CWE-113: Improper Neutralization of CRLF Sequences in HTTP Headers ('HTTP Request Header Injection')

Moderate graemerocher published GHSA-694p-xrhg-x3wm on Mar 30, 2020 · 1 comment

Package

io.micronaut:micronaut-http-client (Maven)

Affected versions	Patched versions
<= 1.2.10, 1.3.1	1.2.11, 1.3.2

Description

Vulnerability

Micronaut's HTTP client is vulnerable to "HTTP Request Header Injection" due to not validating request headers passed to the client.

Example of vulnerable code:

```
@Controller("/hello")
public class HelloController {

    @Inject
    @Client("/")
    RxHttpClient client;

    @Get("/external-exploit")
    @Produces(MediaType.TEXT_PLAIN)
    public String externalExploit(@QueryValue("header-value") String headerValue) {
        return client.toBlocking().retrieve(
            HttpRequest.GET("/hello")
                .header("Test", headerValue)
        );
    }
}
```

In the above case a query value received from a user is passed as a header value to the client. Since the client doesn't validate the header value the request headers and body have the potential to be manipulated.

For example, a user that supplies the following payload, can force the client to make multiple attacker-controlled HTTP requests.

```
List<String> headerData = List.of(
    "Connection: Keep-Alive", // This keeps the connection open so another request can be stuffed in.
    "",
    "",
    "POST /hello/super-secret HTTP/1.1",
    "Host: 127.0.0.1",
    "Content-Length: 31",
    "",
    "{\"new\":\"json\",\"content\":\"here\"}",
    "",
    ""
);
String headerValue = "H\r\n" + String.join("\r\n", headerData);;
URI theURI =
    UriBuilder
        .of("/hello/external-exploit")
        .queryParam("header-value", headerValue) // Automatically URL encodes data
        .build();
HttpRequest<String> request = HttpRequest.GET(theURI);
String body = client.toBlocking().retrieve(request);
```

Note that using `@HeaderValue` instead of `@QueryValue` is not vulnerable since Micronaut's HTTP server does validate the headers passed to the server, so the exploit can only be triggered by using user data that is not an HTTP header (query values, form data etc.).

Impact

The attacker is able to control the entirety of the HTTP body for their custom requests.
As such, this vulnerability enables attackers to perform a variant of Server Side Request Forgery.

Patches

The problem has been patched in the `micronaut-http-client` versions 1.2.11 and 1.3.2 and above.

Workarounds

Do not pass user data directly received from HTTP request parameters as headers in the HTTP client.

References

Fix commits

- 9d1eff5
- 6deb60b
- bc855e4

For more information

If you have any questions or comments about this advisory:

- Open an issue in micronaut-core
- Email us at info@micronaut.io

Credit

Originally reported by **@JLLeitschuh**

Severity

Moderate

CVE ID

CVE-2020-7611

Weaknesses

No CWEs

Credits

JLLeitschuh