CVE-2022-37434
zlib inflate function causes out-of-bounds write

109550060 陳星宇

What is CVE-2022-37434?

- zlib through 1.2.12 has a heap-based buffer over-read or buffer overflow in inflate in inflate.c via a large gzip header extra field.
- only applications that call inflateGetHeader are affected.

How is the CVE discovered?

- An issue in Github with curl tag turned out a zlib CVE issue.
- Test224 in Curl test suite send a HTTP request to a server with content that wants to be encoded accompanied with large extra header field. It should be ignored and refuse to compress the content if the extra header field exceeds max length. However, the developer didn’t do the error check for the header length. This will result in heap overflow and undesired behavior of writing since the compressed content will be “memcpied” into a out-of-range memory.

CVSS scores and scoring vectors

Severity level is medium in Red Hat and severe in NVD.

As Red Hat stated that zlib functions are bundled in their products, it is unable to be utilized by the attacker, which leads to low confidentiality and integrity risks.

CVSS v3 Score Breakdown

	Red Hat	NVD
CVSS v3 Base Score	7.0	9.8
Attack Vector	Network	Network
Attack Complexity	High	Low
Privileges Required	None	None
User Interaction	None	None
Scope	Unchanged	Unchanged
Confidentiality Impact	Low	High
Integrity Impact	Low	High
Availability Impact	High	High

Reproduce the CVE environment

Since only the V1.2.2 patched zlib is vulnerable with the attack, we have to prepare a server with v1.2.2 zlib first, then prepare another client that uses curl or other http-based tools to send a request for gzip content by including “accept-encoding: gzip” header.

Reproduce the exploitation

- Open two VM, one is Debian with apache/nginx server(victim) that accepts the content and then compress the data into gzip file to trigger the CVE exploitation. Another is an arbitrary Linux host served as http client(attacker), and try to embed large extra header into the content.