openEuler

UADK用户态通用加速器框架介绍

加速器场景、软件、生态

Hao Fang

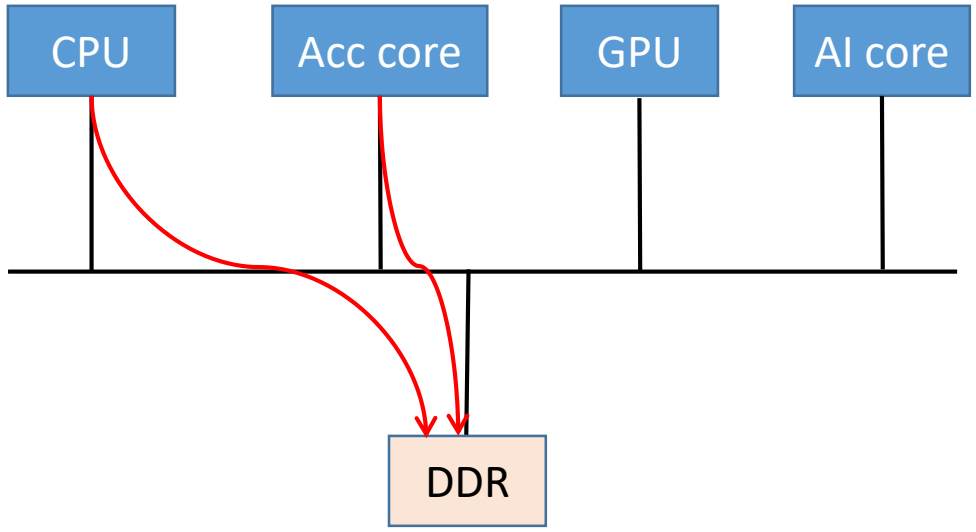
2021.08

目录

CONTENT

openEuler

整体模型



系统特点：

1、异构计算系统，加速器属于IO密集型内存数据访问。
2、用户态的应用需求越来越多。

Acc core可以硬化算法类型

1、对称加解密类：AES/SM4等。
2、摘要类：sha1/sha2/sha3/SM3/等。
3、非对称加解密：RSA/DH/ECC（ECDSA，SM2，ECDH等）
4、压缩类：gzip/zlib/zstd等。
5、随机数生成。

openEuler

linux社区已有方案

- crypto 内核加解密子系统。

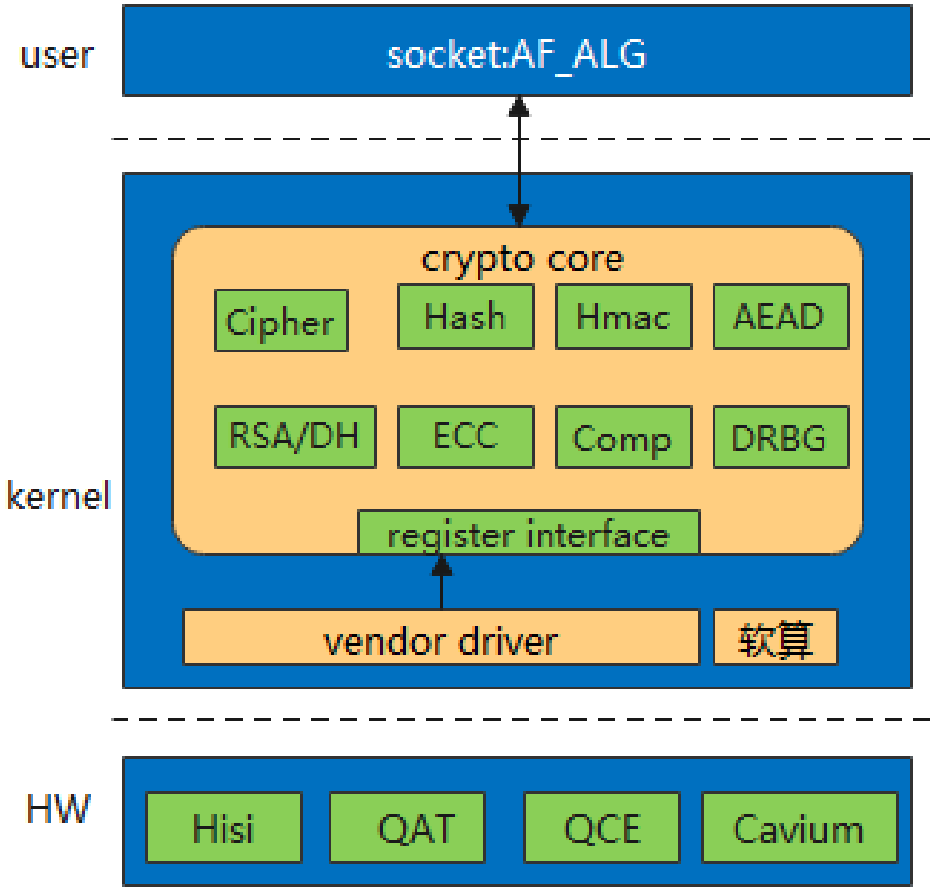
1、内核实现的独立子系统，支持cipher，digest，asymmetric（RSA/DH/ECC），compresss，DRBG等。

2、查看内核支持算法：cat /proc/crypto。

- crypto-usr-if 用户态方案。

通过socket（AF_ALG）实现用户态调用内核态crypto接口，参考Doc/crypto/userspace-if.rst。

--问题：存在系统调用和多次内存copy，效率低。



openEuler

Why UADK

□已有框架的优缺点：

- VFIO-mdev

曾经尝试基于此上传，社区交流结论：作为虚拟化框架很强大，非通用的用户态驱动框架。

1、资源基于Function 级别，动态的队列级别资源申请管理比较麻烦。

2、不支持fork。

- UIO

初衷解决一类板卡的使能；并不通用，地址未有进程隔离。

■核心诉求：

--用户态直接IO，执行dma，不经过系统调用。

--支持用户态的VA可以被设备直接访问。简化用户态的应用编程，免copy和地址转换。

--安全。（隐含，但是社区的核心问题）

openEuler

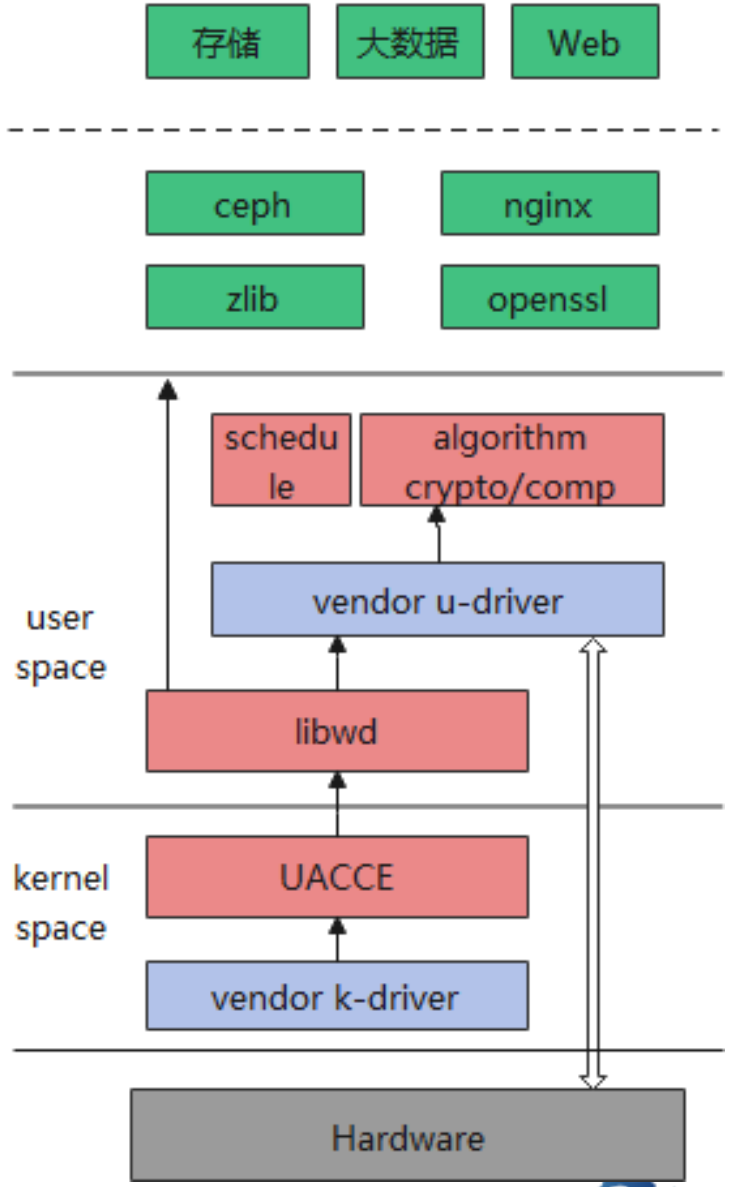
UADK

■ User Space Accelerator Development Kit
访问硬件方式通用，安全，高效。

- 基于IOMMU的Shared Virtual Address（SVA）在加速器和CPU之间共享统一地址页表。
- 共享的是地址，而不是数据。
- 基于IOMMU，限定设备和进程的访问权限和安全边界。

组件包含：内核态UACCE框架, 用户态libwd 和alg libs.

rely on：IOMMU & SVA。

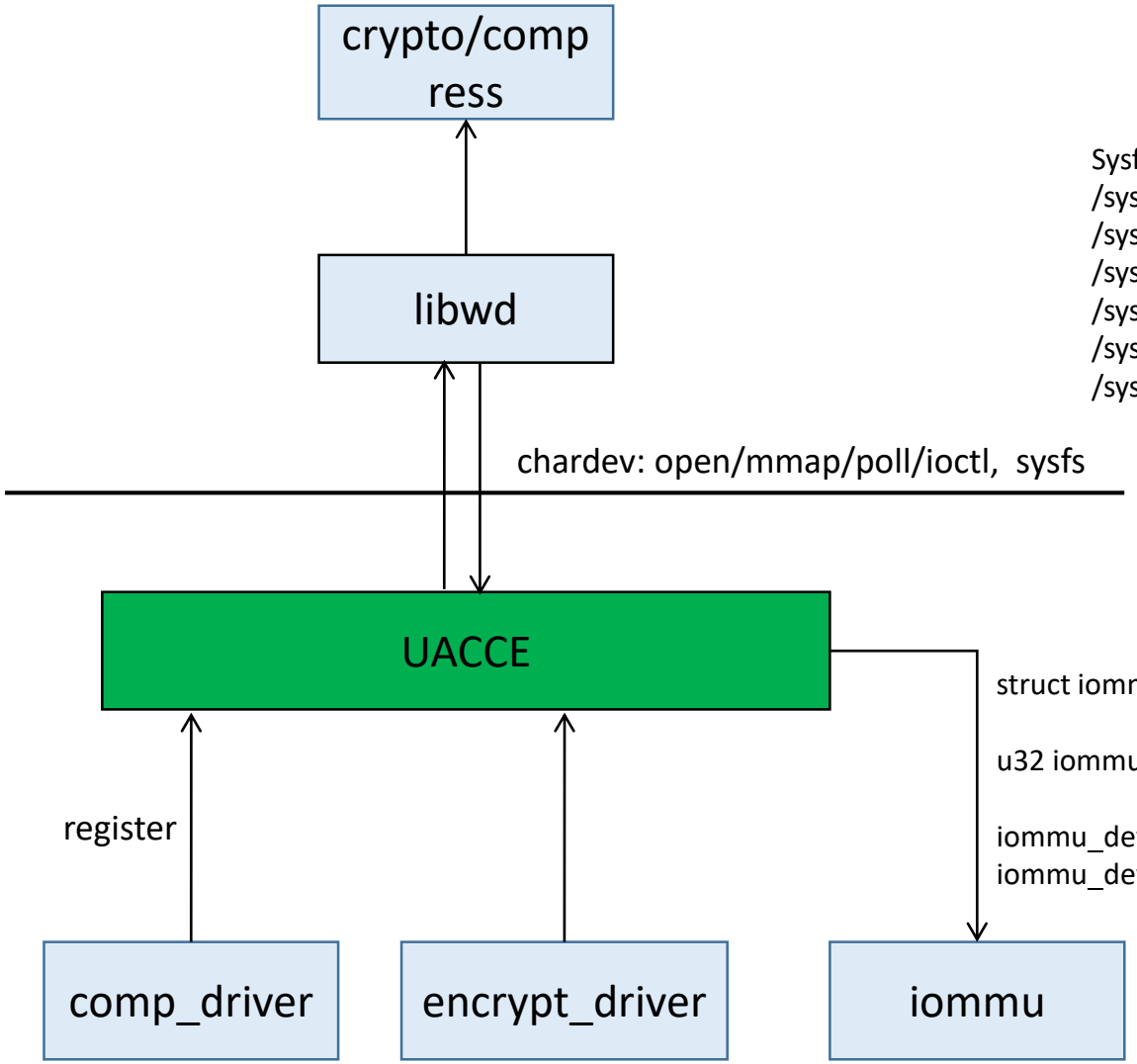


openEuler

UACCE

(Unified/User-space-access-intended Accelerator Framework)

QFR(Queue File Region)
MMIO:
map to the device MMIO space, such as doorbell operation regs.
DUS:
map to memory that share between user and device only, for the user space to send request to the hardware.



Sysfs接口:
/sys/class/uacce/<dev>/api
/sys/class/uacce/<dev>/flags
/sys/class/uacce/<dev>/available_instances
/sys/class/uacce/<dev>/algorithms
/sys/class/uacce/<dev>/region_mmio_size
/sys/class/uacce/<dev>/region_dus_size

struct iommu_sva * iommu_sva_bind_device(struct device *dev,
struct mm_struct *mm, void *drvdata)
u32 iommu_sva_get_pasid(struct iommu_sva *handle)

iommu_dev_enable_feature(dev, IOMMU_DEV_FEAT_IOPF）
iommu_dev_enable_feature(dev, IOMMU_DEV_FEAT_SVA)

openEuler

UADK用户态库

用户态UADK库 contains the following elements:

wd.c

UADK fundamental library which **wraps the basic operation**s to the UACCE device. **libwd** is this library.

drv/*

Hardware user drivers. It helps to fulfill the semantic of algorithm libraries for particular hardware.

wd_[alg].c， wd_sched.c

UADK algorithm libraries. **libwd_comp** is for compression/decompression, **libwd_crypto** is for all encryption/decryption and hash algorithm.

wd_utils.[ch]

Some utility functions used by UADK and its drivers.

test/*

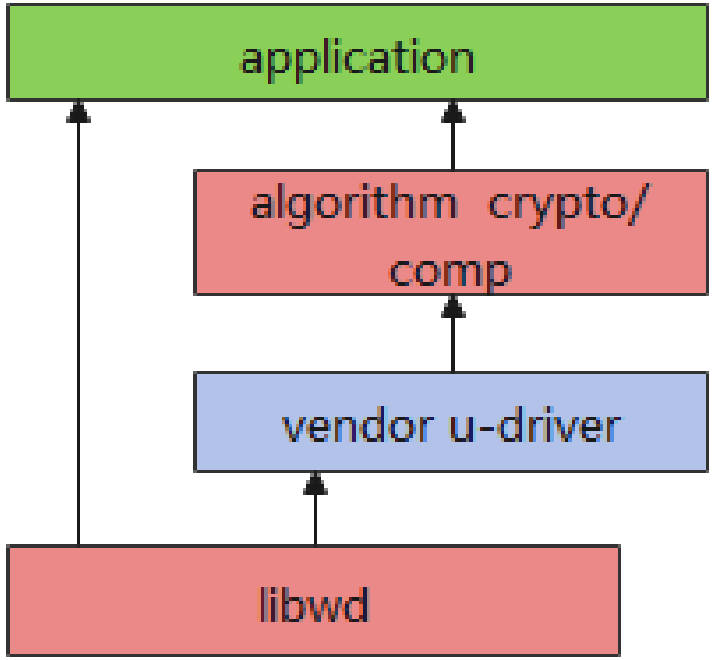
Test applications to use UADK.

include/*

Head files for user APP and hardware drivers.

docs/*

Documentations.



repo地址： https://github.com/Linaro/uadk
https://github.com/Linaro/uadk/tree/master/docs

openEuler

libwd基础库

Context资源申请配置：

- handle_t wd_request_ctx(struct uacce_dev *dev);
- void wd_release_ctx(handle_t h_ctx);
- int wd_ctx_set_io_cmd(handle_t h_ctx, unsigned long cmd, void *arg);

设备查找：

- struct uacce_dev_list *wd_get_accel_list(char *alg_name);
- void wd_free_list_accels(struct uacce_dev_list *list);

内存region映射：（MMIO/DUS）

- void *wd_ctx_mmap_qfr(handle_t h_ctx, enum uacce_qfrt qfrt);
- void wd_ctx_unmap_qfr(handle_t h_ctx, enum uacce_qfrt qfrt);

其他：helper函数，调度，内存池申请（optional）

openEuler

algorithm libs (libwd_crypto/libwd_comp)

libwd_comp为例：

```
struct wd_comp_sess_setup {
    enum wd_comp_alg_type   alg_type;    // zlib or gzip
    enum wd_comp_level      comp_lv;     // compression level
    enum wd_comp_winsz_type win_sz;      // compression window size
    enum wd_comp_op_type    op_type;     // compress or decompress
    enum wd_ctx_mode        mode;        // synchronous or asynchronous
};
```

全局初始化：

```
int wd_comp_init(struct wd_ctx_config *config, struct wd_sched *sched)
void wd_comp_uninit(void)
```

Session句柄：

```
handle_t wd_comp_alloc_sess(struct wd_comp_sess_setup *setup)
void wd_comp_free_sess(handle_t h_sess)
```

同步任务：

```
int wd_do_comp_sync(handle_t h_sess, struct wd_comp_req *req)
```

流模式任务：

```
int wd_do_comp_strm(handle_t h_sess, struct wd_comp_req *req)
```

异步任务：

```
int wd_do_comp_async(handle_t h_sess, struct wd_comp_req *req)
int wd_comp_poll(__u32 expt, __u32 *count)
```

```
typedef void *wd_alg_comp_cb_t(void *cb_param);
struct wd_comp_req {
    void            *src;
    __u32           src_len;
    void            *dst;
    __u32           dst_len;
    wd_alg_comp_cb_t  *cb;
    void            *cb_param;
    __u8            op_type;
    __u32           last;
    __u32           status;
};
```

openEuler

How to use UACCE（vendor)

```
struct uacce_device *uacce_alloc(struct device *parent, struct uacce_interface *interface);
int uacce_register(struct uacce_device *uacce);
void uacce_remove(struct uacce_device *uacce);

struct uacce_ops {
    int (*get_available_instances)(struct uacce_device *uacce);
    int (*get_queue)(struct uacce_device *uacce, unsigned long arg, struct uacce_queue *q);
    void (*put_queue)(struct uacce_queue *q);
    int (*start_queue)(struct uacce_queue *q);
    void (*stop_queue)(struct uacce_queue *q);
    int (*is_q_updated)(struct uacce_queue *q);
    int (*mmap)(struct uacce_queue *q, struct vm_area_struct *vma, struct uacce_qfile_region *qfr);
    long (*ioctl)(struct uacce_queue *q, unsigned int cmd, unsigned long arg);
};
```

Sample：kernel/drivers/crypto/hisilicon/

Require：设备支持iommu normal模式，支持SVA stall/pri mode缺页。

openEuler

性能数据

- vs af_alg (AES算法)

Block size(bytes)	16	64	256	1024	8192
AF_ALG	37.86k	140.05k	565.33k	2530.30k	19802.79k
libwd	6327.14k	24477.50k	97456.55k	335090.47k	1797931.01k

- vs noSVA (zlib算法)

sva: 5.6G at most

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf

Compress bz=8192, speed=556.533 MB/s

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf -c 10

Compress bz=8192, speed=1381.276 MB/s

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf -c 20

Compress bz=8192, speed=3134.403 MB/s

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf -c 30

Compress bz=8192, speed=4316.537 MB/s

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf -c 40

Compress bz=8192, speed=5617.674 MB/s

no-sva: 2.2G at most

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf

Compress bz=8192, speed=2294.555 MB/s

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf -c 10

Compress bz=8192, speed=2274.646 MB/s

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf -c 10 -q 10

Compress bz=8192, speed=2253.909 MB/s

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf -c 20

Compress bz=8192, speed=2252.999 MB/s

$ sudo ./test_bind_api -b 8192 -s 81920000 -o perf -c 20 -q 10

Compress bz=8192, speed=2244.004 MB/s

生态进展

- SVA 基础特性，已进入kernel 5.14 release，已支持stall 缺页设备（2021.07）

  upstream status： https://jpbrucker.net/sva/　　jean-philippe@linaro.org

- UACCE框架，历经V13，已进入kernel 5.7 release.(2020.02)

  https://lkml.org/lkml/2020/2/11/54.

- UADK lib(libwd+alglib+vendor_driver), 已在github/linaro 开源.

  目前支持kunpeng920硬件加速器.

- 内核态Vendor driver已进入kernel主线，from 5.3 release.

  driver/crypto/hisilicon/zip ， sec2， hpre

openEuler

生态进展（openssl-engine）

- 目前支持算法：

  --cipher类：AES，SM4等

  --digest类：SHA-1，SHA-2，SM3，MD5等

  --asymmetric：RSA，DH，ECC（SM2、ECDSA、ECDH、X25519/X448等)

  --sync and async mode.

- git repo：https://github.com/Linaro/openssl-uadk

- 使用参考

  https://github.com/Linaro/openssl-uadk/blob/master/README

openEuler

UADK in OpenEuler

■ 目标OpenEuler21.09/22.03 回合使能

- OpenEuler 21.09 kernel 已回合UACCE

  https://gitee.com/openeuler/kernel/tree/openEuler-21.09/drivers/misc/uacce

- UADK用户态库已回合version 2.3.11

  https://gitee.com/src-openeuler/libwd/tree/master/

- OpenEuler 21.09 kernel 已完整回合支持SVA

openEuler

Todo list

■ 关注vSVA的社区基础特性持续演进

1、dev/iommu_uAPI 设计的社区讨论（intel）

2、vSMMv3/pSMMUv3 2 stage VFIO integration（redhat）

■ OpenSSL engine的持续完善

https://github.com/Linaro/openssl-uadk

1、新增算法的适配（椭圆曲线等）

2、场景的适配优化（Nginx、Ceph等）

openEuler

最后

欢迎大家一起加入UADK开源生态社区。

https://github.com/Linaro/uadk

https://github.com/Linaro/openssl-uadk

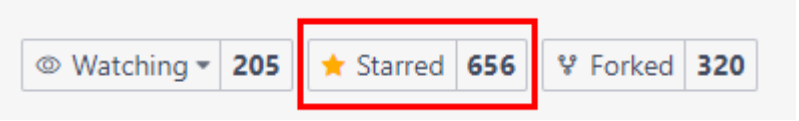
openEuler

✓ openEuler kernel gitee 仓库

源代码仓库

https://gitee.com/openeuler/kernel

欢迎大家多多 Star，多多参与社区开发，多多贡献补丁。



✓ maillist、issue、bugzilla

可以通过邮件列表、issue、bugzilla 参与社区讨论

欢迎大家多多讨论问题，发现问题多提 issue、bugzilla

https://gitee.com/openeuler/kernel/issues

https://bugzilla.openeuler.org

kernel@openeuler.org

✓ openEuler kernel SIG 微信技术交流群

请扫描右方二维码添加小助手微信

或者直接添加小助手微信（微信号：openeuler-kernel）

备注“交流群”或“技术交流”

加入 openEuler kernel SIG 技术交流群

技术交流

openEuler

Thank you

openEuler