include exposure to sensitive information and attacks where limited access to a server could grant the ability to modify the contents of the policy or affect how it is served.

3. References

3.1. Normative References

[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>.

[RFC6068] Duerst, M., Masinter, L., and J. Zawinski, "The 'mailto' URI Scheme", RFC 6068, DOI 10.17487/RFC6068, October 2010, <https://www.rfc-editor.org/info/rfc6068>.

3.2. URIs

[1] https://tools.ietf.org/html/draft-foudil-securitytxt-02

[2] https://sites.google.com/site/bughunteruniversity/improve/alert-1-considered-harmful

[3] https://www.oxforddictionaries.com/

[4] https://www.first.org/cvss/

[5] https://www.axelos.com/corporate/media/files/glossaries/itil_2011_glossary_gb-v1-0.pdf

[6] https://www.axelos.com/corporate/media/files/glossaries/itil_2011_glossary_gb-v1-0.pdf

Author's Address

Edwin Foudil

Email: contact@edoverflow.com