Incorrect handling of duplicate HTTP headers

High **mattklein123** published **GHSA-2v25-cjjq-5f4w** on Sep 29, 2020

Package	
Envoy	
Affected versions	Patched versions
1.15.0, 1.14.4, 1.13.4, 1.12.6 and before.	1.15.1, 1.14.5, 1.13.5, 1.12.7

Description

Brief description

Envoy through 1.15.0 only considers the first value when multiple header values are present for some HTTP headers.
Envoy's setCopy() header map API does not replace all existing occurences of a non-inline header.

CVSS

CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:L/A:L
(8.3, High)

Affected version(s)

Envoy 1.15.0, 1.14.4, 1.13.4, 1.12.6 and before.

Affected component(s)

Request routing via header matching, access log filtering via header matching, RBAC, CEL matching, ext_authz filter, header to metadata filter, JWT filter, Lua filter.

Attack vector(s)

Attacker includes multiple values of "reference" headers (i.e. headers that Envoy has not explicitly listed as "inline"), e.g. `x-foo: bar` and `x-foo: baz` .

Discover(s)/Credits

Harvey Tuch, Mark D. Roth (Google LLC), Matt Klein (Lyft)

Details

Envoy's HeaderMapImpl has two classes of headers, *inline* and *non-inline*. Inline headers treat headers with multiple inline values as a comma-delimited string, as per RFC7230. Non-inline headers have ordered multi-map semantics. Components such as `HeaderUtility::matchHeaders()` that use the `get()` method only observe the first header value.

Attackers can set multiple values of a non-inline header, e.g. `x-foo: bar` and `x-foo: baz` . Affected Envoy components will only observe the first value, `x-foo: bar` , in matchers, but both `x-foo: bar` and `x-foo: baz` will be forwarded to upstreams. Upstreams may take both values into consideration, resulting in an inconsistency between Envoy's request matching and the upstream view of the request.

An example of how this might be exploited is if an Envoy filter validates both `x-request-credential` and `x-request-scope` headers, checking whether the given credentials match the request scopes. Only the first scope header would be validated, but the upstream may interpret all provided scopes as being valid.

Additionally, Envoy's header map implementation has a `setCopy()` API which replaces all existing occurrences of a header with a new value. Previously, when calling `setCopy()` on a non-inline header, Envoy would only replace the first value. This would allow other headers to pass to upstream unmodified, thus causing inconsistency between the upstream and Envoy's view of the request. In particular this API is used by the extauth filter. As part of this CVE release, `setCopy()` now consistently replaces all occurrences of a header with a single value.

References

- CVE: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-25017

Severity

High

CVE ID

CVE-2020-25017

Weaknesses

No CWEs

Credits

htuch
mattklein123
markdroth