



Digital identity

IP Port Mirroring Workshop

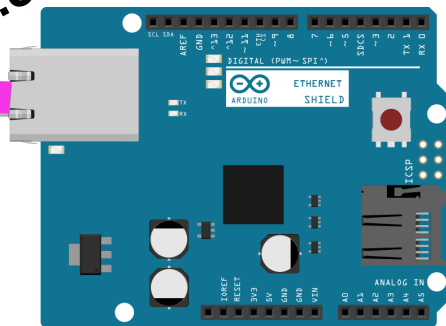
In 2018, Is there a Safe Way to Transmit by network your Confidential Data ?

You will use an Arduino card like an adapter to connect yourself to a computer.

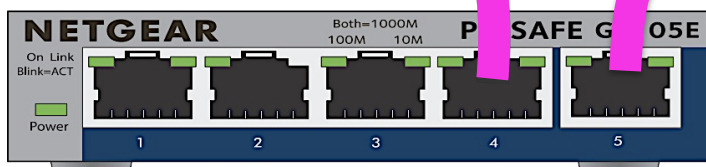
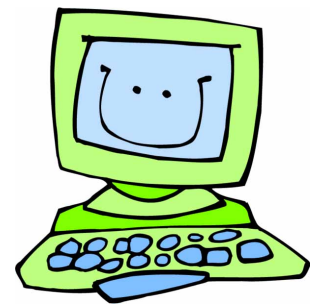
Two teams [A & B] will try to safely transmit to and receive data from each other.

Team A

IP 192.168.1.80

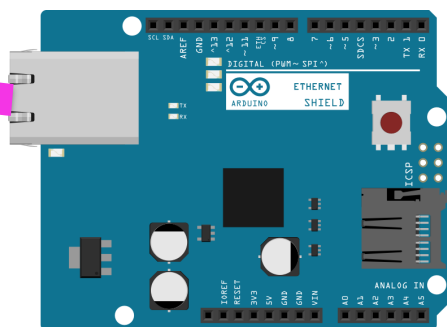


USB link

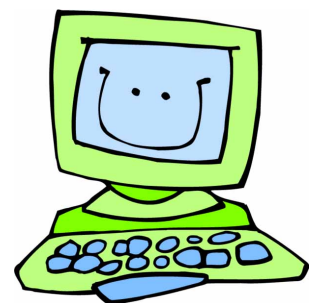


Team B

IP 192.168.1.90



USB link



Team B Step 1 :

Connect your Ethernet shield [previously connected to the Arduino card] to the port 5 of the switch Netgear GS105e V2.

Team B Step 2 :

Open the Arduino IDE [integrated development environment].

- Select an UNO kind of Arduino card
- Select the right serial Port connected to the Arduino card [for instance COM6]
- Upload the sketch "UDPSendReceiveString90To80.ino".

Upload

Compiles your code and uploads it to the configured board. See [uploading](#) below for details.

- Open an Monitoring window

Serial Monitor

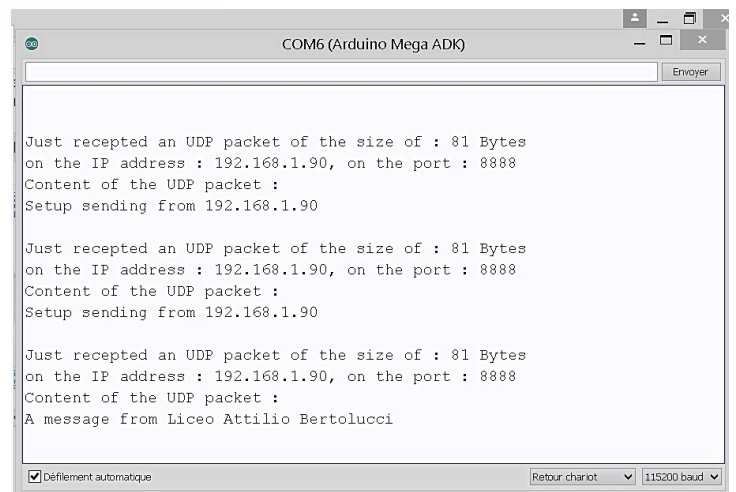
Opens the **serial monitor**.

Select the speed rate of 115200 bits per second

Select Carriage Return mode

Team B Step 3 :

Start a Chat between you and Team A



```
COM6 (Arduino Mega ADK)
Just received an UDP packet of the size of : 81 Bytes
on the IP address : 192.168.1.90, on the port : 8888
Content of the UDP packet :
Setup sending from 192.168.1.90

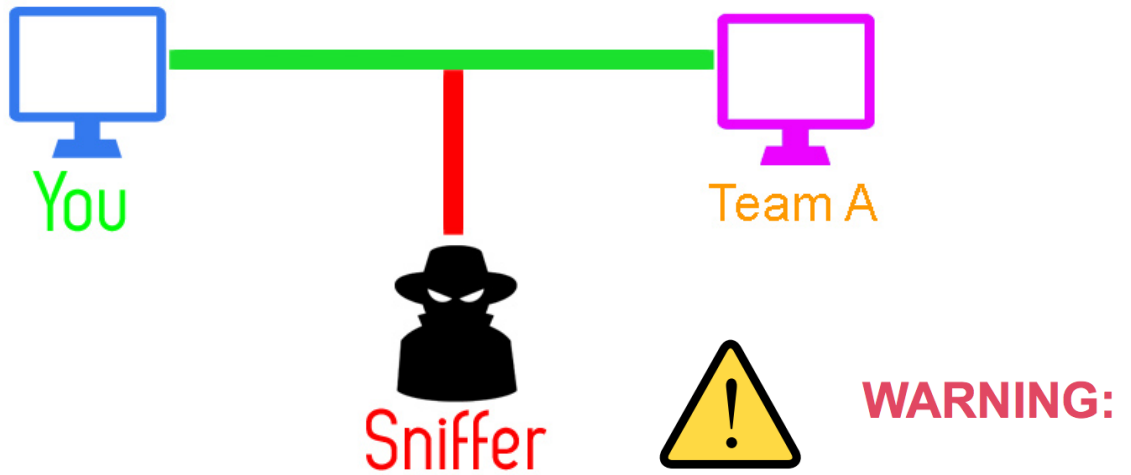
Just received an UDP packet of the size of : 81 Bytes
on the IP address : 192.168.1.90, on the port : 8888
Content of the UDP packet :
Setup sending from 192.168.1.90

Just received an UDP packet of the size of : 81 Bytes
on the IP address : 192.168.1.90, on the port : 8888
Content of the UDP packet :
A message from Liceo Attilio Bertolucci

Défilement automatique Retour chariot 115200 baud
```



Question : Is there any sniffer on the network between you and Team A ?



Check if anyone are sniffing your chat from the switch of your network...

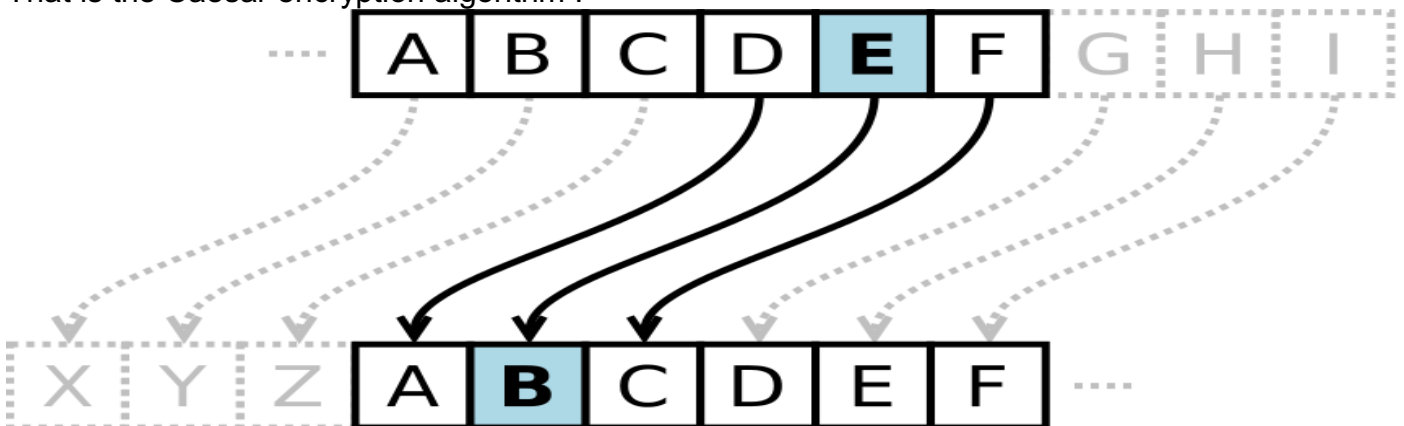
So may we share encrypted messages on our network ?



The goal is to convert a plain text to a cipher text like this :

Plain Text: ABCDEFGHIJKLMNOPQRSTUVWXYZ
Cipher Text: XYZABCDEFGHIJKLMNPOQRSTUVWXYZ

That is the Caesar encryption algorithm :



Team B Step 4 :

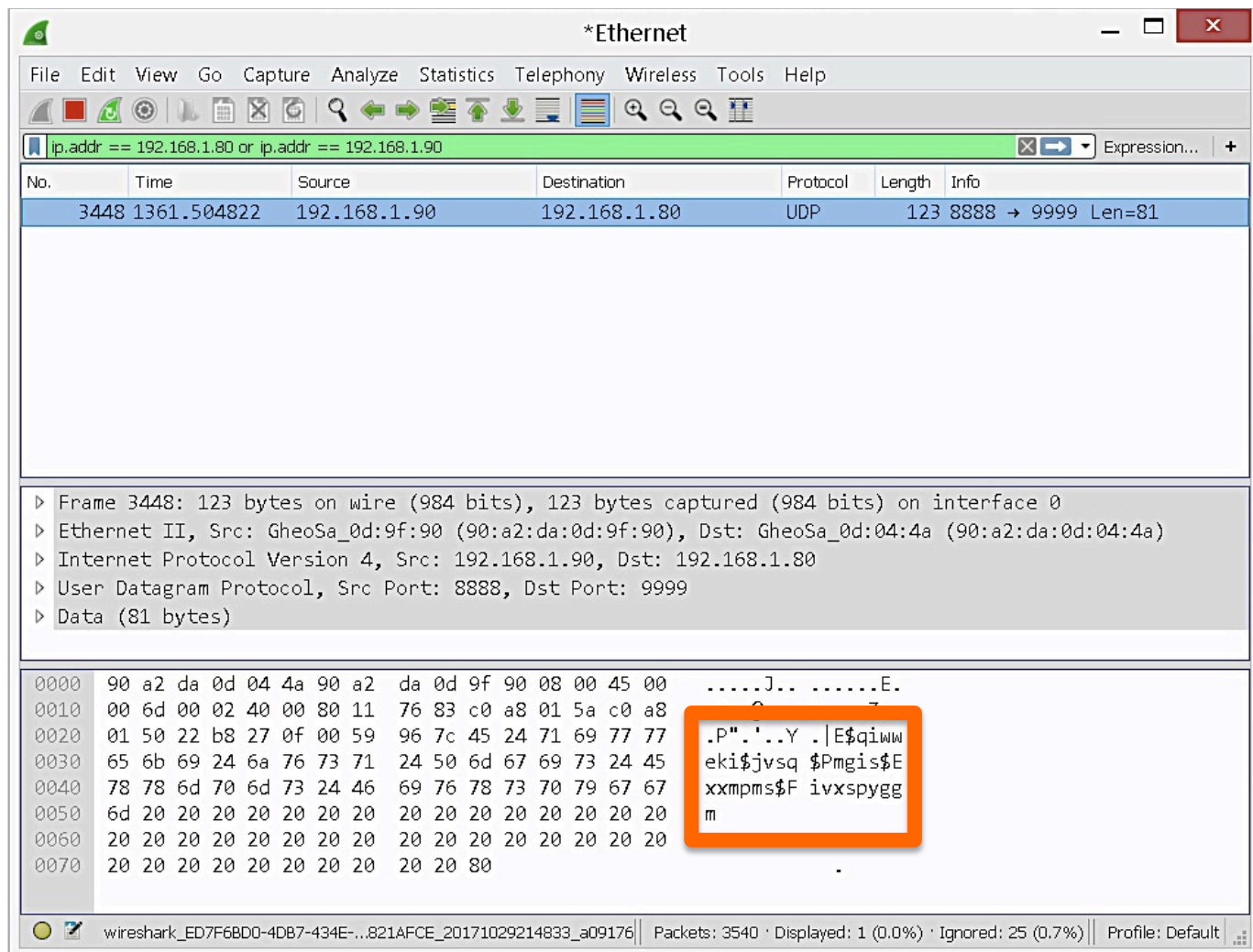
- Upload the sketch "CipherUDPSendReceiveString90To80.ino".

Team B Step 5 :

Start a new Chat between you and Team A

- Open an Monitoring window

Any sniffer guys will only be able to read a cipher message like this :



The image shows a Wireshark network traffic capture window titled "*Ethernet". The filter bar shows the filter "ip.addr == 192.168.1.80 or ip.addr == 192.168.1.90". The packet list pane shows a single packet (No. 3448) at time 1361.504822, from source 192.168.1.90 to destination 192.168.1.80, protocol UDP, length 123, and info "8888 -> 9999 Len=81". The packet details pane shows the following information:

- Frame 3448: 123 bytes on wire (984 bits), 123 bytes captured (984 bits) on interface 0
- Ethernet II, Src: GheoS... (90:a2:da:0d:9f:90), Dst: GheoS... (90:a2:da:0d:04:4a)
- Internet Protocol Version 4, Src: 192.168.1.90, Dst: 192.168.1.80
- User Datagram Protocol, Src Port: 8888, Dst Port: 9999
- Data (81 bytes)

The packet bytes pane shows the raw data in hexadecimal and ASCII. The ASCII column contains the following text:

```
.....J.. .....E.  
.P".'.Y .|E$qiww  
eki$jvsq $Pmgis$E  
xxmpms$F ivxspygg  
m
```

The text in the ASCII column is highlighted with an orange box, indicating that the message has been encrypted (ciphered).

So be aware when you chat online !

