Exposure of Sensitive Information to an Unauthorized Actor in httpie

Low **isidentical** published **GHSA-9w4w-cpc8-h2fq** on Mar 7

Package

httpie (pip)

Affected versions	Patched versions
< 3.1.0	3.1.0

Description

Impact

HTTPie have the practical concept of sessions, which help users to persistently store some of the state that belongs to the outgoing requests and incoming responses on the disk for further usage. As an example, we can make an authenticated request and save it to a named session called `api`:

```
$ http --session api -a user:pass pie.dev/basic-auth/user/pass
```

```
{
    "authenticated": true,
    "user": "user"
}
```

Since we have now saved the authentication data to that session, we won't have to enter it again and again on every invocation. We can simply reference the session, and HTTPie will use the saved state directly from it:

```
$ http --session api pie.dev/basic-auth/user/pass
```

```
{
    "authenticated": true,
```

```
    "user": "user"
}
```

One particular use case of these sessions is storing cookies (commonly referred to as a `Cookie Jar`). If a response has a `Set-Cookie` header, HTTPie will parse it and store the actual cookie in the session. And from that point on, all outgoing requests will attach that cookie (in the form of a `Cookie` header).

This is extremely useful, especially when you are dealing with websites which manage their own state on the client-side through cookies.

```
$ http -F --session jar pie.dev/cookies/set/x/y
```

```
{
    "cookies": {
        "x": "y"
    }
}
```

Before `3.1.0`, HTTPie didn't distinguish between cookies and hosts they belonged. This behavior resulted in the exposure of some cookies when there are redirects originating from the actual host to a third party website, e.g:

```
$ http -F --session jar pie.dev/redirect-to url==https://httpbin.org/cookies
```

(Pre 3.1.0)

```
{
    "cookies": {
        "x": "y"
    }
}
```

(Post 3.1.0)

```
{
    "cookies": {}
}
```

This behavior has been corrected in this release (with taking RFC 6265 — HTTP State Management Mechanism into the consideration).

A huge credit goes to @Glyph for disclosing the original vulnerability to us (through huntr.dev).

Patches

We suggest users to upgrade their HTTPie version to `3.1.0` or higher, and run `httpie cli sessions upgrade` command on their sessions.

For more information

If you have any questions or comments about this advisory:

- Email us: `security@httpie.io`

> Please note that this entry is covered by both CVE-2022-24737 and CVE-2022-0430.

Severity

Low

CVE ID

CVE-2022-24737

Weaknesses

CWE-200