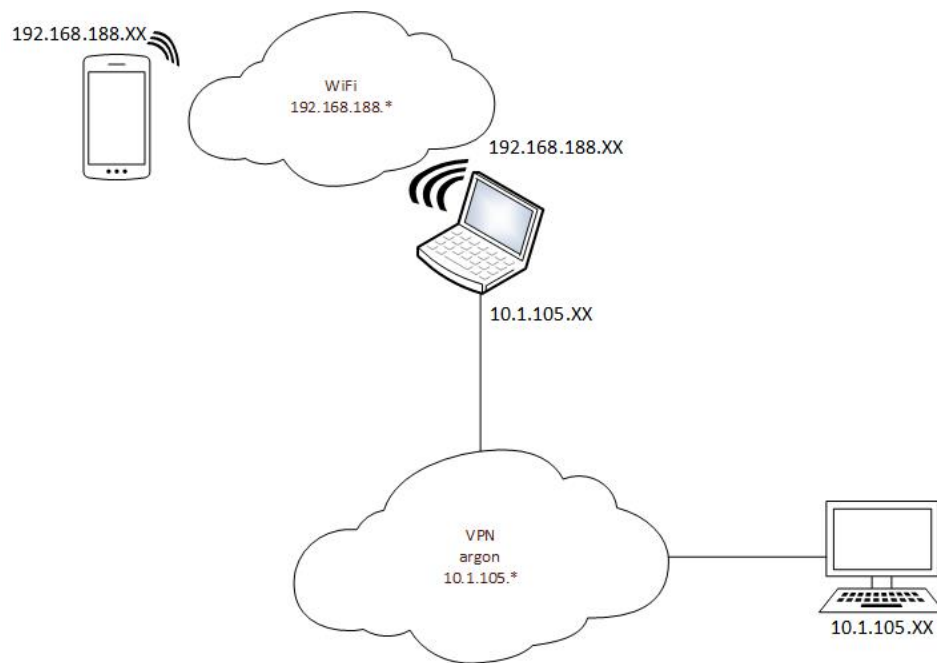


MV3500  
Final Project  
**„Team Germany“**  
by:

Stefan Goericke  
and  
Bernd Weissenberger

**Project Description**

In our final project we send ESPDUs from a mobile device (self-written application) over a private Wi-Fi to a laptop. This PDU contains the actual position of the mobile phone. On the laptop there is a kind of “relay” software running. This software captures the ESPDU packet and send it over the VPN (argon) to another laptop (Stefan) within the argon network. On this machine some part of the ESPDU is printed out.



## Implementation

The code for the Android device was written in KOTLIN. Therefore, the openDIS libraries had to be compiled with JAVA version 1.8.

Because KOTLIN is a relatively new language, here are the important part of the source code out of the MainActivity.kt class (all the code was uploaded to GitLab).

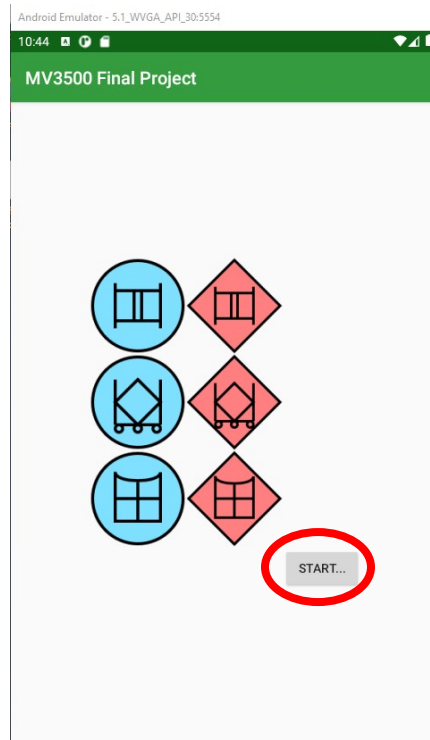
```
startButton.setOnClickListener {
    try{
        val socket = MulticastSocket(3000)
        // position?
        fusedLocationClient = LocationServices.getFusedLocationProviderClient(this)
        if (checkPermission(
            Manifest.permission.ACCESS_COARSE_LOCATION,
            Manifest.permission.ACCESS_FINE_LOCATION
        )) {
            fusedLocationClient.lastLocation?.
            addOnSuccessListener(
                this
            ) { location: Location? ->
                // Got last known location. In some rare
                // situations this can be null.
                if(location == null) {
                    Log.e("GPS_ERROR", location.toString())
                    // TODO, handle it
                } else location.apply {
                    // Handle location object
                    latitude = location.latitude
                    longitude = location.longitude
                    altitude = location.altitude
                    // creating the pdu
                    var espdu = EntityStatePdu()
                    var espduLocation = Vector3D.Double()
                    espdu.exerciseID = 1.toByte()
                    espdu.entityID = EntityID().setEntityID(666)
                    espdu.forceID = ForceID.OPPOSING
                    var entityLocation = espdu.setEntityLocation(espduLocation)
                    var disCoordinates = CoordinateConversions.
                        getXYZfromLatLonDegrees(
                            latitude,
                            longitude,
                            2.0
                        )
                    espduLocation.x = disCoordinates[0]
                    espduLocation.y = disCoordinates[1]
                    espduLocation.z = disCoordinates[2]
                    espdu.entityLocation = espduLocation
                    var entityType = EntityType()
                    entityType.country = Country.GERMANY_DEU
                    entityType.entityKind = EntityKind.PLATFORM
                }
            }
        }
    }
}
```

```

entityType.domain = Domain.inst(PlatformDomain.LAND)
entityType.category = 1.toByte() //tank
entityType.subCategory = 1.toByte() //something ??
entityType.specific = 1.toByte() //something ??
espdu.entityType = entityType
Log.e("lat", "" + latitude)
Log.e("long", "" + longitude)
Log.e("alt", "" + altitude)
var timeStamp = DisTime().disAbsoluteTimestamp
espdu.timestamp = timeStamp
val baos = ByteArrayOutputStream()
val dos = DataOutputStream(baos)
espdu.marshal(dos)
val data = baos.toByteArray()
var ina = InetAddress.getByName("192.168.188.106")
var dp = DatagramPacket(data, data.size)
dp.setAddress(ina)
dp.setPort(3000)
socket.send(dp)
socket.close()
Toast.makeText(this@MainActivity, "try...")
}
}
}

```

After compiling and installing the software on my mobile phone (Samsung J7) it looks like this:



Pressing the “Start” button the mobile creates an ESPDU and send it via Wi-Fi to the laptop (192.168.188.106).

The laptop received the package (see Wireshark capture on 192.168.188.106):

No.	Time	Source	Destination	Protocol	Length	Info
7	3.816358	192.168.188.47	192.168.188.106	DIS	186	PDUType: 1 Entity State, Platform, Land, (0:0:666)

```
> Frame 7: 186 bytes on wire (1488 bits), 186 bytes captured (1488 bits) on interface \Device\NPF_{BD05C314-3911-4ED4-ADFD-88EF820717C6}, id 0
> Ethernet II, Src: SamsungE_c0:e5:34 (88:9f:6f:c0:e5:34), Dst: 32:d1:1a:ee:54:86 (32:d1:1a:ee:54:86)
> Internet Protocol Version 4, Src: 192.168.188.47, Dst: 192.168.188.106
> User Datagram Protocol, Src Port: 3000, Dst Port: 3000
▼ Distributed Interactive Simulation
  > Header
  ▼ Entity State PDU
    ▼ Entity ID
      Entity ID Site: 0
      Entity ID Application: 0
      Entity ID Entity: 666
      Force ID: 0
      Number of Articulation Parameters: 0
    ▼ Entity Type, (1:1:78:1:1:1:0)
      Kind: Platform (1)
      Domain: Land (1)
      Country: Germany (78)
      Category / Land: Tank (1)
      Subcategory: 1
      Specific: 1
      Extra: 0
    ▼ Alternative Entity Type, (0:0:0:0:0:0:0)
      Kind: Other (0)
      Domain: Other (0)
      Country: Other (0)
      Category: 0
      Subcategory: 0
      Specific: 0
      Extra: 0
    > Entity Linear Velocity
    ▼ Entity Location
      X: -2709927.85954189
      Y: -4349067.2103765
      Z: 3784971.91737023
    > Entity Orientation
      Appearance: 0x00000000
    > Dead Reckoning Parameters
    > Entity Marking
      Capabilities: 0
```

The mobile device has the IP 192.168.188.47.

In the class **GermanyEspduReceiverEspduVPNSender** the package is captured, printed out on the console, and send it over a VPN.

This is the console output:

```
[MV3500Cohort2020JulySeptember.homework4.WeissenbergerGoericke.GermanyEspduVPNReceiver] started...
[MV3500Cohort2020JulySeptember.homework4.WeissenbergerGoericke.GermanyEspduVPNReceiver] listening for PDU packets on port 3000
=====
1. received PDU type 1=ENTITY_STATE edu.nps.moves.dis7.pdus.EntityStatePdu
   entityID triplet: [0, 0, 666]
   Location in DIS coordinates:      [-2709914.4707966796, -4349073.305945409, 3784974.4819766334]
2. received PDU type 1=ENTITY_STATE edu.nps.moves.dis7.pdus.EntityStatePdu
   entityID triplet: [0, 0, 666]
   Location in DIS coordinates:      [-2709914.4707966796, -4349073.305945409, 3784974.4819766334]
3. received PDU type 1=ENTITY_STATE edu.nps.moves.dis7.pdus.EntityStatePdu
   entityID triplet: [0, 0, 666]
   Location in DIS coordinates:      [-2709914.4707966796, -4349073.305945409, 3784974.4819766334]
```

The laptop sends the package (see Wireshark capture on 10.1.105.9):

No.	Time	Source	Destination	Protocol	Length	Info
7	3.376665	10.1.105.9	10.1.105.10	DIS	186	PDUType: 1 Entity State, Platform, Land, (0:0:666)

```

> Frame 7: 186 bytes on wire (1488 bits), 186 bytes captured (1488 bits) on interface \Device\NPF_{70714248-BA26-4E1A-8C47-6B68A531C563}, id 0
> Ethernet II, Src: 02:50:41:00:00:02 (02:50:41:00:00:02), Dst: 02:50:41:00:00:02 (02:50:41:00:00:02)
> Internet Protocol Version 4, Src: 10.1.105.9, Dst: 10.1.105.10
  User Datagram Protocol, Src Port: 3000, Dst Port: 3000
    Source Port: 3000
    Destination Port: 3000
    Length: 152
    Checksum: 0xa144 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 0]
    [Timestamps]
  Distributed Interactive Simulation
    Header
    Entity State PDU
      Entity ID
        Entity ID Site: 0
        Entity ID Application: 0
        Entity ID Entity: 666
        Force ID: 0
        Number of Articulation Parameters: 0
      Entity Type, (1:1:78:1:1:1:0)
        Kind: Platform (1)
        Domain: Land (1)
        Country: Germany (78)
        Category / Land: Tank (1)
        Subcategory: 1
        Specific: 1
        Extra: 0
      Alternative Entity Type, (0:0:0:0:0:0:0)
      Entity Linear Velocity
      Entity Location
        X: -2709927.85954189
        Y: -4349067.2103765
        Z: 3784971.91737023
      Entity Orientation
        Appearance: 0x00000000
      Dead Reckoning Parameters
      Entity Marking
        Capabilities: 0
  
```

It is sent to Stefan's laptop with the IP: 10.1.105.10.

On this machine the class **GermanyEspduVPNReceiver** is running. It catches the ESPDU and print some values out.

```

System.out.println(receiptMessage);
EntityID entityID = ((EntityStatePdu) aPdu).getEntityID();
Vector3Double position = ((EntityStatePdu) aPdu).getEntityLocation();
System.out.println("    entityID triplet: [" + entityID.getSiteID() + ", " + entityID.getApplicationID() + ", " + entityID.getEntityID() + "] ");
System.out.println("    Location in DIS coordinates:      [" + position.getX() + ", " + position.getY() + ", " + position.getZ() + "]");
  
```