



**MIDDLE EAST TECHNICAL UNIVERSITY
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING**

CHASING GROUP HANDSHAKE PROTOCOL CHEAT SHEET FOR RASPBERRY PI

Instructor:	Murat EYÜBOĞLU
Company:	SolidVisio Softcorp
Company DS:	DS-2 (Aydan ERKMEN)

A. INTRODUCTION

When chasing group representatives gathered together to test handshake protocol communication system, every attendee thought that their implementation worked because they tried it with some other platform -like their smartphones or computers. But when tested with other groups, it was observed that these systems either didn't work together at all or did work but not in accordance with the standards set previous semester. Some examples are as follows:

- The system works but via a hotspot router (not peer-to-peer Ad-Hoc connection like it was stated in standard report).
- The system works but without a static IP (all groups should have their decided static IP numbers)

With the help of another group, a solution which satisfies all the standards has been found. This is an **informal** basic cheat sheet for other groups.

B. LIBRARY FOR TCP SOCKET COMMUNICATION

Raspberry Pi's 'socket' library is a powerful, simple and reliable tool which satisfies the standard report and can be utilized for handshake protocol. Below is a simple code to implement a server and a client:

WARNING: It is your responsibility to integrate handshake protocol code to your main system. These are just basic building blocks. You may want to construct handshake protocol with 'try-except' for error mitigation. Also, it is your responsibility to test it to obtain incoming data. For example; when you use print function to print the data, it shows it as b'XXXX' but if you put the message in len function, maybe it will just ignore b' ' part, I don't know. Just test it several times and make the required changes.

```
import socket
import select
import time, sys
from time import sleep

# AS SERVER

host = '192.168.1.18' # this is your group's IP address. Please note that if you fail to
# set your device's IP address accordingly, it will not work and give an error.
# Please refer to the section where I explain setting up Ad-Hoc wifi network.

port = 5000

def server():

    socket.sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    socket.sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
    socket.host = host
    socket.port = port
    socket.sock.bind((host, port))
    socket.sock.listen(1)
    print('Started socket server (host {}, port {})'.format(socket.host, socket.port))
    client_sock, client_addr = socket.sock.accept()
    print('Client {} has been connected'.format(client_addr)) # This is other group which will be client
    print('You can communicate now')

    while True: #This is a dummy code that just gets the data Client sent and sends it back to Client
        data = client_sock.recv(255) #How to get incoming data
        if len(data) != 0:
            print(data)
            client_sock.send(data) # How to send data

server()
```

Figure 1. Server Example

```
import socket
import select
import time, sys
from time import sleep

# AS CLIENT

def client():
    s = socket.socket()
    s.connect(('192.168.1.20',5000)) #This is the IP address of server group you will connect to

    while True:
        s.send(b'XX01') #It just sends a dummy message. Please note that XX represents your group ID as
        # it was written in standard report. Also don't forget to put b'' part. The message goes inside.
        sleep(1)

client()
```

Figure 2. Client Example

NOTE: It should be obvious that server should be established first for client to connect 😊

C. HOW TO SETUP AN AD-HOC NETWORK WITH STATIC IP

BIG WARNING: Please make sure that you have never ever touched any network / wifi related settings or changed any network / wifi related file in Raspberry Pi before proceeding. If you did so, reinstall your operating system. Otherwise, it might not work. We had this problem with one group.

Step 1

<https://github.com/simondlevy/RPiAdHocWiFi>

Go to the link above and download RPiAdHocWiFi folder. Unzip it and copy it to your desktop in Raspberry Pi.

Step 2

Open the terminal and go to the folder above using cd command. Then, type the code below:

```
sudo ./install.sh
```

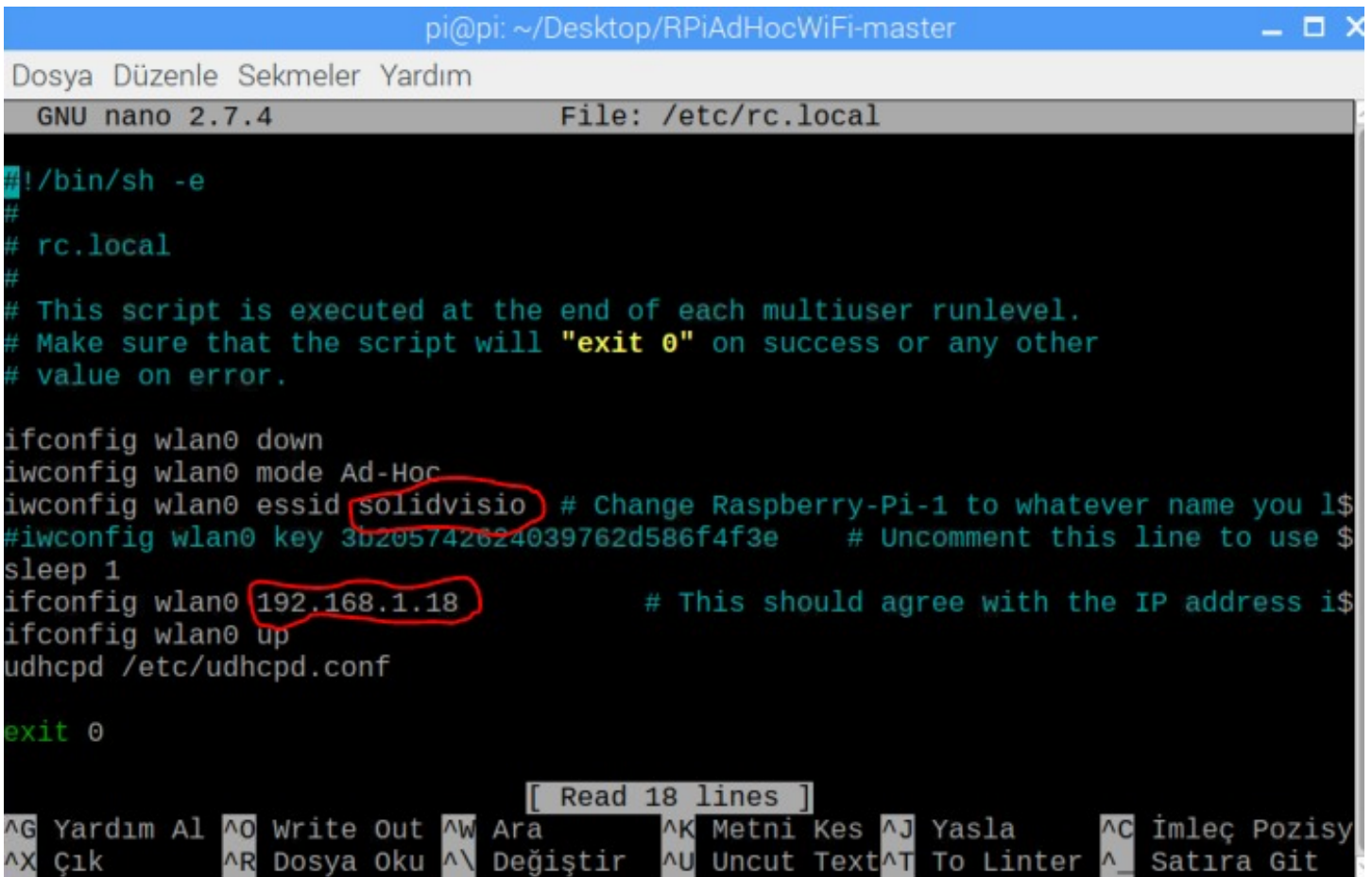
Then, it will install the required package. Obviously, your RPi needs to be connected to a working wifi network with internet connection.

Step 3

Now, we will modify rc.local file to get a static IP that we want.

```
sudo nano /etc/rc.local
```

The document should look like this:



```
pi@pi: ~/Desktop/RPiAdHocWiFi-master
Dosya Düzenle Sekmeler Yardım
GNU nano 2.7.4 File: /etc/rc.local

#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.

ifconfig wlan0 down
iwconfig wlan0 mode Ad-Hoc
iwconfig wlan0 essid solidvisio # Change Raspberry-Pi-1 to whatever name you l$
#iwconfig wlan0 key 3b205742624039762d586f4f3e # Uncomment this line to use $
sleep 1
ifconfig wlan0 192.168.1.18 # This should agree with the IP address i$
ifconfig wlan0 up
udhcpd /etc/udhcpd.conf

exit 0

[ Read 18 lines ]
^G Yardım Al ^O Write Out ^W Ara ^K Metni Kes ^J Yasla ^C İmleç Pozisy
^X Çık ^R Dosya Oku ^\ Değiştir ^U Uncut Text ^T To Linter ^_ Satıra Git
```

Figure 3

It comes with Raspberry-Pi-1 name as default, change it to another name because everybody will have the same name otherwise. The second marked area shows your IP address. Our group has ID 18 according to standards. Change it accordingly. Then press CTRL+X > E (or Y). > ENTER

Now, we need to modify udhcpd.conf.

```
sudo nano /etc/udhcpd.conf
```

The document should look like this:

```
pi@pi: ~/Desktop/RPiAdHocWiFi-master
Dosya Düzenle Sekmeler Yardım
GNU nano 2.7.4 File: /etc/udhcpd.conf
#
# This configuration will serve up IP addresses for an ad-hoc wireless network
#
start      192.168.1.0
end        192.168.1.254
interface  wlan0
max_leases 64

[ Read 8 lines ]
^G Yardım Al ^O Write Out ^W Ara ^K Metni Kes ^J Yasla ^C İmleç Pozisy
^X Çık ^R Dosya Oku ^\ Değiştir ^U Uncut Text ^T Denetime ^_ Satıra Git
```

Figure 4

Change the numbers as such and CTRL+X > E (or Y) > ENTER

Now, your Ad-Hoc wifi network is ready. Reboot your RPi and it will be done. You can see the network you setup via your smartphone. Please note that the Figure 4 setting is just for the devices that don't have their own static IP, actually we won't even need this feature since we will all have static IP addresses but let's stick to the original document description.

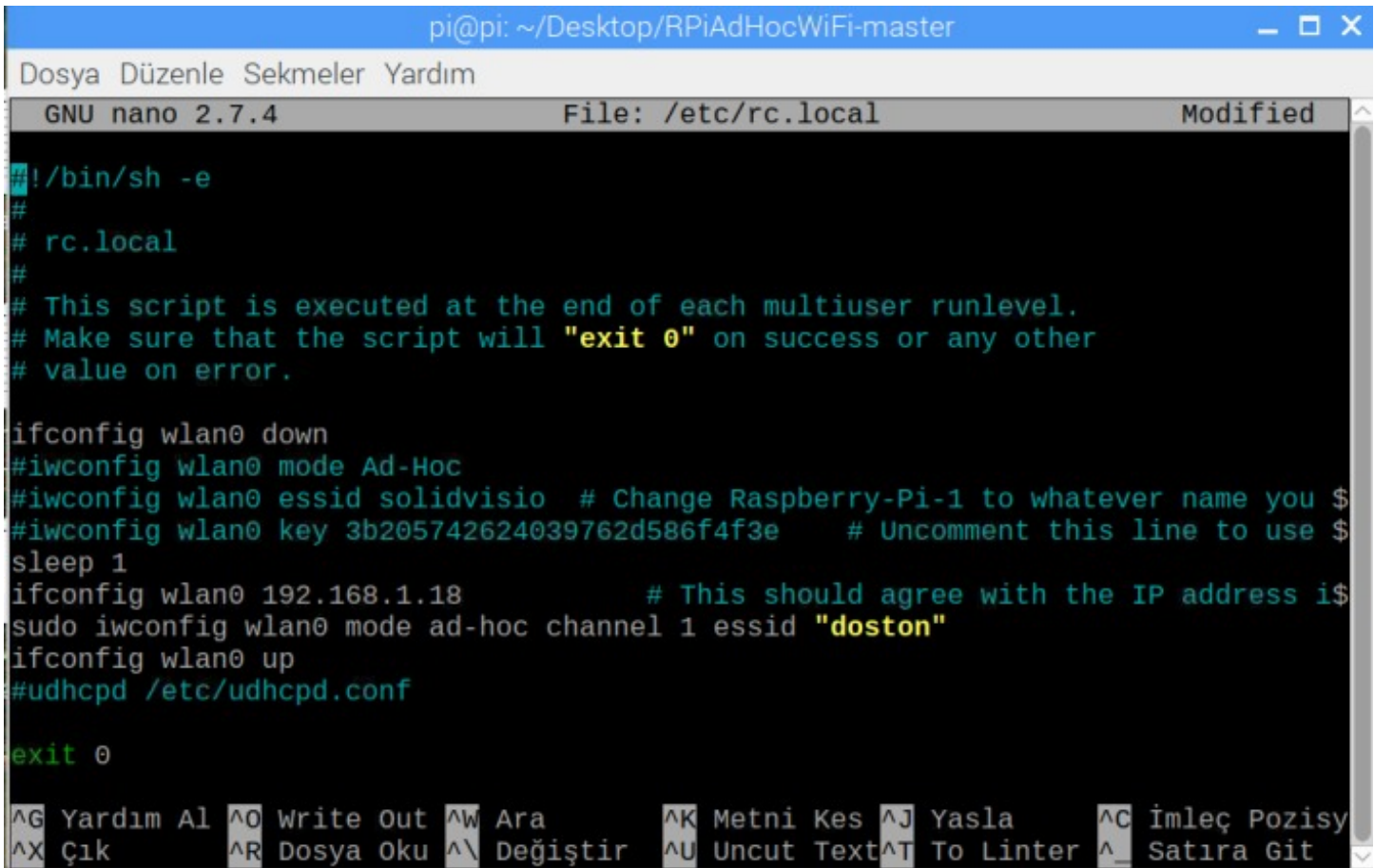
D. HOW TO JOIN AN AD-HOC NETWORK WITH STATIC IP

I guess you already have Github code setup as I described in step 1 and 2.

Step 1:

```
sudo nano /etc/rc.local
```

The document should look like this:



```
pi@pi: ~/Desktop/RPiAdHocWiFi-master
Dosya Düzenle Sekmeler Yardım
GNU nano 2.7.4 File: /etc/rc.local Modified
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.

ifconfig wlan0 down
#iwconfig wlan0 mode Ad-Hoc
#iwconfig wlan0 essid solidvisio # Change Raspberry-Pi-1 to whatever name you $
#iwconfig wlan0 key 3b205742624039762d586f4f3e # Uncomment this line to use $
sleep 1
ifconfig wlan0 192.168.1.18 # This should agree with the IP address is$
sudo iwconfig wlan0 mode ad-hoc channel 1 essid "doston"
ifconfig wlan0 up
#udhcpd /etc/udhcpd.conf

exit 0

^G Yardım Al ^O Write Out ^W Ara ^K Metni Kes ^J Yasla ^C İmleç Pozisy
^X Çık ^R Dosya Oku ^\ Değiştir ^U Uncut Text ^T To Linter ^_ Satıra Git
```

Figure 5

Please note that you IP address stays the same (192.168.1.18 for me) but there is a new line below it. Change doston to whatever network you are trying to connect. If we all choose simple names with lower cases, life will be easier at KKM ☺

Please note that we just modified the rc.local file with some # to uncomment the lines we don't want. You will do the same if you want to switch roles and be a network again. Just make the rc.local look like in Figure 3 again. Voila, you are a network now.

```
sudo nano /etc/network/interfaces
```

Make it look like this if it doesn't already:


```
pi@pi: ~/Desktop/RPiAdHocWiFi-master
Dosya Düzenle Sekmeler Yardım
GNU nano 2.7.4 File: /etc/network/interfaces

## interfaces(5) file used by ifup(8) and ifdown(8)

# Please note that this file is written to be used with dhcpcd
# For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'

# Include files from /etc/network/interfaces.d:
#source-directory /etc/network/interfaces.d

[ Read 7 lines ]
^G Yardım Al ^O Write Out ^W Ara ^K Metni Kes ^J Yasla ^C İmleç Pozisy
^X Çık ^R Dosya Oku ^\ Değiştir ^U Uncut Text ^T Denetime ^_ Satıra Git
```

Figure 6

E. CONCLUSION

I guess I don't have to warn you that the sequence of events should be like this:

1. Group A sets up Ad-Hoc network and reboots (and waits properly until RPi stabilizes)
2. Group B changes the settings accordingly for connecting a network and reboots.
3. Group A (or B) starts running a server.
4. Group B (or A) connects to the server as a client

Another warning: We will have approximately 5 minutes before the race to set the network names and IP addresses of opponents accordingly. You don't have to (and can't) make the process automatic.

After these settings, you won't be able to connect your RPi to another normal wifi network for internet. If you want to do so, you may uninstall the package (check the github page, they explain how to do that there), do whatever you want to do and install it back again for handshake protocol. We tried this, and there was no problem.