Bug 58204 - Skia buffer overflow

Original Reference: https://bugzilla.kaios.tech/show_bug.cgi?id=58204

Attachments:

fix_58204.patch

Fabrice Desré [:fabrice] | 2019-03-23 00:14:52 CST

Reported in https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-7386 with the details at https://s3curityb3ast.github.io/KSA-Dev-007.md

It's been reported for the 8110 but it's not device specific and affects all gecko48 based products.

Fabrice Desré [:fabrice] | 2019-03-23 00:15:18 CST

Here's the exploit details in case this goes offline:

```
==================================================
DoS and gecko reboot in the nokia 8810 4G handset
==================================================

. contents:: Table Of Content

Overview
========

Title:- DoS and gecko reboot in the nokia 8810 4G handset
Author: Kaustubh G. Padwad
```

CVE ID: CVE-2019-7386
Vendor: HMD Global, Nokia, KaiOS
Products: Nokia 88104G

Tested Version: :
        Model :- Nokia 8810 4G
        Software : 10.05
        Kai OS Version : 2,5
        Build Number : 10.05
        Platfirm ver : 48.0.a2
Severity: High--Critical

Advisory ID
============
KSA-Dev-007

About the Product:
==================
Brand Nokia
Developer HMD Global
Manufacturer Foxconn
Operating System : kaios
Nokia 8110 4G is a Nokia-branded mobile phone developed by HMD Global. It was announced on 25 February 2018 at Mobile World Congress (MWC) 2018 in Barcelona, Spain, as a revival of the original Nokia 8110, which was popularly known as the "Matrix phone" or "banana phone". It runs on an operating system based on KaiOS, and through the company's partnership with Google also features Google services like Maps and Assistant.

Description:
============

A Denial of Service issue has been discovered in the Gecko component of KaiOS 2.5 10.05 (platform 48.0.a2) on Nokia 8810 4G devices. When a crafted web page is visited with the internal browser, the Gecko process crashes with a segfault. Successful exploitation could lead to the remote code
 execution on the device.

Affected Product Code Base

Nokia 8810 4G - Software : 10.05 , Kai OS Version : 2,5 ,Build Number : 10.05 ,Platform ver : 48.0.a2

Vulnerability Class:
====================
Buffer Overflow

Attack Type
==========
Remote

Impact Denial of Service
========================
 true

Attack Vectors
==============
To exploit this vulnerability one needs to visit the crafted webpage using inbuilt browser in the device

Affected Component
the Denial of Service issue has been discovered in the the gecko component of the KaiOS used in Nokia 8810 4G, When crafted web page is
visited by internal browser of Nokia the gecko process crash with segfault

How to Reproduce: (POC):
========================
1. Host the webpage with below contain on the controlled server Eg. 192.168.1.1 as crash.html.

```
<!DOCTYPE html>
<html>
<body>
  <canvas id="canvas" width="500", height="500"> </canvas>
  <script>
  var canvas = document.getElementById("canvas");
```

```
  var width = canvas.width;
  var height = canvas.height;
  for (var x=0; x < 400; x++){
    var ctx = canvas.getContext("2d");
    for (var i = 0; i < width; i += 10) {
      ctx.moveTo(i, 0);
      ctx.lineTo(i, height);
      ctx.stroke();
    }
  }
  </script>
</body>
</html>
```

2. Now visit the url http://192.168.1.1/crash.html using the inbuilt browser.

3. As soon as page render it cause the buffer overflow in skiaGL component of the gecko and cause gecko to reboot.

Mitigation
==========

Not Available

Disclosure:
===========
01-JAN-2019 Discoverd the Vulnerability
01-jan-2018 Reported to Nokia
02-JAN-2019 Nokia ask to report to the HMD Global,
02-JAN-2019 Reported to HMD Global using info@hmdglobal.com (No Repsonse)
04-JAN-2019 twitted to them (No Reposne)
05-Jan-2019 Requested For CVE-ID
04-FEB-2019: CVE Assigned

credits:
========
* Kaustubh Padwad
* Information Security Researcher
* kingkaustubh@me.com

```
*
* https://twitter.com/s3curityb3ast
* http://breakthesec.com
* https://www.linkedin.com/in/kaustubhpadwad
```

Vincent CHANG | 2019-03-24 16:55:21 CST

Created attachment 72161 [details]
kernel and logcat log for lowmemkiller

I am able to reproduce the issue using QRD device. Looks like lowmemkiller is triggered while loading EGL related libraries.

Chrono, could you please help to check if it makes sense to consume so many memory while loading EGL libraries, any chance we can optimize it.

Vincent CHANG | 2019-03-24 16:55:55 CST

Started working on this

Chrono WU | 2019-03-27 09:49:28 CST

Update current status:
1. It's the cntx stroke operation that consumes much memory and leads to the lowmemorykiller kill app issue. Not the operation of loading OpenGL library.

2. Confirmed that the issue it's not caused by mess amount of gralloc memory allocation. So it's not related to GPU buffer, texture..., etc.

3. Continue to trace what will be done within cntx.stroke operation, it's calling Stroke of DrawTargetSkia. Will go further into it and find out the reason that it consumes so much memory.

Fabrice Desré [:fabrice] | 2019-03-27 09:50:56 CST

Can you check if that was fixed in a more recent version of skia than the one we use?

Chrono WU | 2019-03-29 17:11:03 CST

Find the reason that cant.stroke consumes so much memory. The following script generate batch draw command for the corresponding draw behavior.
index.html:L12

```
      ctx.moveTo(i, 0);
      ctx.lineTo(i, height);
```

And the script ctx.stroke() pushes the batch command to the context. The data structure used to store the batch command is SkTArray. In this case, there will be approximately 20000 draw commands that will need to be pushed into this cntx which leads to the mess memory usage and trigger LMK to kill the process.

It's not easy to find a way to fix the issue since we don't have a way to limit the memory usage in system memory, need to further investigate for the solution.

Chrono WU | 2019-04-10 16:10:08 CST

Suggest to provide limitation for canvas rendering batch count and drop the overflowed batch command.

I've also tried to flush the batches each time when the batches count reach 500. But it does no obvious improvement since the flush operation generates corresponding buffer such as vertex/texture according to the batch content. The memory usage is even more worse in this case. The better way is to limit the batch count size that can handle in GrDrawTarget::recordBatch.

Chrono WU | 2019-04-15 16:55:00 CST

Created attachment 74655 [details]
Link to GitLab merge-request:

https://git.kaiostech.com/KaiOS/gecko48/merge_requests/2084

Fabrice Desré [:fabrice] | 2019-04-16 05:24:49 CST

Was that addressed upstream in some way also?

Chrono WU | 2019-04-23 16:57:11 CST

Update some findings for 256 MB projects:
1. The issue can't be reproduced through 256 MB devices such as pier2_kk and modric_kk_256. The reason is that in 256 MB projects, SkBitmapDevice(SW rendering) is chosen rather than SkGpuDevice(GPU rendering). The rendering requests with SkBitmapDevice is directly rendered each time and that's why it would not cause memory issue.

2. Upon 1, there is no need to do special limitations in 256 MB projects.

3. The rendering speed of SkBitmapDevice is very slow compared to SkGpuDevice. It's memory vs speed trade-off.

Vincent CHANG | 2019-05-02 16:12:45 CST

Comment on attachment 74655 [details]
Link to GitLab merge-request:
https://git.kaiostech.com/KaiOS/gecko48/merge_requests/2084

Looks good to me.
Let's try to get some game apps in Kai store to do the test before merging the MR.

Vincent CHANG | 2019-05-03 10:29:29 CST

After consult with QA team, we are going to use the game BubbleShooter in below link to verify the commit. BubbleShooter uses the canvas and WebGL APIs and is also a heavy CPU and memory usage game.

https://git.kaiostech.com/itriad/bubble-shooter/wikis/releases/BubbleShooter__prod_1.0.0_por_KAI_2019-02-22.zip

Chrono WU | 2019-05-03 13:58:11 CST

Confirmed that the game BubbleShooter can be played normally with the patch applied. I just play it about 10 minutes, there is no issue occurred.

Vincent CHANG | 2019-05-06 10:29:45 CST

The MR has been merged to

https://git.kaiostech.com/KaiOS/gecko48/commit/9d32fbbb1b1fe141143def1

Seinlin LI | 2019-06-20 06:46:51 CST

*** Bug 64433 has been marked as a duplicate of this bug. ***

Seinlin LI | 2019-06-20 07:10:24 CST

(In reply to Fabrice Desré [:fabrice] from comment #5)
> Can you check if that was fixed in a more recent version of skia than the
> one we use?

According to the discussion in Moz bug, the new version of skia fixed this. It looks like they are the same issue.
https://bugzilla.mozilla.org/show_bug.cgi?id=1306890#c8

BTW, before we update the skia to the new version, the current solution of Limit skia batch seems to be reasonable.

gitlab integration | 2019-06-20 10:10:41 CST

[KaiOS/gecko48]
Pushed to v2.5:
https://git.kaiostech.com/KaiOS/gecko48/commit/147c54afbf1edcdade9d193

bug 58204 - Limit skia batch size to avoid triggering LMK for
killing APP. r=Vincent,Eastern

(cherry picked from commit
9d32fbbb1b1fe141143def11dc324b07ec6c89e3)

gitlab integration | 2019-06-20 10:12:24 CST

[KaiOS/gecko48]
Pushed to v2.5R2:
https://git.kaiostech.com/KaiOS/gecko48/commit/18625edc03861ed439959c0

bug 58204 - Limit skia batch size to avoid triggering LMK for
killing APP. r=Vincent,Eastern

(cherry picked from commit
9d32fbbb1b1fe141143def11dc324b07ec6c89e3)

gitlab integration | 2019-06-20 10:12:27 CST

[KaiOS/gecko48]
Pushed to jio_pier:
https://git.kaiostech.com/KaiOS/gecko48/commit/aebddcb899d8cc9b95678c4

bug 58204 - Limit skia batch size to avoid triggering LMK for
killing APP. r=Vincent,Eastern

(cherry picked from commit
9d32fbbb1b1fe141143def11dc324b07ec6c89e3)

gitlab integration | 2019-06-20 10:12:37 CST

[KaiOS/gecko48]
Pushed to jio_quoin:
https://git.kaiostech.com/KaiOS/gecko48/commit/46861ba2b59cf13f8a7666a

bug 58204 - Limit skia batch size to avoid triggering LMK for
killing APP. r=Vincent,Eastern

(cherry picked from commit
9d32fbbb1b1fe141143def11dc324b07ec6c89e3)