

# Riconoscitore di Protocolli Non Standard

---

**Autore:** Lorenzo Iannarella

**Email:** l.iannarella@studenti.unpi.it (mailto:l.iannarella@studenti.unpi.it)

## Introduzione

---

Il documento presente descrive `recognition_nsp.lua`, un plugin Wireshark scritto in Lua. Questo strumento è progettato per riconoscere i flussi di dati generati dalle chiamate e videochiamate effettuate tramite le applicazioni come Telegram, WhatsApp, Teams, Meet e Zoom.

Queste applicazioni utilizzano molti protocolli, ma tra questi vi è il protocollo STUN che viene usato per stabilire connessioni tra gli utenti (in quanto permette ad un host di conoscere il proprio indirizzo pubblico).

I pacchetti STUN hanno permesso la realizzazione di questo plugin, in quanto l'identificazione di questi protocolli non standard si basa proprio sull'analisi dei diversi campi contenuti nel layer STUN.

## Descrizione

---

Il plugin `recognition_nsp.lua` si basa su un post-dissector, un tipo specifico di dissector Wireshark, che si occupa dell'analisi dei pacchetti. In particolare, questo plugin si concentra sull'analisi dei pacchetti UDP, poiché le applicazioni menzionate precedentemente utilizzano principalmente il protocollo di trasporto UDP per la trasmissione dei dati.

Per ogni pacchetto UDP, `recognition_nsp.lua` inizia verificando se il pacchetto è un pacchetto STUN. Se lo è, il pacchetto viene analizzato ulteriormente per determinare quale protocollo non standard rappresenta. Successivamente, il flusso - identificato tramite **IP sorgente, porta sorgente, IP destinazione e porta destinazione** - viene registrato nella `flows_table`. Se il pacchetto non è un pacchetto STUN, il plugin verifica se appartiene a un flusso già riconosciuto, cercandolo nella `flows_table`.

Dopo aver completato l'analisi di un pacchetto, vengono inserite nell'albero di dissezione di Wireshark informazioni quali il nome del protocollo non standard identificato. Questo permette di visualizzare tale informazione anche nella *Packet List* di Wireshark.

## Strutture dati utilizzate

```
local request_table = {}

local flows_table = {}

local processed_packets = {}
```

La `flows_table` è stata impiegata per registrare i flussi e i relativi protocolli applicativi non standard associati.

La `request_table` è stata impiegata per conservare informazioni relative ai pacchetti STUN che rappresentano delle *Bind Request*. Questi pacchetti hanno permesso di identificare il protocollo applicativo sottostante. Inoltre, hanno facilitato l'identificazione del protocollo dei pacchetti di risposta, ovvero le *Binding Success Response*.

La `processed_packets` è stata impiegata per tenere traccia dei pacchetti che sono stati già analizzati. Questo è stato fatto per prevenire che il post-dissector li elaborasse nuovamente in maniera diversa, a causa dell'evoluzione delle informazioni raccolte attraverso le analisi effettuate sui pacchetti precedenti e sui flussi identificati.

## Definizione del layer

```
local l7_proto = Proto("l7", "Layer 7 Protocol")
l7_proto.fields = {}
local l7_fds = l7_proto.fields
l7_fds.proto = ProtoField.new("Protocol Name", "l7.proto", ftypes.STRING)
```

Per aggiungere informazioni sai nella *Packet List* che nella *Details packet* è stato definito un protocollo ,ovvero **“Layer 7 Protocol”** e gli è stato aggiunto un campo, **“Protocol Name”**.

## Aggiunta dell’informazione all’albero di dissezione

```
local subtree = tree:add(l7_proto, tvb(), "Application Protocol")
```

Questa istruzione ha consentito l’integrazione del layer appena creato nell’albero di dissezione. Di conseguenza, il layer è ora visibile nella sezione *Details Packet* di Wireshark. Questo assicura che tutte le informazioni pertinenti siano facilmente accessibili e visualizzabili anche nella *Packet List*.

```
subtree:add(l7_fds.proto, protocol)
```

Questa istruzione fa in modo che venga aggiunto nel campo **Protocol Name** il nome del protocollo individuato.

## Prerequisiti

---

Avere **Wireshark** installato sul pc.

## Istruzioni

---

### Aggiungere una colonna alla Packet List

1. **Vai su** `Edit` .
2. **Clicca su** `Preferences` .
3. **Clicca su** `Columns` .
  - 3.1 aggiungi una colonna premendo sul `+` .
  - 3.2 seleziona la spunta sulla riga appena creata .
  - 3.3 Nella colonna `Title` scrivi `Application Protocol` .
  - 3.4 Nella colonna `Type` seleziona `Custom` .
  - 3.5 Nella colonna `Fields` scrivi `l7.proto` .
4. **Clicca** `ok` .

## Caricamento del Plugin

---

### Caricamento dello script da Terminale

#### 1 Trovare la Cartella dei Plugin di Wireshark

Wireshark carica automaticamente gli script Lua dalla sua directory di plugin. Ecco dove puoi trovare questa directory su diversi sistemi operativi:

- **Windows:** C:\Program Files\Wireshark\plugins\<version>\
- **macOS:** /Applications/Wireshark.app/Contents/PlugIns/wireshark/<version>/
- **Linux:** /usr/lib/wireshark/plugins/<version>/

## 2 Copiare lo Script Lua nella Cartella dei Plugin

Copia il tuo script Lua ( recognition\_nsp.lua ) nella directory dei plugin di Wireshark. Se stai usando la directory personale, assicurati che esista e crea la cartella plugins se necessario.

## Caricare lo Script Lua tramite la GUI di Wireshark

1. Apri Wireshark.
2. Clicca su Help .
3. Clicca About Wireshark :
4. Vai in Folders :
5. Clicca sul link individuato da Personal Plugin
6. Inserisci lo script lua

## Riavvia Wireshark

- Chiudi Wireshark e riapilo per caricare il nuovo script Lua, questa operazione indimentemente la modalità di caricamento utilizzata.

## Test

Per effettuare dei test sul funzionamento del plugin sono disponibili dei file pcap contenuti nella cartella test\_pcap.

- Meet

No.	Time	Source	Destination	Application Protocol	Protocol	Info
49	2.927284936	192.168.1.8	142.250.82.220		UDP	34158 → 3478 Len=107
50	2.944818839	192.168.1.8	142.250.82.220		UDP	34158 → 3478 Len=105
51	2.963994680	192.168.1.8	142.250.82.220		UDP	34158 → 3478 Len=100
52	2.980505193	192.168.1.8	142.250.82.220	Meet	STUN	Binding Request user: lS7UPEkTqFwlg0KAAJKYgCIAMQ;aff3c934
53	2.981998778	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=108
54	2.992882788	142.250.82.220	192.168.1.8	Meet	STUN	Binding Success Response XOR-MAPPED-ADDRESS: 2.38.42.220:34158
55	2.994716431	192.168.1.8	142.250.180.142		UDP	34664 → 443 Len=1357
56	2.994748593	192.168.1.8	142.250.180.142		UDP	34664 → 443 Len=1357
57	2.994761019	192.168.1.8	142.250.180.142		UDP	34664 → 443 Len=420
58	2.999334556	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=104
59	2.117562157	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=106
60	2.124526656	142.250.180.142	192.168.1.8		UDP	443 → 34664 Len=36
61	2.124603769	192.168.1.8	142.250.180.142		UDP	34664 → 443 Len=33
62	2.135946931	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=103
63	2.135208197	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=98
64	2.171990791	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=100
65	2.189656906	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=37
66	2.192975286	142.250.180.142	192.168.1.8		UDP	443 → 34664 Len=1222
67	2.192975534	142.250.180.142	192.168.1.8		UDP	443 → 34664 Len=101
68	2.193564106	192.168.1.8	142.250.180.142		UDP	34664 → 443 Len=41
69	2.222646510	142.250.180.142	192.168.1.8		UDP	443 → 34664 Len=29
70	2.142025659	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=107
71	2.261100599	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=112
72	2.279217250	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=115
73	2.297773491	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=116
74	2.316410329	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=111
75	2.334557618	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=110
76	2.352072193	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=118
77	2.370433383	192.168.1.8	142.250.82.220	Meet	UDP	34158 → 3478 Len=118

- Teams

No.	Time	Source	Destination	Application Protocol	Protocol	Info
286	11.041749261	192.168.1.8	52.123.129.14		TCP	58962 → 443 [ACK] Seq=6204 Ack=18545 Win=12317 Len=0
287	11.041794670	192.168.1.8	52.123.129.14		TCP	58962 → 443 [ACK] Seq=6204 Ack=18617 Win=12317 Len=0
288	11.043542511	52.112.100.9	192.168.1.8		TLVsl.2	Application Data
289	11.055445968	52.123.129.14	192.168.1.8		TCP	443 → 58962 [ACK] Seq=18617 Ack=6294 Win=16385 Len=0
290	11.067080750	192.168.1.8	131.114.161.32	Teams	DTLSv1.	Client Hello
291	11.071897510	131.114.161.32	192.168.1.8	Teams	STUN	Binding Success Response XOR-MAPPED-ADDRESS: 2.38.42.220:58188
292	11.084108803	192.168.1.8	52.112.100.9		TCP	37646 → 443 [ACK] Seq=3156 Ack=14366 Win=961 Len=0
293	11.103770593	52.112.100.9	192.168.1.8		TCP	443 → 37646 [ACK] Seq=14366 Ack=3156 Win=2845 Len=0
294	11.119797810	131.114.161.32	192.168.1.8	Teams	DTLSv1.	Server Hello, Certificate, Server Key Exchange, Certificate Request, Server Hello Done
295	11.121399023	192.168.1.8	131.114.161.32	Teams	DTLSv1.	Certificate, Client Key Exchange, Certificate Verify, Change Cipher Spec, Encrypted Handshake Message
296	11.124106960	131.114.161.32	192.168.1.8	Teams	DTLSv1.	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
297	11.235721810	131.114.161.32	192.168.1.8	Teams	UDP	7945 → 58188 Len=85
298	11.245722983	131.114.161.32	192.168.1.8	Teams	SRTPC	Sender Report
299	11.245723030	131.114.161.32	192.168.1.8	Teams	SRTPC	Sender Report
300	11.249219420	192.168.1.8	131.114.161.32	Teams	UDP	58188 → 7945 Len=108
301	11.249270780	192.168.1.8	131.114.161.32	Teams	UDP	58188 → 7945 Len=107
302	11.249527750	192.168.1.8	131.114.161.32	Teams	UDP	58188 → 7945 Len=107
303	11.249719221	192.168.1.8	131.114.161.32	Teams	UDP	58188 → 7945 Len=97
304	11.285716665	131.114.161.32	192.168.1.8	Teams	UDP	7945 → 58188 Len=91
305	11.285716814	131.114.161.32	192.168.1.8	Teams	SRTPC	Sender Report
306	11.285716829	131.114.161.32	192.168.1.8	Teams	SRTPC	Sender Report
307	11.323354754	192.168.1.8	131.114.161.32	Teams	UDP	58188 → 7945 Len=105
308	11.323498105	192.168.1.8	131.114.161.32	Teams	UDP	58188 → 7945 Len=101
309	11.340183783	131.114.161.32	192.168.1.8	Teams	UDP	7945 → 58188 Len=88
310	11.359603422	192.168.1.8	131.114.161.32	Teams	UDP	58188 → 7945 Len=101
311	11.359677479	192.168.1.8	131.114.161.32	Teams	UDP	58188 → 7945 Len=104
312	11.361877183	131.114.161.32	192.168.1.8	Teams	SRTPC	Sender Report
313	11.361877238	131.114.161.32	192.168.1.8	Teams	SRTPC	Sender Report

- telegram

No.	Time	Source	Destination	Application Protocol	Protocol	Info
244	14.873648370	192.168.1.8	149.154.167.92		SSL	Continuation Data
245	14.874489785	192.168.1.8	8.8.8.8		DNS	Standard query 0xa4d7 A reflector-3-1659348528.reflector OPT
246	14.874566670	192.168.1.8	8.8.8.8		DNS	Standard query 0xa421 AAAA reflector-3-1659348528.reflector OPT
247	14.890029332	192.168.1.8	131.114.161.32	Telegram	STUN	Binding Request user: ZwYCKyNp
248	14.890787290	149.154.167.92	192.168.1.8		TCP	443 → 41116 [PSH, ACK] Seq=28367 Ack=21494 Win=16178 Len=89 TSval=1669350020 TSecr=3546569341 [TCP segment of a re
249	14.890652155	91.108.13.57	192.168.1.8		UDP	596 → 34869 Len=120
250	14.900708289	8.8.8.8	192.168.1.8		DNS	Standard query response 0xa4d7 No such name A reflector-3-1659348528.reflector SOA a.root-servers.net OPT
251	14.900841478	192.168.1.8	8.8.8.8		DNS	Standard query 0xa4d7 A reflector-3-1659348528.reflector
252	14.900861423	192.168.1.8	91.108.13.57		UDP	34869 → 596 Len=88
253	14.910660555	91.108.9.1	192.168.1.8		UDP	596 → 42929 Len=120
254	14.910660683	192.168.1.8	91.108.9.1		UDP	42929 → 596 Len=88
255	14.910782637	8.8.8.8	192.168.1.8		DNS	Standard query response 0xa421 No such name AAAA reflector-3-1659348528.reflector SOA a.root-servers.net OPT
256	14.917032240	192.168.1.8	8.8.8.8		DNS	Standard query 0xa421 AAAA reflector-3-1659348528.reflector
257	14.920834610	131.114.161.32	192.168.1.8	Telegram	STUN	Binding Success Response XOR-MAPPED-ADDRESS: 2.38.42.220:45741
258	14.925320818	91.108.13.57	192.168.1.8		UDP	596 → 34869 Len=88
259	14.925321020	131.114.161.32	192.168.1.8	Telegram	STUN	Binding Request user: KynpZwYc
260	14.925322223	149.154.167.92	192.168.1.8		TCP	443 → 41116 [PSH, ACK] Seq=28367 Ack=21494 Win=16255 Len=89 TSval=1669350047 TSecr=3546569360 [TCP segment of a re
261	14.925321114	131.114.161.32	192.168.1.8	Telegram	STUN	Binding Success Response XOR-MAPPED-ADDRESS: 2.38.42.220:45741
262	14.925403175	192.168.1.8	149.154.167.92		TCP	41116 → 443 [ACK] Seq=21494 Ack=28456 Win=11442 Len=0 TSval=13045060430 TSecr=1669350020
263	14.925520857	192.168.1.8	131.114.161.32	Telegram	STUN	Binding Success Response XOR-MAPPED-ADDRESS: 131.114.161.32:57314
264	14.932686323	91.108.9.1	192.168.1.8		UDP	596 → 42929 Len=588
265	14.934778187	8.8.8.8	192.168.1.8		DNS	Standard query 0xa4d7 No such name A reflector-3-1659348528.reflector SOA a.root-servers.net
266	14.935959226	192.168.1.8	131.114.161.32	Telegram	DTLSv1.	Application Data
267	14.940705012	8.8.8.8	192.168.1.8		DNS	Standard query response 0xa421 No such name AAAA reflector-3-1659348528.reflector SOA a.root-servers.net
268	14.940705230	91.108.9.1	192.168.1.8		UDP	596 → 42929 Len=109
269	14.940901324	192.168.1.8	131.114.161.32	Telegram	DTLSv1.	Application Data
270	14.947196831	192.168.1.8	8.8.8.8		DNS	Standard query 0xbf72 A reflector-3-1659348528.reflector OPT
271	14.947250618	192.168.1.8	8.8.8.8		DNS	Standard query 0xb5b1d AAAA reflector-3-1659348528.reflector OPT
272	14.950825443	192.168.1.8	131.114.161.32	Telegram	UDP	45741 → 57314 Len=256
273	14.960979124	192.168.1.8	131.114.161.32	Telegram	UDP	45741 → 57314 Len=58
274	14.960979404	192.168.1.8	131.114.161.32	Telegram	UDP	45741 → 57314 Len=126

- Zoom

No.	Time	Source	Destination	Application Protocol	Protocol	Info
31	3.225.189.90	192.168.1.117	13.225.84.162		TCP	54798 → 443 [ACK] Seq=1 Ack=1 Win=4134 Len=0
32	3.009892	192.168.1.117	92.202.62.238		TLV1.2	Client Hello (SN=log.zoom.us)
33	3.055769	192.168.1.117	192.168.1.117		TCP	443 → 54798 [RST] Seq=1 Win=0 Len=0
34	3.109036	192.168.1.117	192.255.38.14		CLASSI.	Message: Binding Request
35	3.117998	52.202.62.238	192.168.1.117	Zoom	TCP	443 → 54864 [ACK] Seq=1 Ack=518 Win=28672 Len=0
36	3.119256	192.168.1.117	162.255.37.14	Zoom	CLASSI.	Message: Binding Request
37	3.119716	52.202.62.238	192.168.1.117		TLV1.2	Server Hello
38	3.120690	52.202.62.238	192.168.1.117		TCP	443 → 54864 [ACK] Seq=1453 Ack=518 Win=28672 Len=1452 [TCP segment of a reassembled PDU]
39	3.120123	192.168.1.117	52.202.62.238		TCP	54864 → 443 [ACK] Seq=518 Ack=2905 Win=259232 Len=0
40	3.120218	52.202.62.238	192.168.1.117		TCP	443 → 54864 [ACK] Seq=2905 Ack=518 Win=28672 Len=1452 [TCP segment of a reassembled PDU]
41	3.120232	52.202.62.238	192.168.1.117		TLV1.2	Certificate, Server Key Exchange, Server Hello Done
42	3.120269	192.168.1.117	52.202.62.238		TCP	54864 → 443 [ACK] Seq=518 Ack=5609 Win=259532 Len=0
43	3.129384	192.168.1.117	92.202.62.238		TCP	[TCP Window Update] 54864 → 443 [ACK] Seq=518 Ack=5609 Win=261792 Len=0
44	3.129387	192.168.1.117	162.255.37.14	Zoom	CLASSI.	Message: Binding Request
45	3.129313	192.168.1.117	52.202.62.238		TLV1.2	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message, Encrypted Handshake Message
46	3.140342	192.168.1.117	162.255.38.14	Zoom	CLASSI.	Message: Binding Request
47	3.160726	192.168.1.117	162.255.38.14	Zoom	CLASSI.	Message: Binding Request
48	3.161209	192.168.1.117	162.255.38.14	Zoom	CLASSI.	Message: Binding Request
49	3.173221	192.168.1.117	162.255.38.14	Zoom	CLASSI.	Message: Binding Request
50	3.169808	192.168.1.117	162.255.38.14	Zoom	CLASSI.	Message: Binding Request
51	3.194106	192.168.1.117	162.255.38.14	Zoom	CLASSI.	Message: Binding Request
52	3.240903	52.202.62.238	192.168.1.117		TLV1.2	Change Cipher Spec, Encrypted Handshake Message
53	3.249133	192.168.1.117	52.202.62.238		TCP	54864 → 443 [ACK] Seq=709 Ack=5608 Win=262080 Len=0
54	3.242563	192.168.1.117	52.202.62.238		TLV1.2	Application Data
55	3.260899	192.168.1.117	164.199.65.42		TCP	53867 → 80 [PSH, ACK] Seq=1 Ack=1 Win=4096 Len=263 TSval=631036707 TSecr=3017133313
56	3.261213	192.168.1.117	35.186.224.53		TCP	53872 → 443 [ACK] Seq=1 Ack=1 Win=4096 Len=1360 TSval=631036706 TSecr=2242352867 [TCP segment of a reassembled PDU]
57	3.261216	192.168.1.117	35.186.224.53		TLV1.2	Application Data
58	3.273602	35.186.224.53	192.168.1.117		TCP	443 → 53872 [ACK] Seq=1 Ack=1369 Win=1015 Len=0 TSval=2242373817 TSecr=631036708
59	3.273621	35.186.224.53	192.168.1.117		TCP	443 → 53872 [ACK] Seq=1 Ack=1444 Win=1015 Len=0 TSval=2242373817 TSecr=631036708
60	3.276157	162.255.37.14	192.168.1.117	Zoom	CLASSI.	Message: Shared Secret Request
61	3.285500	162.255.37.14	192.168.1.117	Zoom	CLASSI.	Message: Shared Secret Request
62	3.290102	164.199.65.42	192.168.1.117		TCP	80 → 53867 [PSH, ACK] Seq=1 Ack=264 Win=42 Len=55 TSval=3017154259 TSecr=631036707
63	3.298184	192.168.1.117	164.199.65.42		TCP	53867 → 80 [ACK] Seq=264 Ack=56 Win=4094 Len=0 TSval=631036735 TSecr=3017154259
64	3.298964	162.255.37.14	192.168.1.117	Zoom	CLASSI.	Message: Shared Secret Request
65	3.298406	162.255.38.14	192.168.1.117	Zoom	CLASSI.	Message: Shared Secret Request

- WhatsApp

No.	Time	Source	Destination	Application Protocol	Protocol	Info
10	0.916198	10.8.0.1	184.173.179.46		SSL	Continuation Data
11	0.916381	184.173.179.46	10.8.0.1		TCP	443 → 35480 [ACK] Seq=79 Ack=231 Win=65535 Len=0
12	1.110180	184.173.179.46	10.8.0.1		SSL	Continuation Data
13	1.314148	10.8.0.1	184.173.179.46		TCP	35480 → 443 [ACK] Seq=231 Ack=248 Win=15544 Len=0
14	1.314239	10.8.0.1	184.173.179.46		SSL	Continuation Data
15	1.314361	184.173.179.46	10.8.0.1		TCP	443 → 35480 [ACK] Seq=248 Ack=263 Win=65535 Len=0
16	1.304746	10.8.0.1	184.173.179.46		SSL	Continuation Data
17	1.304929	184.173.179.46	10.8.0.1		TCP	443 → 35480 [ACK] Seq=248 Ack=313 Win=65535 Len=0
18	1.751464	10.8.0.1	184.173.179.46		SSL	Continuation Data
19	1.751617	184.173.179.46	10.8.0.1		TCP	443 → 35480 [ACK] Seq=248 Ack=669 Win=65535 Len=0
20	1.805984	184.173.179.46	10.8.0.1		SSL	Continuation Data
21	1.805987	10.8.0.1	184.173.179.46		TCP	35480 → 443 [ACK] Seq=669 Ack=271 Win=15544 Len=0
22	1.856170	184.173.179.46	10.8.0.1		SSL	Continuation Data
23	1.906433	10.8.0.1	184.173.179.46		TCP	35480 → 443 [ACK] Seq=669 Ack=292 Win=15544 Len=0
24	2.254406	184.173.179.46	10.8.0.1		SSL	Continuation Data
25	2.306396	10.8.0.1	184.173.179.46		TCP	35480 → 443 [ACK] Seq=669 Ack=787 Win=16616 Len=0
26	3.338134	10.8.0.1	31.13.64.48	Whatsapp	STUN	Allocate Request
27	3.675904	31.13.64.48	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:53889
28	6.769890	10.8.0.1	31.13.64.48	Whatsapp	STUN	Allocate Request
29	8.968832	31.13.64.48	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:35711
30	8.969154	10.8.0.1	31.13.74.48	Whatsapp	STUN	Allocate Request
31	9.196319	31.13.74.48	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:37639
32	1.308991	10.8.0.1	31.13.74.48	Whatsapp	STUN	Allocate Request
33	9.425750	31.13.74.48	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:44844
34	9.425842	10.8.0.1	31.13.64.48	Whatsapp	STUN	Allocate Request
35	9.729577	31.13.64.48	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:46396
36	9.725769	10.8.0.1	31.13.64.48	Whatsapp	STUN	Allocate Request
37	10.625573	31.13.64.48	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:34235
38	19.126197	10.8.0.1	173.252.121.1	Whatsapp	STUN	Allocate Request
39	10.270214	173.252.121.1	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:42912
40	10.270275	10.8.0.1	173.252.121.1	Whatsapp	STUN	Allocate Request
41	10.506042	173.252.121.1	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:46535
42	10.506184	10.8.0.1	179.60.192.48	Whatsapp	STUN	Allocate Request
43	10.795054	179.60.192.48	10.8.0.1	Whatsapp	STUN	Allocate Success Response XOR-MAPPED-ADDRESS: 191.162.107.92:32848
44	10.795745	10.8.0.1	179.60.192.48	Whatsapp	STUN	Allocate Request