FFmpeg HLS vulnerability

FFmpeg is an open source software used for processing audio and video formats. You can use a malicious HLS playlist inside an AVI video to read arbitrary files.

Exploits

```
1. `./gen_xbin_avi.py file://<filename> file_read.avi`
2. Upload `file_read.avi` to some website that processes videofiles
3. (on server side, done by the videoservice) `ffmpeg -i file_read.avi output.mp4`
4. Click "Play" in the videoservice.
5. If you are lucky, you'll the content of `<filename>` from the server.
```

How it works (Explanations from neex - Hackerone links)

the script creates an AVI that contains an HLS playlist inside GAB2. The playlist generated by this script looks like this:

```
#EXTM3U
#EXT-X-MEDIA-SEQUENCE:0
#EXTINF:1.0
GOD.txt
#EXTINF:1.0
/etc/passwd
#EXT-X-ENDLIST
```

To process a playlist ffmpeg concatenates all segments and processes it as single file. To determine the type of this file FFmpeg uses the first segment of the playlist. FFmpeg processes .txt files in a special way. It tries to show a screen capture of a tty printing this file.

So, the playlist above will be processed as follows: FFmpeg sees #EXTM3U signature inside GAB2 chunk and determines file type as HLS playlist. The file GOD.txt doesn't even exist, but it's name is enough for FFmpeg to detect file type as .txt. FFmpeg concatenates the contents of all segments of the playlist. As only one of two segments actually exists, the result of concatenation is just the contents of the file we want to retrieve. Because the type of this concatenation is .txt, FFmpeg draws a tty that prints the file.

Thanks to

- Hackerone - Local File Disclosure via ffmpeg @sxcurity (https://hackerone.com/reports/242831)

- Hackerone - Another local file disclosure via ffmpeg (https://hackerone.com/reports/243470)
- PHDays - Attacks on video converters:a year later, Emil Lerner, Pavel Cheremushkin (https://docs.google.com/presentation/d/1yqWy_aE3dQNXAhW8kxMxRqtP7qMHaIfMzUDpEqFneos/edit#slide=id.p)
- Script by @neex (https://github.com/neex/ffmpeg-avi-m3u-xbin/blob/master/gen_xbin_avi.py)