Reserved Memory Low-Memor-Killer

Broadcom Proprietary and Confidential

Release note: this document can be shared with customers.

CONTENTS

rmlmk: Reserved Memory Low-Memory-Killing

The "reserved memory" low-memory-killing is a set of procedures designed to address shortcoming of the classic low-memory-killing available in Android.

The low-memory-killer in Android may not necessarily be aware of the actual usage of the reserved memory used by the middleware integration for various reasons:

- Such memory may not actually be visible to Linux|Android and so cannot be accounted for in the classic low-memory-killer calculations
    - This is the case for 'bmem' regions defined for nexus usage. Only the middleware (nexus) can accurately determine the usage of those regions.
- Even while aware of the 'overall' memory region used, Linux|Android may sometimes not be able to determine finer grained use of the memory within a region and therefore may not be able to make appropriate decisions regarding potential life cycle of those processes associated with the memory, let alone when such memory is potentially shared between processes.
    - This is the case for the 'brcm_cma' region which is defined to be used by various middleware users but may appear as a single large allocation from a single permanent process to Linux.

For those reasons, we need to implement a set of procedures which can be **complementary** to the classic low-memory-killer and that can properly integrate with the middleware knowledge and behavior.

The mechanism must respect the architecture of Android and the basic integration of the existing low-memory-killer. In particular:

- We will always try to minimize the need for killing applications, killing applications is not a user friendly experience, therefore as much as possible we want to avoid killing applications unless strictly necessary.
- Decision on which application to kill is entirely based on the Android process classification, alongside the private memory usage knowledge of the reserved region(s) content, this means we will not actively seek to eliminate applications that are not properly marked as candidate by Android.
    - Here we make the assumption that Android properly classifies all applications, which happens to be true.
    - This also means that the purpose of the rmlmk is not (as example) to eliminate badly behaved applications that may hog the memory. As long as those are within the accepted realm of Android operations and as long as such applications are not marked as potential candidate for removal, it is not the purpose of the rmlmk to proactively change that.

Disclaimer

Memory management is a difficult process overall which requires compromises, which can be root for a lot of possible complications or outcomes, therefore it should be understood that the purpose of the rmlmk is not to solve all possible problems, but rather provide a set of well defined procedures that can be tuned as needed to achieve certain pre-determined goals.

rmlmk-v1

The version 1 of the rmlmk, also referred to as "**rmlmk-v1**" is dedicated to address low memory situations due to lack of cma (brcm_cma) based graphics memory while actively switching between activities (ie applications) in Android.

Foreground applications being placed in the background (such as when exiting an application by going back to "home" launcher) do not necessarily release their associated memory. When such applications are heavy consumers of cma memory, as example due to heavy graphics use, it is possible to end up in a situation of memory shortage for other applications appearing in the foreground that also need to allocate cma based graphics memory. The reason for such is that because overall the system is not short in memory necessarily, the classic low-memory-killer would not see the need to kick in, but the cma region used by middleware is effectively being exhausted.

The rmlmk-v1 would in this case allow reclaim of background allocated cma memory to be used by the foreground application.

The memory reclaim procedure happens in two fashions:

1. As a background process actively monitoring existing high consumers and eliminating them if suitable.
    - This process runs constantly but it has a rather high threshold to avoid unnecessary background killing of applications that do not hold back too much memory. Thus facilitating their ability to resume quickly.
2. As a more aggressive on-demand process when an active allocation fails which would contribute to a foreground application not able to startup properly.
    - This process is triggered specifically when a middleware bound cma allocation fails and we would like to give it another chance by removing non critical allocations. This process is more aggressive than the background one since we want to give the best chance to allow the new application to run.

Integration pointers: rmlmk-v1 on n-mr1-tv-dev

On "squashfs" file system

squashfs file system is inefficient in terms of performances and consumes additional memory on the linux side to allow real time decompression of the file system use. We mandate that such support be removed from n-mr1-tv-dev as part of the rmlmk-v1 integration.

The Broadcom reference devices previously making use of squashfs have been reverted to ext4 instead and the patchset listed as part of the rmlmk-v1 support includes those changes necessary to do so.

Patchset

By default, this feature is already delivered and enabled in our Android N-MR1 and later code bases. For earlier versions of the code (ie. N-MR1 prior to the delivery) the following patchset is available on the **n-mr1-tv-dev branch**, all patches are necessary.

vendor/broadcom/bcm_platform

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-
stb/vendor/broadcom/bcm_platform
```

[hash]

```
a079f41 nxserver: expand scope of memory tracker to include nxmem and bcmpip.
6b565fd nxserver: allow configuration (ro) of the cma oom lmk thresholds.
0d3a4af memtrack: shortcut invalid results generation.
b22e045 nxserver: hook up trim callback from inexus with cma oom process.
53c22ee nxif: hook up trimCamFromClientContext callback.
9a38a7c nxserver: setup oom monitoring and killing for private cma region.
```

```
a4ff1bf nxserver: hook up bcmpip to gather oom adjust scores of processes using
gfx cma.
e3e17fb nxserver: hook up gathering gfx per process stats when v3d mmu
available.
```

vendor/broadcom/refsw

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-stb/vendor/broadcom/refsw
```

[hash]

```
0a57806 memory:android: make use of the bcmpip interface to track allocations.
989aace gpu: hook up memory allocator with nexus' trim callback.
a6229ba nexus memory heap selection.
98a08db Add BEGL memory interface "Init"
```

vendor/broadcom/release_prebuilts

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-
stb/vendor/broadcom/release_prebuilts
```

[hash]

```
ceb626b prebuilts: generated new plugin for widevine following binder api
update.
12ac287 pr: update prebuilts with binder api update
```

vendor/broadcom/drivers

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-stb/vendor/broadcom/drivers
```

[hash]

```
a3d66c0 nx_ashmem: add ioctl to retrieve per process allocations.
```

device/broadcom/common

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-stb/device/broadcom/common
```

[hash]

```
cba44e8 common: disable mmu for the time being.
bff5f8f bcmpip: expand scope to allow apps to pass private memory information.
f866e80 common: allow configuration of device capabilities.
6c01f8a sepolicy: allow nxserver to issue kill message for lmk.
5ee01dd bcmpip: proc info proxy - driver allowing to gather data otherwise
denied.
e4ef2c8 sepolicy: move v3d dri gem debugfs to specific genfs context.
5007671 common: add mmu variant of memory driver, fix permissions on gpu+gem.
6893439 common: removal of squashfs support in favor of ext4.
```

device/broadcom/cypress

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-stb/device/broadcom/cypress
```

[hash]

```
6370f10 cypress: disable mmu for the time being.
3cabb45 cypress: remove usage of squashfs in favor of ext4.
09052aa cypress: adding _l version for non-a|b mode.
aa3e7b7 cypress: fix up permissions on gpu+gem, align growth size to mmu.
```

device/broadcom/dawson

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-stb/device/broadcom/dawson
```

[hash]

```
36b4148 dawson: disable mmu for the time being.
da965a5 dawson: remove usage of squashfs in favor of ext4.
19a7ed8 dawson: align growth size to mmu.
```

device/broadcom/elfin

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-stb/device/broadcom/elfin
```

[hash]

```
0b899ec elfin: disable mmu for the time being.
0ff46ec elfin: remove usage of squashfs in favor of ext4.
352bda5 elfin: align growth size to mmu.
```

kernel/private/bcm-97xxx/linux-4.1

[repo]

```
git clone ssh://bcmuser@bcm-code.broadcom.com/google-stb/kernel/private/bcm-
97xxx
```

[hash]

```
664395e drivers:misc:brcmstb-proc-info: expand driver scope to private memory
tracker.
affec67 drivers:misc:brcmstb-proc-info: allowing user to query procfs
information.
1544aae drm:gem: report usage per tgid, add single entry for automatic query.
```