



Univerzitet u Novom Sadu

Fakultet tehničkih nauka

Odsek za računarsku tehniku i
računarske komunikacije



Projektovanje Namenskih Računarskih Struktura 2

Sistemsko programiranje u Androidu

Auditorne vežbe – Android internals 1

Zimski semestar školske godine 2015/2016

C/C++ Servis

- ❖ Program koji nema direktnu interakciju sa korisnikom
- ❖ Pruža usluge drugim procesima
- ❖ Aktivan je tokom celokupnog vremena rada operativnog sistema (*daemon*)
- ❖ Naziv se po konvenciji završava slovom *d* (npr. `bold` – *Volume Daemon*, `dhcpcd` – *DHCP Client Daemon*)
- ❖ Primer: `exd` – Example Daemon

```
int main()
{
    while(1)
    {
        ALOGD("TICK");
        sleep(5); // seconds
    }
}
```

Zadatak 1

- ❖ Napraviti C/C++ servis na osnovu datog primera. Servis nazvati `exd` (Example Daemon).
- ❖ Rezultat smestiti u:
`vendor/rtrk/pnrs_x86/dan04/zadatak01`

*.rc

- ❖ Datoteke koje se učitavaju od strane *init* procesa
- ❖ `/init.rc` primarna *.rc datoteka, zadužena za osnovnu inicijalizaciju sistema
- ❖ `/init.${ro.hardware}.rc` inicijalizacije specifične za produkt
- ❖ za emulator `ro.hardware = goldfish`
(`device/generic/goldfish/init.goldfish.rc`)
- ❖ Sadržaj *.rc datoteka definisan je Android Init jezikom
(`system/core/init/readme.txt`)

- ❖ Primer servisa:

```
service exd /system/bin/exd
    oneshot
```

```
on boot
    start exd
```

SELinux

- ❖ Security-Enhanced Linux (SELinux)
- ❖ Obavezujuća kontrola pristupa (eng. MAC – Mandatory Access Control)
- ❖ Prisutan od 4.3 verzije Androida, obavezan od verzije 5.0

- ❖ **Primer:**

- ❖ `vendor/rtrk/pnrs_x86/sepolicy/exd.te`

```
type exd, domain;  
type exd_exec, domain;  
init_daemon_domain(exd)
```

- ❖ `vendor/rtrk/pnrs_x86/sepolicy/file_contexts`
`/system/bin/exd u:object_r:exd_exec:s0`

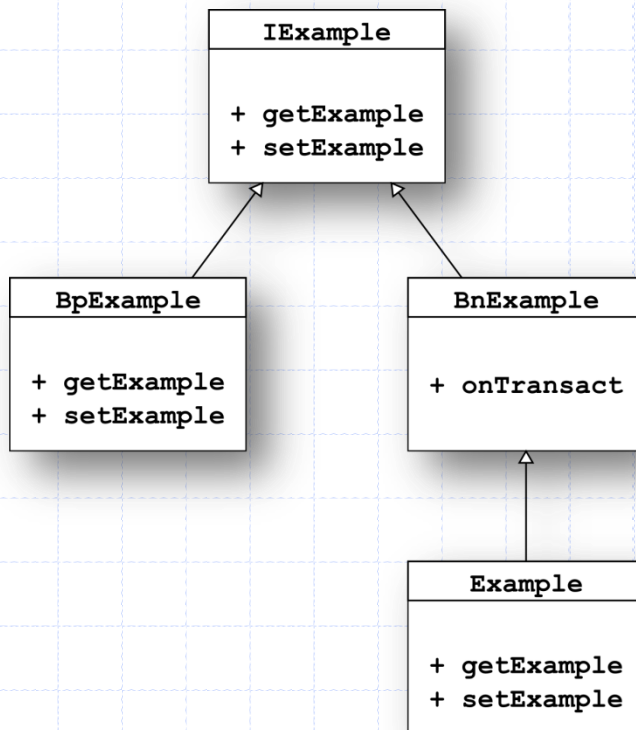
- ❖ `vendor/rtrk/pnrs_x86/BoardConfig.mk`
`BOARD_SEPOLICY_DIRS += vendor/rtrk/pnrs_x86/sepolicy`

Zadatak 2

- ❖ Pokrenuti servis iz zadatka 1 uz pomoć *init* procesa.
- ❖ Dodati `exd` servis u `init.goldfish.rc`
- ❖ Definirati SELinux pravila za `exd` servis

Binder

- ❖ IPC mehanizam Androida
- ❖ Proširenje Linux jezgra
- ❖ Klijent-server organizacija
- ❖ Java - Java, C++ - C++, Java - C++



Primer:

```

##### Service Implementation #####
#include "Example.h"
#include <binder/IServiceManager.h>
#include <binder/IPCThreadState.h>

defaultServiceManager()->addService(
    String16("example"), new Example());
ProcessState::self()->startThreadPool();
IPCThreadState::self()->joinThreadPool();
    
```

```

##### Client Implementatin #####
#include <IExample.h>
#include <binder/IServiceManager.h>
#include <binder/IBinder>

sp<IServiceManager> smanager =
    defaultServiceManager();
sp<IBinder> binder = smanager->getService(
    String16("example"));
sp<IExample> example =
    interface_cast<IExample>(binder);

example->setExample(value);
example->getExample();
    
```

Zadatak 3

- ❖ Proširiti primer iz zadatka 2 Binder spregom uz pomoc `resursi/libbinderexample` programske biblioteke.
- ❖ Analizirati tok programskih poziva pri komunikaciji preko bindera.
- ❖ **Napomena:** Za ovaj zadatak pokretati emulator na sledeći način
`emulator -selinux permissive`
- ❖ Rezultat smestiti u: `vendor/rtrk/pnrs_x86/dan04/zadatak03`

Zadatak 4

- ❖ Zameniti postojeću asemblersku implemetaciju memcpy funkcije u bionic biblioteci
- ❖ Napisati izvršni sistemski Android program koji će potvrditi ispravnost nove implementacije
- ❖ Rezultat smestiti u:
`vendor/rtrk/pnrs_x86/dan04/zadatak04`