Android internals – lab

1.create a stand alone service

- Add a new directory in ~/aosp/external
- Create a makefile – Android.mk (copy from netcat and make changes)
- Add the following source:

```
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>

int main() {
           int fd=open("data1",O_RDWR|O_CREAT,0660);
    char buffer[100];
    struct sockaddr_in addr = {0};
    size_t addrlen, n;
    int sockfd = socket(AF_INET, SOCK_DGRAM, 0);

    addr.sin_family = AF_INET;
    addr.sin_port = htons(2000);
    addr.sin_addr.s_addr = INADDR_ANY;
    bind(sockfd, (struct sockaddr*)&addr, sizeof(addr));

    addrlen = sizeof(addr);
    while(1)
    {
        n = recvfrom(sockfd, (void*)buffer, 100, 0,
                (struct sockaddr*)&addr, (unsigned int *)
&addrlen);
        buffer[n] = '\n';
        write(fd,buffer,n+1);
    }
    close(fd);
    return 0;
}
```

2. compile the rom, run the emulator and start the service using adb:
   - cd ~/aosp
   - source build/envsetup.sh
   - lunch 1
   - make
   - emulator &
   - adb shell
   - cd data
   - applog &

3. create android application to use it
   - create a simple activity with one button
   - implement click event: create and start a thread to run the following code:

```
    void sendonemsg()
    {
     DatagramSocket s;
     try {
          s = new DatagramSocket();
          InetAddress local = InetAddress.getByName("127.0.0.1");
          String str="hello service";
          int msg_length = str.length();
          DatagramPacket p = new DatagramPacket(str.getBytes(),
msg_length, local, 2000);
          s.send(p);
     } catch (Exception e) {
          // TODO Auto-generated catch block
          e.printStackTrace();
     }

    }
```

   - add internet permissions to the project
   - run the application on the custom device