

به نام خدا

# گزارش آزمایش سوم

دستیار آموزشی

آقای سیامکی

اعضای گروه

ایمان محمدی

۹۹۱۰۲۲۰۷

نگار باباشاه

۹۹۱۰۹۳۲۵

محمد مهدی میرزایی

۹۹۱۷۱۰۲۲

نیم سال تابستان ۱۴۰۳

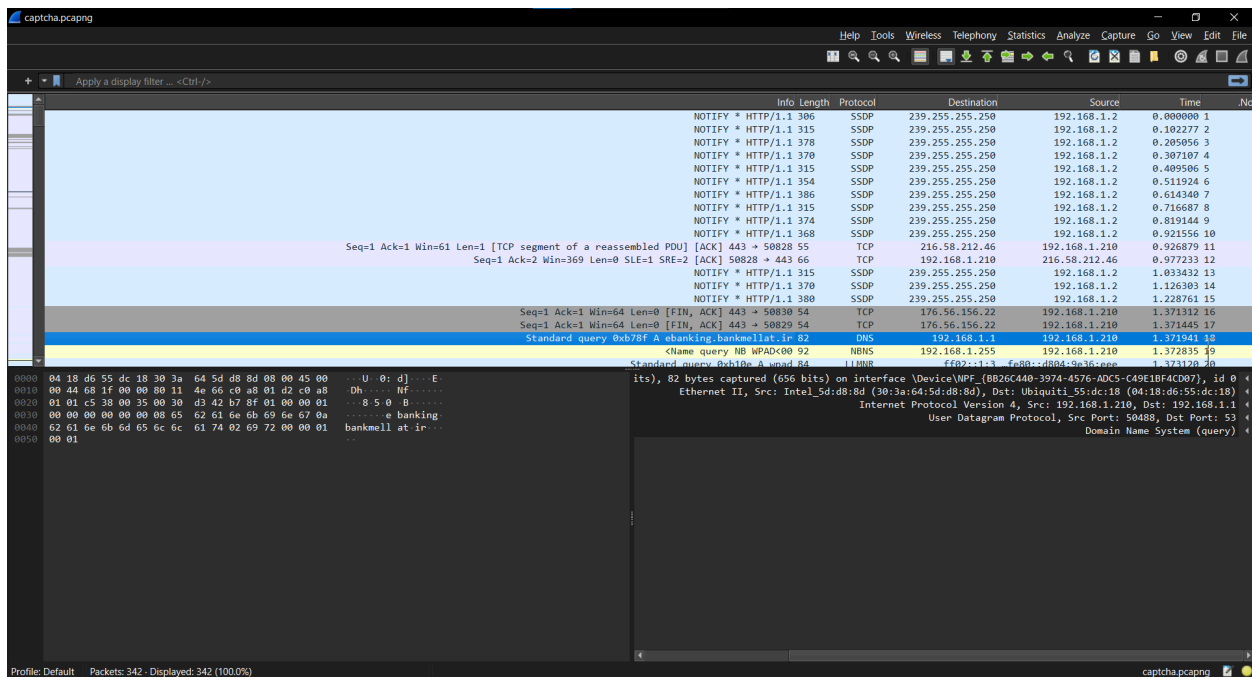
# بخش اول. به دست آوردن captcha

## شرح آزمایش

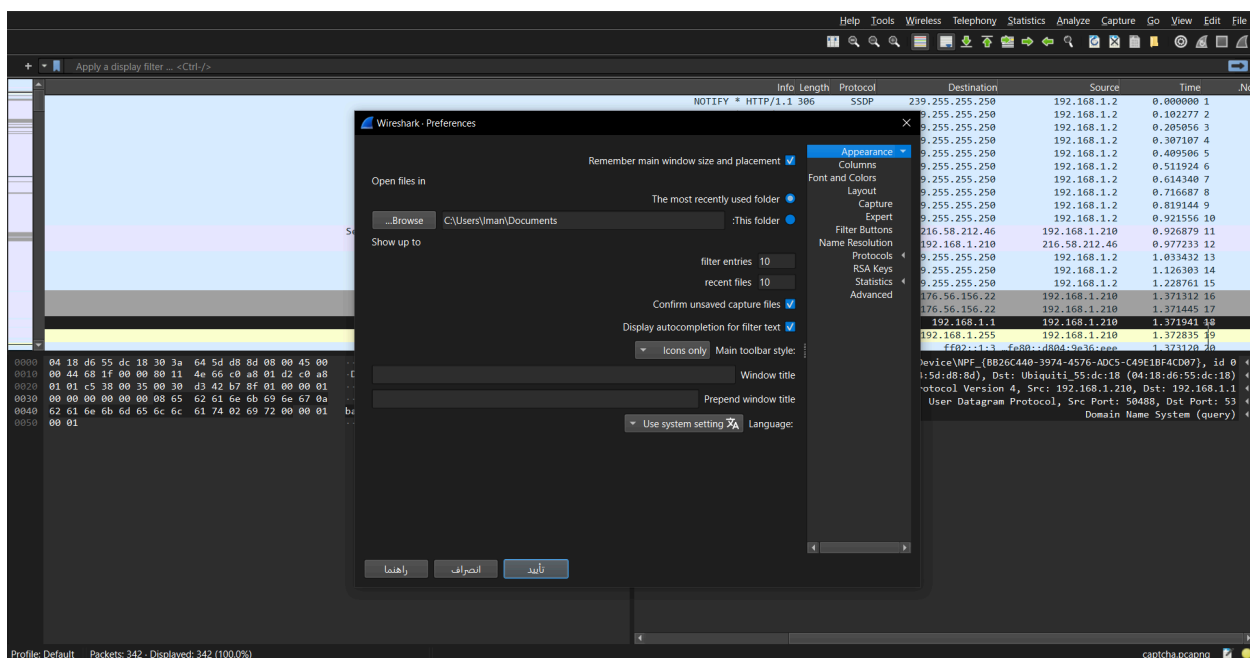
در این بخش هدف این است که با یک سری از قابلیت‌های کاربردی نرم‌افزار Wireshark، آشنا شویم.

## مراحل

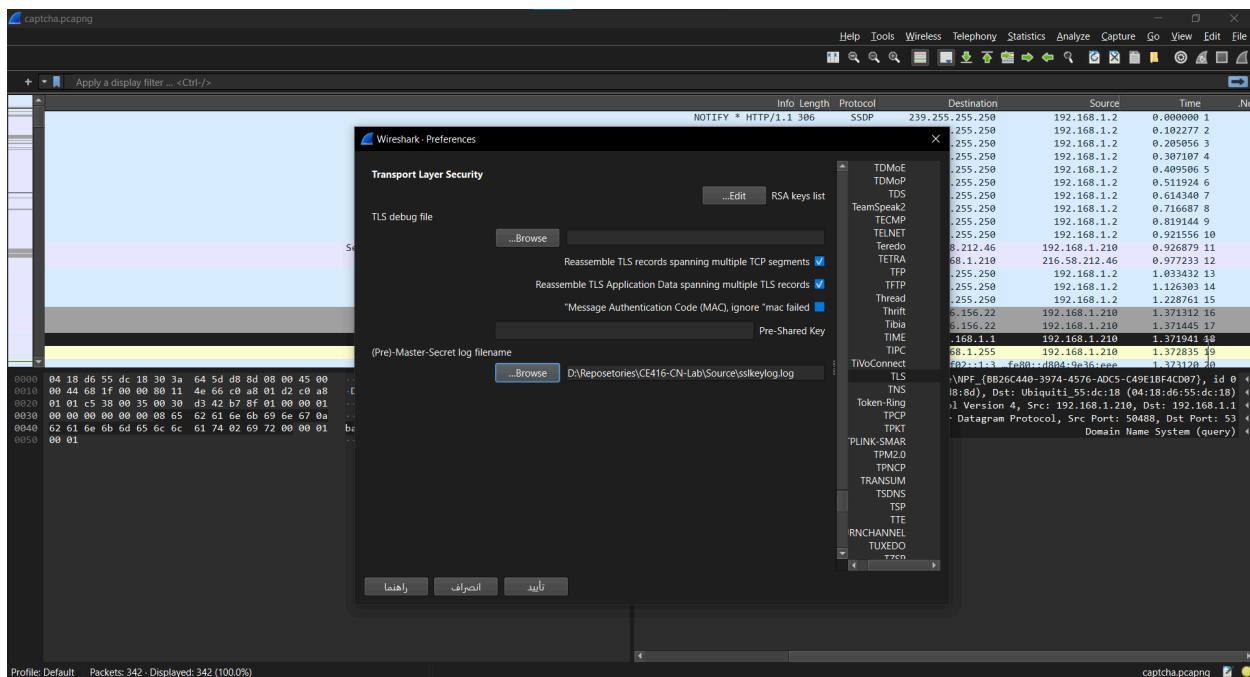
ابتدا فایل captcha.pcapng را در نرم‌افزار Wireshark باز می‌کنیم. تعداد زیادی رکورد مشاهده می‌شود پس از باز کردن فایل مربوطه که در اسکرین‌شات، مشخص است.



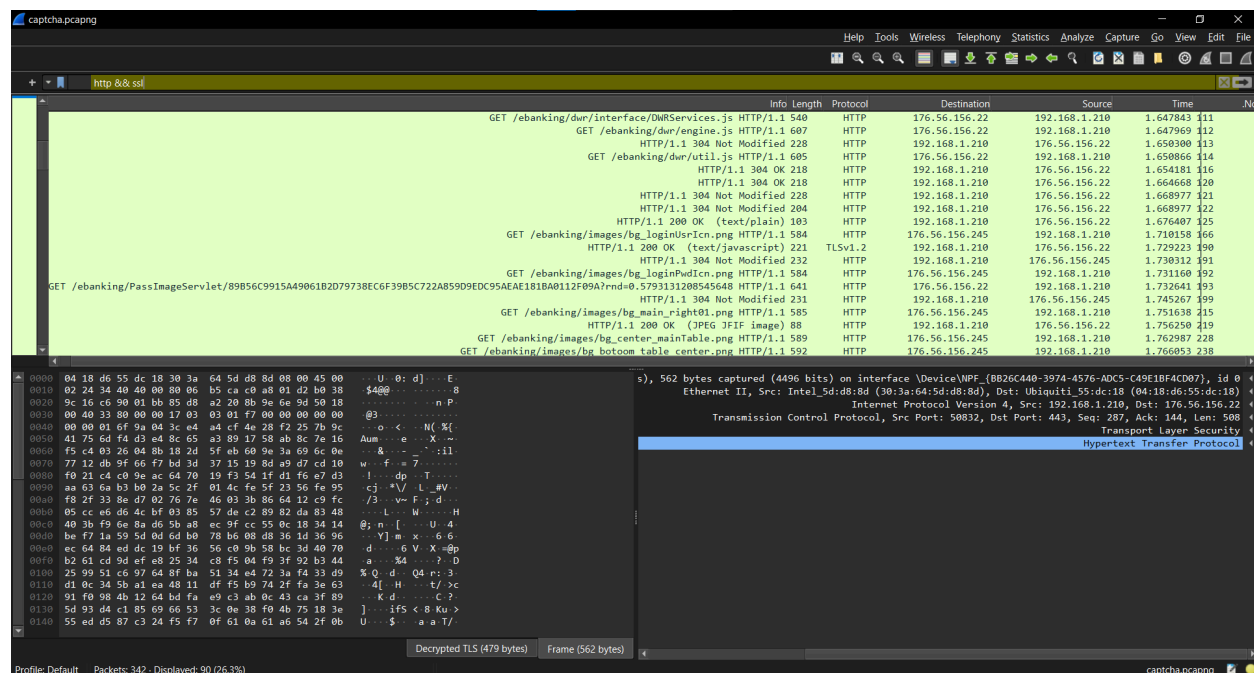
حالا طبق گزارش کار جلو رفته و با انتخاب گزینه‌ی Edit و سپس گزینه‌ی Preferences، به این صفحه می‌رسیم:



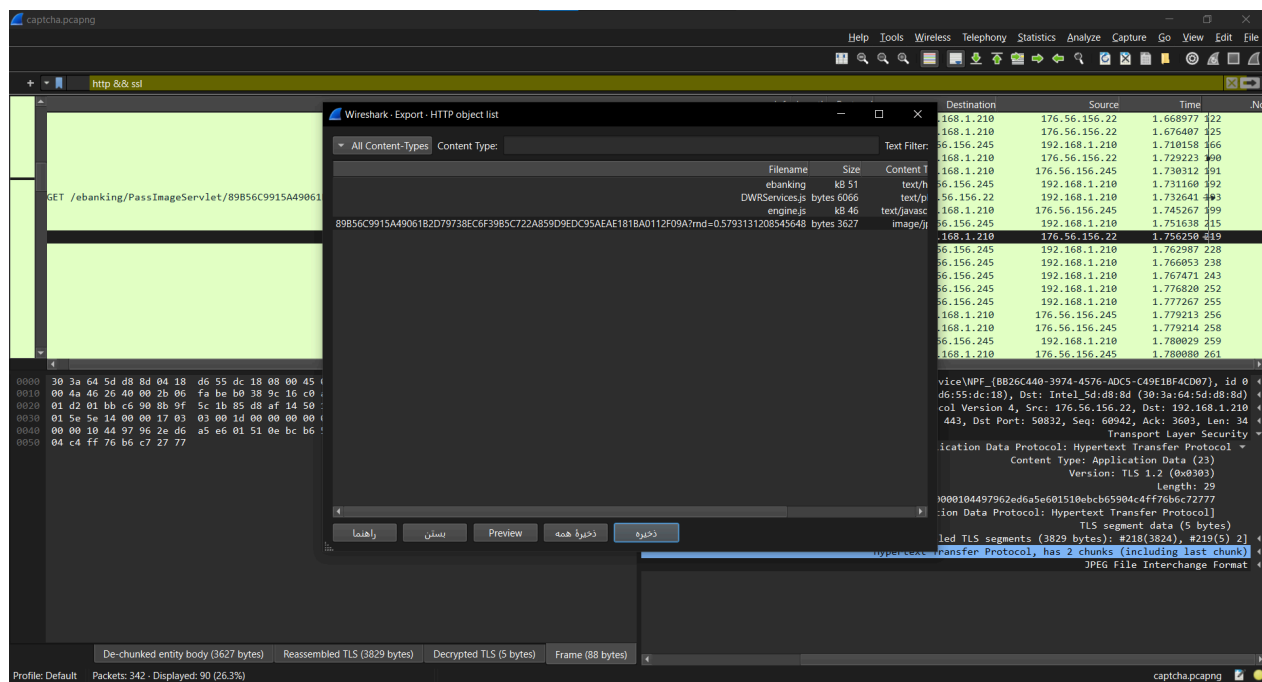
حالا این‌جا به قسمت Protocols رفته و از بین پروتکل‌های موجود باید پروتکل SSL را طبق گزارش انتخاب کنیم اما از اون‌جایی که این پروتکل در نسخه‌های بروز Wireshark وجود ندارد، به‌جای آن پروتکل TLS را انتخاب می‌کنیم.



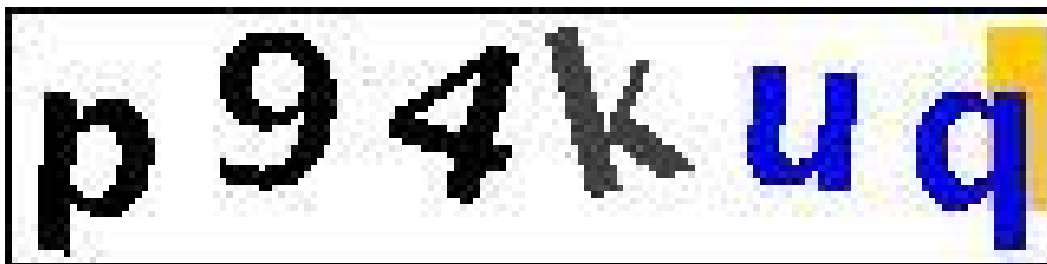
حالا پس از این کار، لیستی از ترافیک رد و بدل شده را مشاهده خواهیم کرد. طبق گزارش، باید این جا فیلتر کنیم این لیست را در نتیجه در قسمت فیلتر، عبارت http && ssl را وارد می‌کنیم و فیلتر می‌کنیم. با این کار، بسته‌هایی که از طریق پروتکل TLS رد و بدل شده‌اند و همچنین توانستیم آن‌ها را رمزگشایی کنیم، به دست خواهند آمد.



سپس به بخش File رفته و گزینه‌ی Objects Export و سپس گزینه‌ی HTTP را انتخاب می‌کنیم. بین ۴ تا رکوردی که مشاهده می‌کنیم در این صفحه، تصویر داده شده رکورد ۴ام هستش که آن را در مکان مورد نظر خودمان با اسم دل‌خواه ذخیره می‌کنیم.



تصویر ذخیره شده، تصویر زیر می‌باشد که مقدار captcha داده شده از طرف بانک ملت به کاربر است.



## پاسخ سوالات:

### سوال ۱، بدست آوردن اطلاعات آماری بسته‌ها

با انتخاب گزینه‌ی statistics در نرم‌افزار Wireshark، لیستی از تمام گزینه‌ها که هر کدام شامل صفحه‌ای حاوی اطلاعاتی مخصوص به خود هستند، باز می‌شود که می‌توانیم با انتخاب هر کدام از گزینه‌ها نیز به اطلاعات مربوطه دسترسی داشته باشیم.

**Capture File Properties:** اطلاعات کلی در مورد فایل ضبط شده، مانند نام، مسیر، اندازه، تاریخ و زمان ضبط، و فرمت فایل را نمایش می‌دهد.

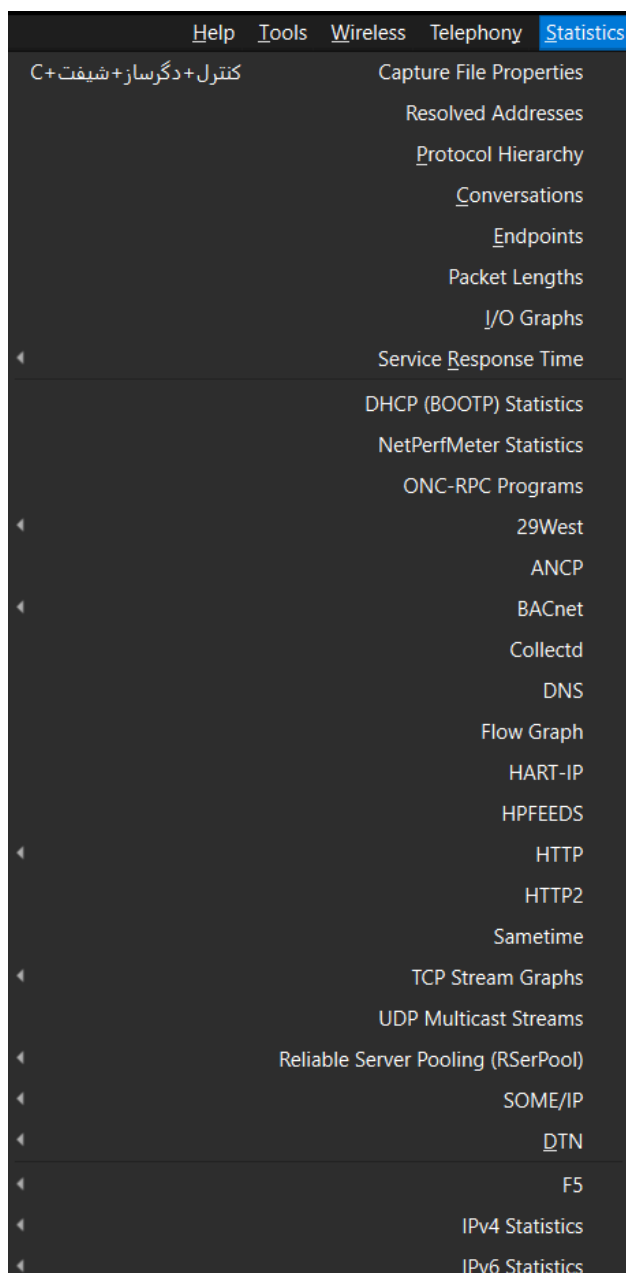
**Protocol Hierarchy:** سلسله مراتبی از پروتکل‌های استفاده شده در ضبط را نمایش می‌دهد.

**Conversations:** لیستی از مکالمات بین دو میزبان را نمایش می‌دهد.

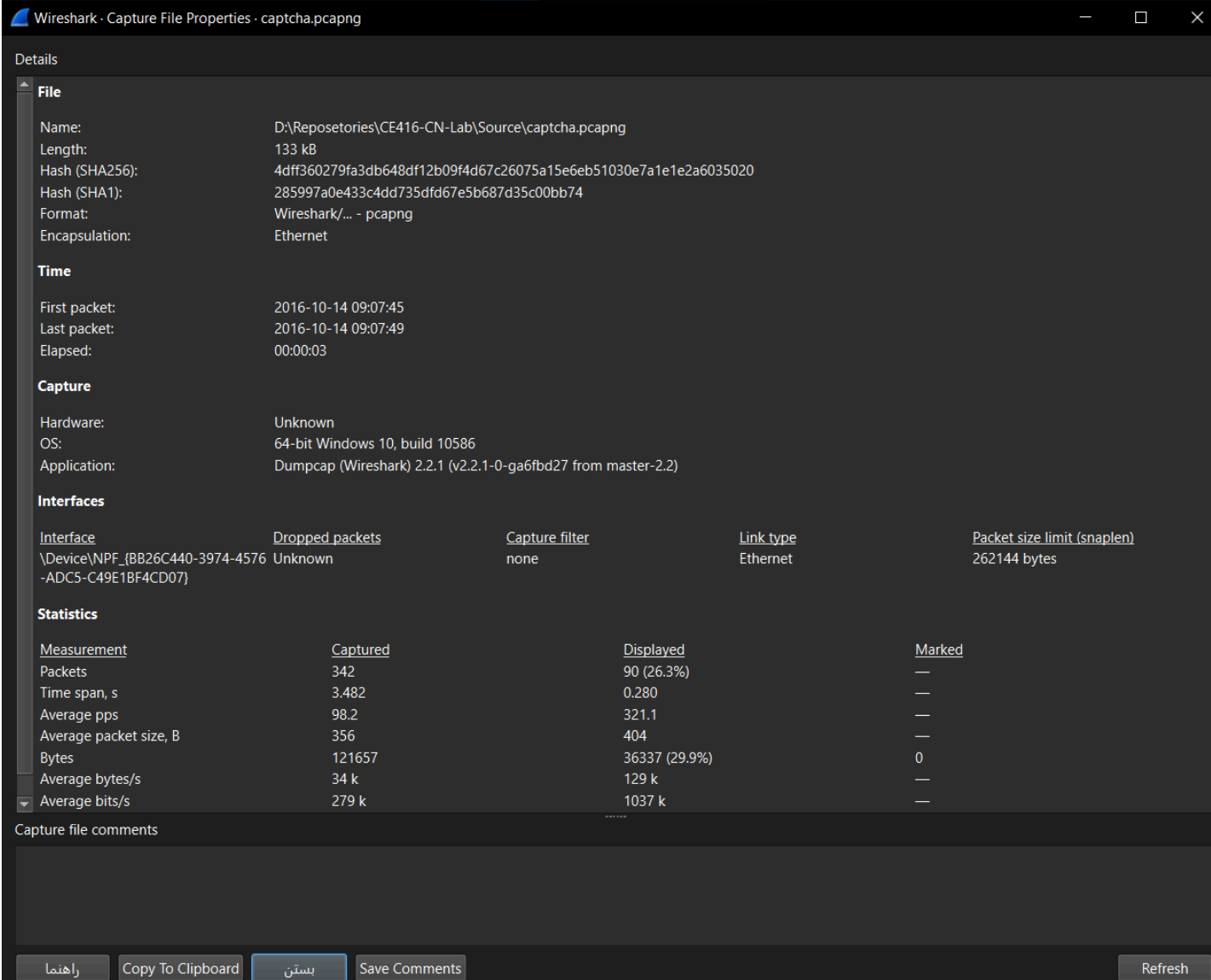
**Endpoints:** لیستی از آدرس‌های IP و پورت‌های استفاده شده در ضبط را نمایش می‌دهد.

**IO Graphs:** نمودارهایی از حجم ترافیک ورودی و خروجی را در طول زمان نمایش می‌دهد.

**Expert Filters:** لیستی از فیلترهای تخصصی را نمایش می‌دهد که می‌توانید برای تجزیه و تحلیل دقیق‌تر ترافیک از آنها استفاده کنید.



حالا برای مثال، پنجره‌ی Capture File Properties را باز می‌کنیم و اطلاعات را با توجه به شکل پایین، بررسی می‌کنیم.



The image shows the 'Wireshark · Capture File Properties · captcha.pcapng' window. It is divided into several sections: 'File', 'Time', 'Capture', 'Interfaces', 'Statistics', and 'Capture file comments'. The 'File' section shows the file name, length (133 kB), SHA256 and SHA1 hashes, format (Wireshark/... - pcapng), and encapsulation (Ethernet). The 'Time' section shows the first and last packet times and the elapsed time (00:00:03). The 'Capture' section shows hardware (Unknown), OS (64-bit Windows 10, build 10586), and application (Dumpcap (Wireshark) 2.2.1). The 'Interfaces' section shows a table with columns for Interface, Dropped packets, Capture filter, Link type, and Packet size limit. The 'Statistics' section shows a table with columns for Measurement, Captured, Displayed, and Marked. The 'Capture file comments' section is empty. At the bottom, there are buttons for 'راهنما', 'Copy To Clipboard', 'بستن', 'Save Comments', and 'Refresh'.

| Interface  | Dropped packets | Capture filter | Link type | Packet size limit (snaplen) |
|--|-----------------|----------------|-----------|-----------------------------|
| \Device\NPF_{BB26C440-3974-4576-ADC5-C49E1BF4CD07} | Unknown         | none           | Ethernet  | 262144 bytes                |

| Measurement            | Captured | Displayed     | Marked |
|------------------------|----------|---------------|--------|
| Packets                | 342      | 90 (26.3%)    | —      |
| Time span, s           | 3.482    | 0.280         | —      |
| Average pps            | 98.2     | 321.1         | —      |
| Average packet size, B | 356      | 404           | —      |
| Bytes                  | 121657   | 36337 (29.9%) | 0      |
| Average bytes/s        | 34 k     | 129 k         | —      |
| Average bits/s         | 279 k    | 1037 k        | —      |

### پنجره Capture File Properties در Wireshark

با استفاده از پنجره Capture File Properties می‌توانیم اطلاعات کلی در مورد فایل ضبط شده خود را به دست آوریم. این اطلاعات می‌تواند برای عیب یابی شبکه، تجزیه و تحلیل عملکرد و بررسی امنیت شبکه مفید باشد.

## توضیح اطلاعات:

- **File:** این بخش اطلاعات کلی در مورد فایل ضبط شده را نمایش می‌دهد، مانند:
  - **Name:** نام فایل
  - **Path:** مسیر فایل
  - **Size:** اندازه فایل بر حسب بایت
  - **Date:** تاریخ ضبط
  - **Time:** زمان ضبط
  - **Format:** فرمت فایل (مانند pcap یا pcapng)
- **Time:** این بخش اطلاعات مربوط به بازه زمانی ضبط را نمایش می‌دهد، مانند:
  - **First Packet:** زمان اولین بسته ضبط شده
  - **Last Packet:** زمان آخرین بسته ضبط شده
  - **Elapsed:** مدت زمان ضبط بر حسب ثانیه
- **Capture Environment:** این بخش اطلاعات مربوط به محیط ضبط را نمایش می‌دهد، مانند:
  - **Capture Interface:** نام رابط شبکه‌ای که ضبط از آن انجام شده است
  - **Capture Filter:** فیلتری که در هنگام ضبط اعمال شده است
  - **Link-Layer Type:** نوع لایه پیوند داده‌ای (مانند Ethernet یا Wi-Fi)
- **Comment:** این بخش برای افزودن نظرات در مورد فایل ضبط شده در نظر گرفته شده است.
- **Capture File Properties Refresh:** این دکمه برای به روز رسانی اطلاعات نمایش داده شده در پنجره استفاده می‌شود.

## نکات:

- اطلاعات نمایش داده شده در پنجره Capture File Properties ممکن است بسته به فرمت فایل ضبط شده و تنظیمات Wireshark متفاوت باشد.
- می‌توانیم با کلیک راست بر روی هر یک از بخش‌ها و انتخاب گزینه‌ی **Copy**، اطلاعات آن بخش را کپی کنیم.
- برای ذخیره اطلاعات پنجره Capture File Properties به عنوان یک فایل متنی، می‌توانیم از گزینه‌ی **File > Export As Text** استفاده کنیم.



## سوال ۲، پروتکل RTP و استفاده از Wireshark برای تحلیل آن

پروتکل RTP یا Real-time Transport Protocol یک پروتکل شبکه است که برای انتقال داده‌های رسانه‌ای به صورت بلادرنگ (Real-time) و از انتها به انتها (End-to-end) طراحی شده است. این پروتکل بیشتر در برنامه‌های استریم مدیا مانند ویدئو کنفرانس، پخش زنده، و تماس‌های صوتی اینترنتی استفاده می‌شود.

### ویژگی‌های کلیدی پروتکل RTP:

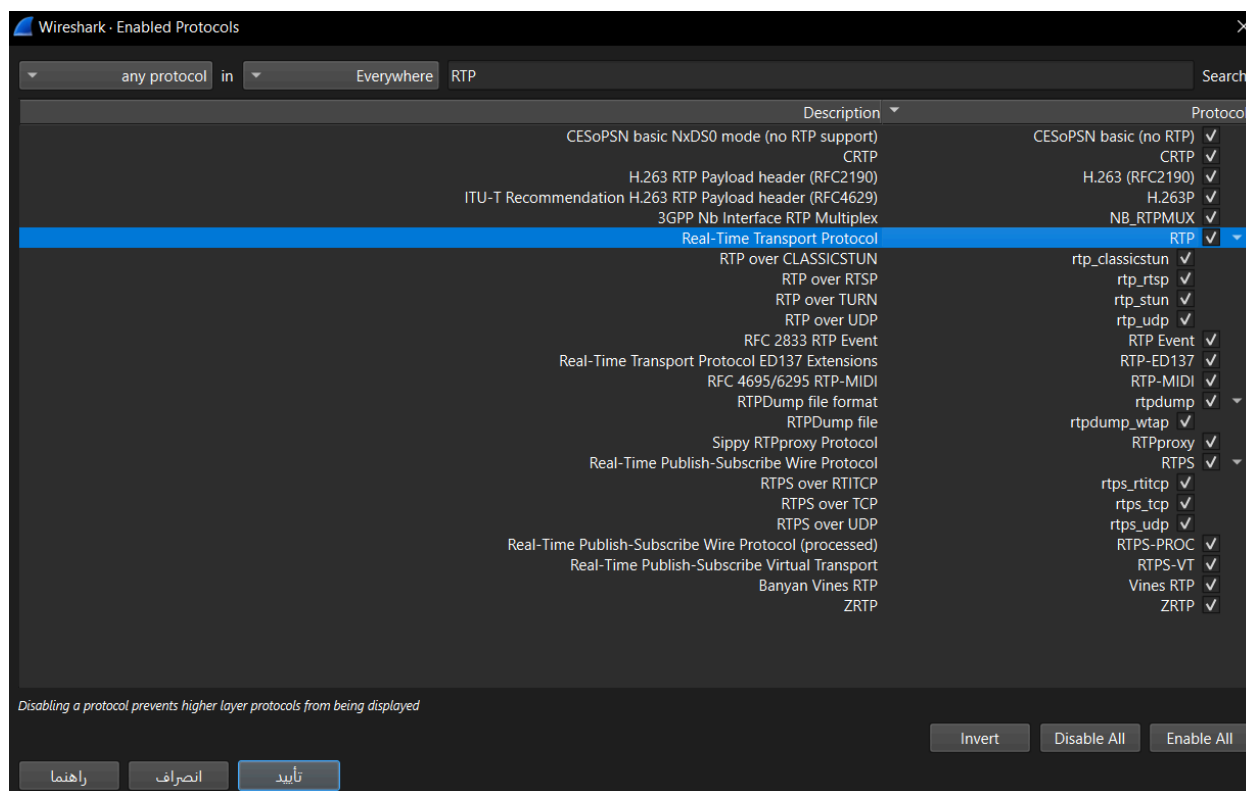
- **بازسازی زمان‌بندی:** RTP از مهر زمان (Timestamp) برای همگام‌سازی جریان‌های صوتی و تصویری استفاده می‌کند. این امر به گیرنده اجازه می‌دهد تا زمان‌بندی اصلی که منبع در نظر گرفته را بازسازی کند.
- **شناسایی نوع محتوا:** RTP اطلاعات مربوط به نوع محتوا را حمل می‌کند تا گیرنده فرمت رسانه دریافتی (مانند نوع کدک) را تشخیص دهد.
- **شماره‌گذاری دنباله:** RTP از شماره‌های دنباله برای تشخیص از دست رفتن بسته‌ها و بازگرداندن ترتیب بسته‌ها که ممکن است در طول انتقال مختل شود، استفاده می‌کند.
- **نظارت بر تحویل:** RTP از مکانیسم‌های نظارت بر کیفیت سرویس (QoS) پشتیبانی می‌کند و بازخورد عملکرد را از طریق پروتکل کنترل بلادرنگ (RTCP) ارائه می‌دهد.

### پروتکل RTP در عمل:

- **مدیریت جلسه:** RTP به طور خودکار مدیریت جلسات را انجام نمی‌دهد و مکانیسمی برای راه‌اندازی و تخریب کانال ارائه نمی‌دهد. در عوض، بر روی پروتکل‌های زیربنایی مانند Session Initiation Protocol یا Session Description Protocol برای مدیریت جلسه تکیه دارد.
- **انتقال:** RTP معمولاً از طریق UDP یا User Datagram Protocol برای استفاده از ویژگی‌های زمان پاسخ کم (Low latency) آن استفاده می‌شود، هرچند که می‌توان آن را با سایر پروتکل‌های انتقال نیز به کار برد.

### تحلیل پروتکل RTP با Wireshark:

- **فعال‌سازی تحلیل RTP:** برای تحلیل پروتکل RTP در Wireshark، به منوی **Analyze** بروید و سپس **Enabled Protocols** را انتخاب کنید و RTP را فعال کنید.



- **نمونه تحلیل:** برای مثال، می‌توانیم به صفحه‌ای که از RTP استفاده می‌کند برویم (مانند Google Meet) و با برقراری تماس صوتی یا اشتراک‌گذاری تصویر، ترافیک RTP را ضبط کنیم. ممکن است البته نیاز باشد اسکرین‌شیر شود یا حتماً وبکم داده شود تا امتحان برقراری این ارتباط صورت گیرد.
- **ابزارهای تحلیل RTP: Wireshark** در منوی **Telephony** و بخش **RTP** ابزارهای مختلفی برای تحلیل RTP ارائه می‌دهد.
  - **RTP Streams:** این ابزار برای جداسازی جریان‌های RTP و تحلیل و پخش آن‌ها استفاده می‌شود.
- **کاربردهای تحلیل RTP با Wireshark:**
  - **رمزگشایی جریان‌های RTP: Wireshark** می‌تواند جریان‌های RTP را رمزگشایی کند و اطلاعات مفصلی درباره بسته‌ها، مانند نوع بارها، شماره‌های دنباله، مهرهای زمانی و SSRC (شناسه منبع همگام‌سازی) را نمایش دهد.

- **شناسایی مشکلات:** با تجزیه و تحلیل جریان‌های RTP، می‌توان مشکلات رایج مانند از دست رفتن بسته، جیتر (تغییرات زیاد در زمان رسیدن بسته) و بسته‌های خارج از ترتیب را شناسایی کرد که می‌تواند بر کیفیت تجربه استریم تأثیر بگذارد.
- **بازسازی جریان:** Wireshark دارای ویژگی‌هایی برای بازسازی جریان‌های صوتی و تصویری از داده‌های RTP ضبط شده است که امکان پخش مجدد و تجزیه و تحلیل عمیق‌تر کیفیت جریان رسانه را فراهم می‌کند.
- **آمار و تحلیل مشکلات:** Wireshark ابزارهای آماری برای جریان‌های RTP ارائه می‌دهد، از جمله استفاده از پهنای باند، نرخ از دست رفتن بسته‌ها و تجزیه و تحلیل جیتر که به درک عملکرد و کیفیت سرویس کمک می‌کند.

به شکل خلاصه، RTP انتقال داده‌های رسانه‌ای به صورت بلادرنگ از طریق شبکه‌ها را تسهیل می‌کند که برای برنامه‌هایی که نیاز به جریان‌های صوتی و تصویری همزمان دارند، حیاتی است. Wireshark به تحلیل جریان‌های RTP کمک می‌کند و بینش‌هایی در مورد کیفیت و یکپارچگی انتقال رسانه ارائه می‌دهد که برای عیب‌یابی و بهینه‌سازی سیستم‌های ارتباطی بلادرنگ ضروری است.