

به نام خدا

محمد مهدی آفاجانی

تمرین اول مهندسی اینترنت

دکتر بخشی

پاییز ۹۵

## تمرین اول :

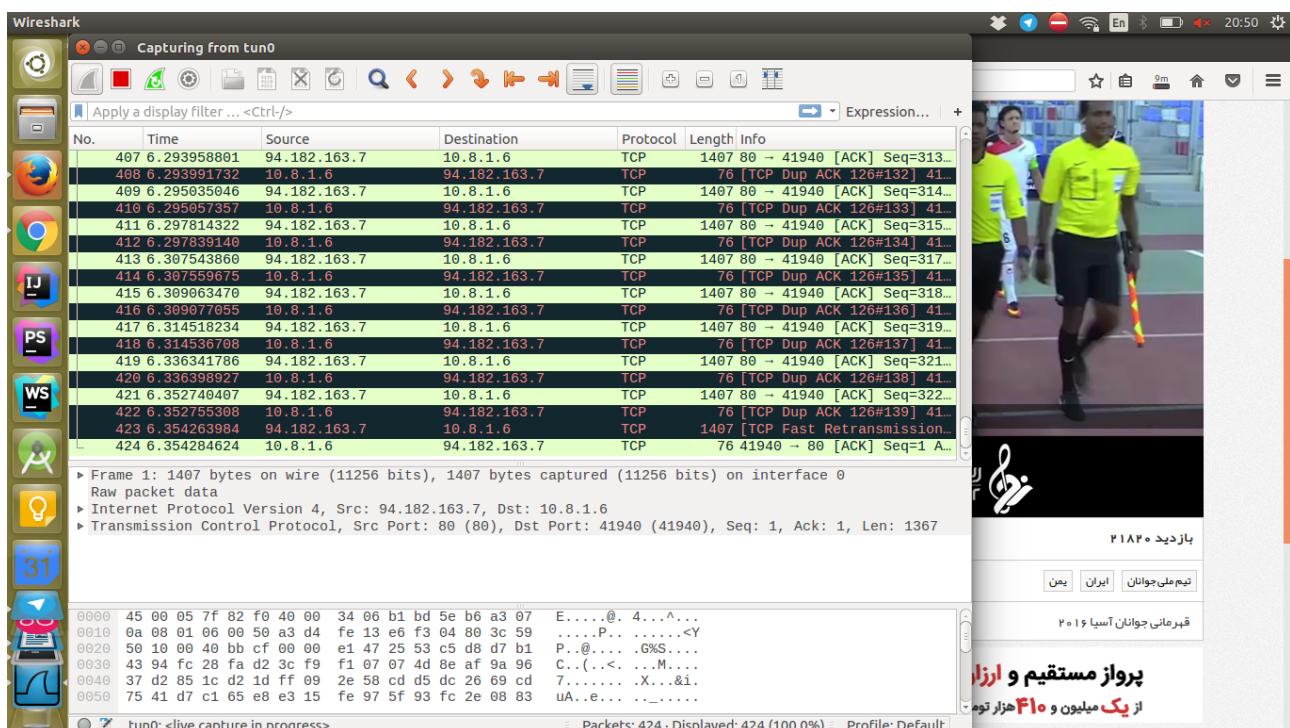
این سرویس را برخی از سایتها ارایه میکنند به این صورت که URL در خواستی را گرفته و آن را با الگوریتمی به یک کلید نگاشت میکنند سپس URL ای تولید میکنند که آن همان دامین سایت خودشان است و بعد از آن کلید تولید شده را به URL میچسبانند سپس هرگاه آن URL درخواست شد به دیتا بیس نگاه میکنند و کلید دریافتی را با لینه ک قبلی مطابقت داده و از طریق redirect HTTP HEADERS اقدام به redirect کردن می نمایند.

## تمرین دوم

فایل انتخاب شده ویدیو خلاصه بازی ایران و یمن از سری رقابت‌های جوانان آسیا بود که از سایت ورزش ۳ به آدرس

[http://video.varzesh3.com/video/۱۲۵۸۴۱/%D۸%AE%D۹%۸۴%D۸%A۷%D۸%B۰%D۹%۸۷-%D۸%A۸%D۸%A۷%D۸%B۲%D۸%C۸-%D۸%A۷%D۸%DB%۸C%D۸%۸C%D۸%B۱%D۸%A۷%D۹%۸۶-۱۰-%DB%۸C%D۹%۸۵%D۹%۸۶-\(%D۸%B۲%D۹%۸۸%D۹%BE%D۸%B۱%D۹%AF%D۹%۸۴-%D۸%B۱%D۸%B۲%D۸%A۷%D۹%۸۲-%D۹%BE%D۹%۸۸%D۸%B۱\)](http://video.varzesh3.com/video/۱۲۵۸۴۱/%D۸%AE%D۹%۸۴%D۸%A۷%D۸%B۰%D۹%۸۷-%D۸%A۸%D۸%A۷%D۸%B۲%D۸%C۸-%D۸%A۷%D۸%DB%۸C%D۸%۸C%D۸%B۱%D۸%A۷%D۹%۸۶-۱۰-%DB%۸C%D۹%۸۵%D۹%۸۶-(%D۸%B۲%D۹%۸۸%D۹%BE%D۸%B۱%D۹%AF%D۹%۸۴-%D۸%B۱%D۸%B۲%D۸%A۷%D۹%۸۲-%D۹%BE%D۹%۸۸%D۸%B۱))

دانلود شد ( ۱۹ مگابایت حجم دارد ) . در ابتدا این فایل را از طریق مرورگر فایرفاکس دانلود نمودم و همزمان بسته های رد و بدل شده را با نرم افزار wireshark در سیستم عامل لینوکس مشاهده کردم که تصاویر آن به صورت زیر است :



\*tun0

tcp

No. Time Source Destination Protocol Length Info

5805 108.326072622 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3308141 Win=8397 Len=0  
5806 108.326047751 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3308141 Ack=1 Win=64 Len=1367  
5807 108.326051038 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3309508 Win=8397 Len=0  
5808 108.326073173 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3309508 Ack=1 Win=64 Len=1367  
5809 108.326078246 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3310875 Win=8397 Len=0  
5810 108.326104402 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3310875 Ack=1 Win=64 Len=1367  
5811 108.326110606 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3312242 Win=8397 Len=0  
5812 108.326137967 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3312242 Ack=1 Win=64 Len=1367  
5813 108.326143896 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3313609 Win=8397 Len=0  
5814 108.326170977 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3313609 Ack=1 Win=64 Len=1367  
5815 108.326177035 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3314976 Win=8397 Len=0  
5816 108.326205750 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3314976 Ack=1 Win=64 Len=1367  
5817 108.326212530 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3316343 Win=8397 Len=0  
5818 108.326240552 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3316343 Ack=1 Win=64 Len=1367  
5819 108.326247576 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3317710 Win=8397 Len=0  
5820 108.326275725 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3317710 Ack=1 Win=64 Len=1367  
5821 108.326282326 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3319077 Win=8397 Len=0  
5822 108.326312817 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3319077 Ack=1 Win=64 Len=1367  
5823 108.326320323 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3320444 Win=8397 Len=0  
5824 108.326350522 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3320444 Ack=1 Win=64 Len=1367  
5825 108.326357434 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3321811 Win=8397 Len=0  
5826 108.326907513 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3321811 Ack=1 Win=64 Len=1367  
5827 108.335240431 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3323178 Ack=1 Win=64 Len=1367  
5828 108.335271575 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3324545 Win=8397 Len=0  
5829 108.366698775 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3324545 Ack=1 Win=64 Len=1367  
5830 108.367717266 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3325912 Ack=1 Win=64 Len=1367  
5831 108.367724777 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3327279 Win=8397 Len=0  
5832 108.969723663 10.8.1.6 209.85.200.136 TLSV1.2 98 Application Data

0000 45 00 05 7f 84 f2 40 00 34 06 af bb 5e b6 a3 07 E.....@. 4....^...  
0010 0a 08 01 06 00 50 a3 d4 fe 1d da 0e 04 80 3c 59 .....P.. ....<Y  
0020 50 10 00 40 02 65 00 00 3a 7c df 7b f9 fd c5 31 P..@.e.. :|.{...1  
0030 b4 c4 ad d2 d9 44 d8 ee 11 de 3d 13 7a 6e fa f0 .....D...=.zn..  
0040 83 dd 27 a5 77 fa 19 93 fd ee 4d db f6 f5 04 bb ..'w....M....  
0050 9f d5 ff 7b 5f 82 64 4b b3 74 21 09 da 07 9d a7 ...{\_.dk t!....

Transmission Control Protocol: Protocol

Packets: 5832 - Displayed: 5772 (99.0%) Profile: Default

\*tun0

tcp

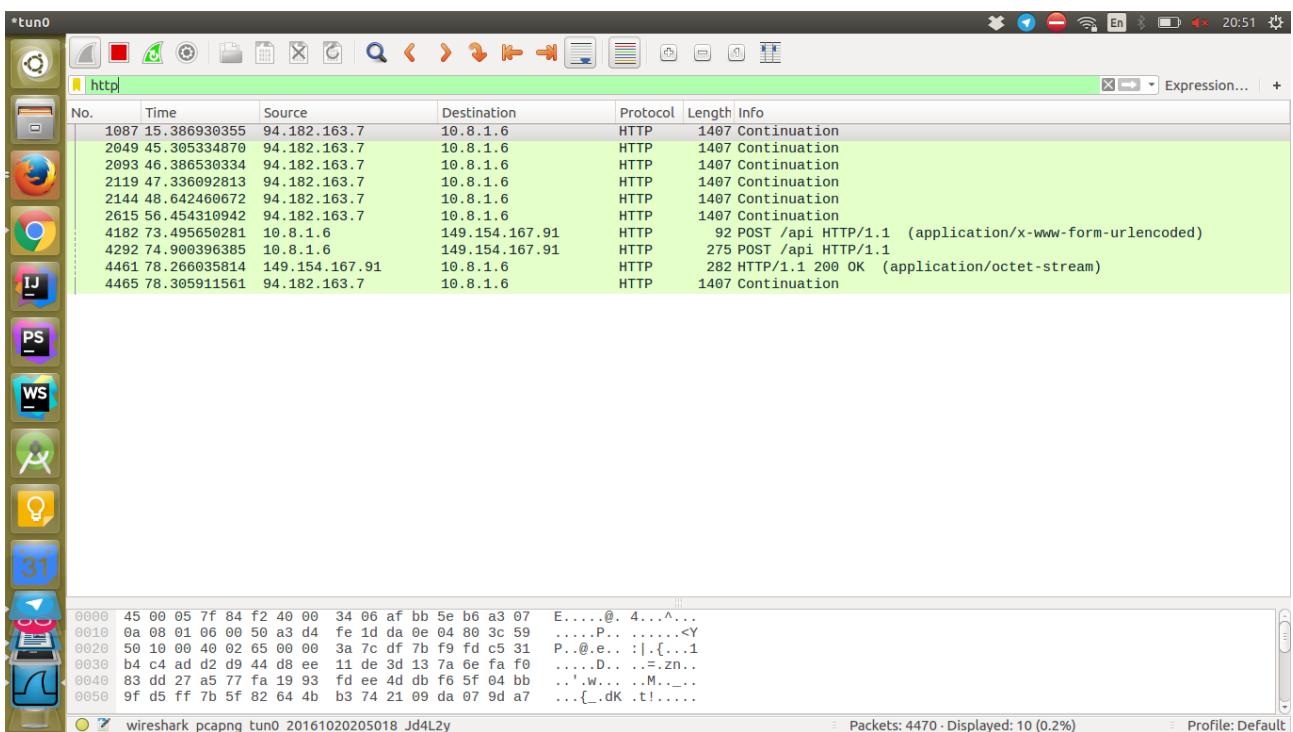
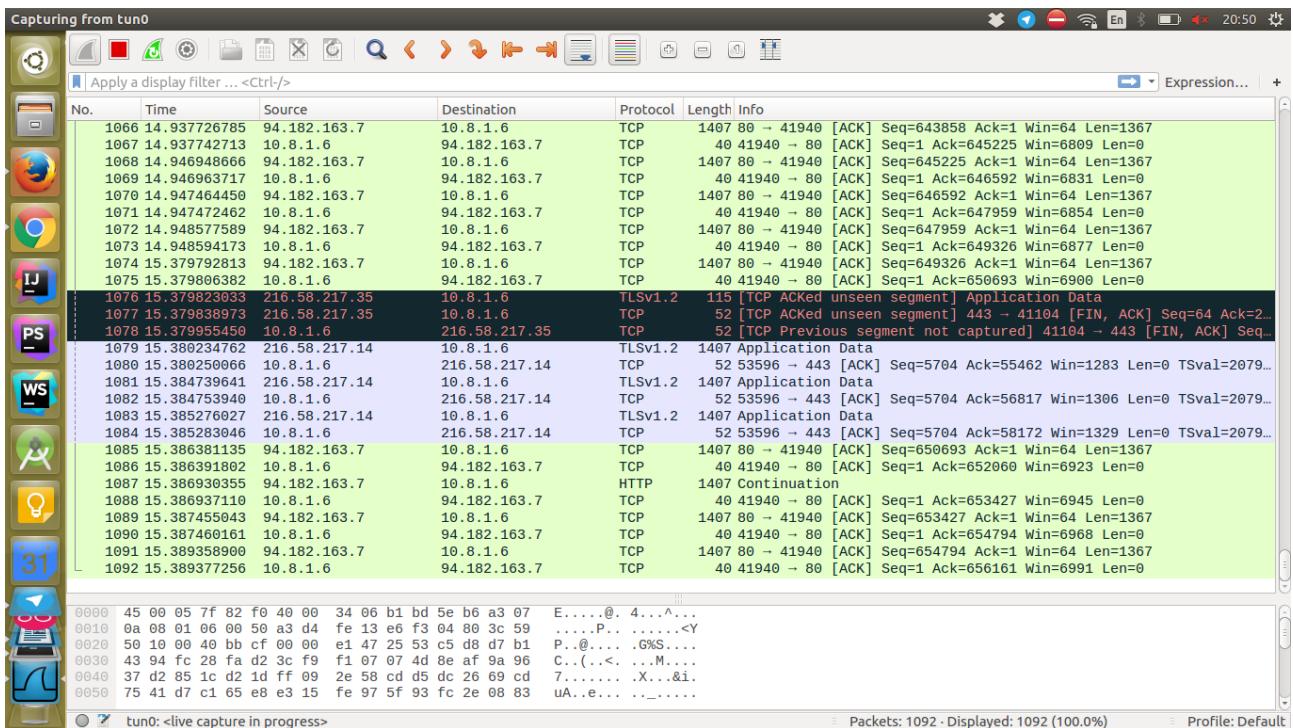
No. Time Source Destination Protocol Length Info

5805 108.326072622 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3308141 Win=8397 Len=0  
5806 108.326047751 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3308141 Ack=1 Win=64 Len=1367  
5807 108.326051038 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3309508 Win=8397 Len=0  
5808 108.326073173 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3309508 Ack=1 Win=64 Len=1367  
5809 108.326078246 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3310875 Win=8397 Len=0  
5810 108.326104402 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3310875 Ack=1 Win=64 Len=1367  
5811 108.326110606 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3312242 Win=8397 Len=0  
5812 108.326137967 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3312242 Ack=1 Win=64 Len=1367  
5813 108.326143896 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3313609 Win=8397 Len=0  
5814 108.326170977 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3313609 Ack=1 Win=64 Len=1367  
5815 108.326177035 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3314976 Win=8397 Len=0  
5816 108.326205750 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3314976 Ack=1 Win=64 Len=1367  
5817 108.326212530 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3316343 Win=8397 Len=0  
5818 108.326240552 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3316343 Ack=1 Win=64 Len=1367  
5819 108.326247576 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3317710 Win=8397 Len=0  
5820 108.326275725 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3317710 Ack=1 Win=64 Len=1367  
5821 108.326282326 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3319077 Win=8397 Len=0  
5822 108.326312817 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3319077 Ack=1 Win=64 Len=1367  
5823 108.326320323 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3320444 Win=8397 Len=0  
5824 108.326350522 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3320444 Ack=1 Win=64 Len=1367  
5825 108.326357434 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3321811 Win=8397 Len=0  
5826 108.326907513 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3321811 Ack=1 Win=64 Len=1367  
5827 108.335240431 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3323178 Ack=1 Win=64 Len=1367  
5828 108.335271575 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3324545 Win=8397 Len=0  
5829 108.366698775 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3324545 Ack=1 Win=64 Len=1367  
5830 108.367717266 94.182.163.7 10.8.1.6 TCP 1407 80 → 41940 [ACK] Seq=3325912 Ack=1 Win=64 Len=1367  
5831 108.367724777 10.8.1.6 94.182.163.7 TCP 40 41940 → 80 [ACK] Seq=1 Ack=3327279 Win=8397 Len=0  
5832 108.969723663 10.8.1.6 209.85.200.136 TLSV1.2 98 Application Data

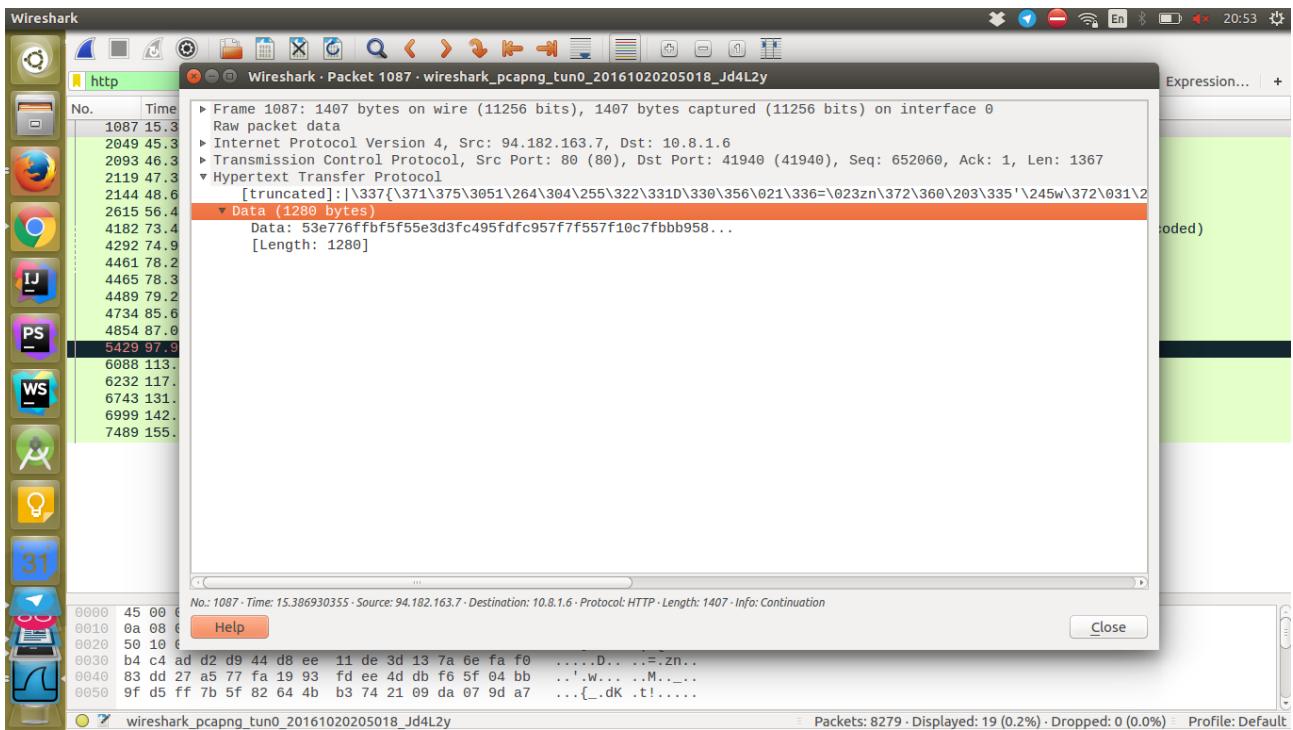
0000 45 00 05 7f 84 f2 40 00 34 06 af bb 5e b6 a3 07 E.....@. 4....^...  
0010 0a 08 01 06 00 50 a3 d4 fe 1d da 0e 04 80 3c 59 .....P.. ....<Y  
0020 50 10 00 40 02 65 00 00 3a 7c df 7b f9 fd c5 31 P..@.e.. :|.{...1  
0030 b4 c4 ad d2 d9 44 d8 ee 11 de 3d 13 7a 6e fa f0 .....D...=.zn..  
0040 83 dd 27 a5 77 fa 19 93 fd ee 4d db f6 f5 04 bb ..'w....M....  
0050 9f d5 ff 7b 5f 82 64 4b b3 74 21 09 da 07 9d a7 ...{\_.dk t!....

Transmission Control Protocol: Protocol

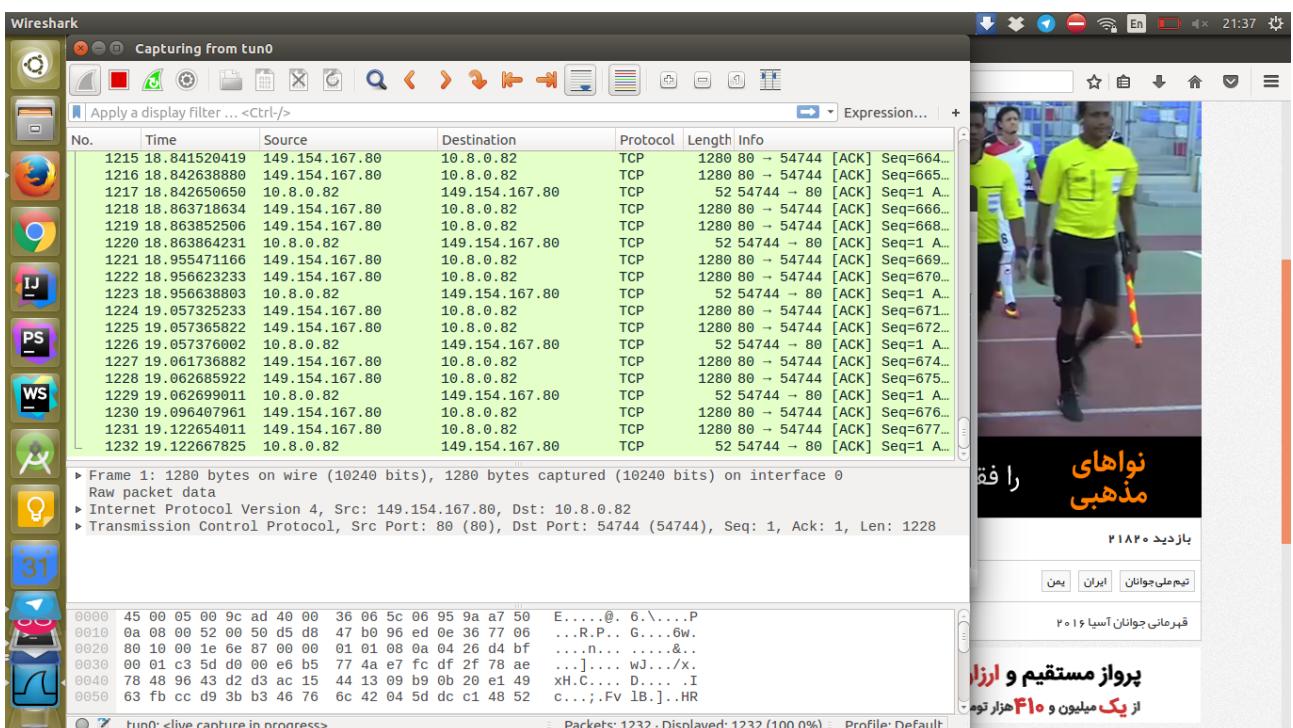
Packets: 5832 - Displayed: 5772 (99.0%) Profile: Default

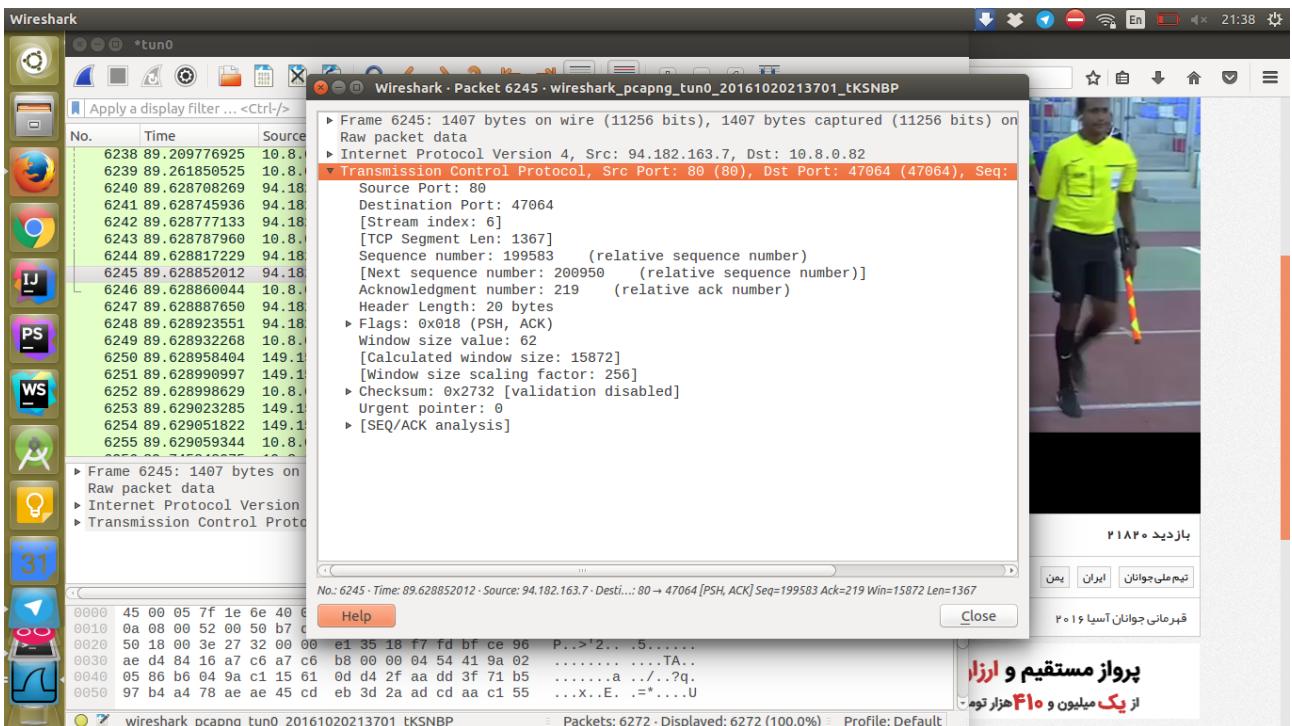


اتفاقی که در این بین میافتد این است که بعد از باز کردن یک TCP connection به صورت دوره ای یک ارسال http request می شود که بدنه آن بخشی از دیتا فایل را دربر دارد که در تصویر بالا قابل مشاهده است همچنین جزئیات این درخواست ها در تصویر پایین قابل مشاهده است .

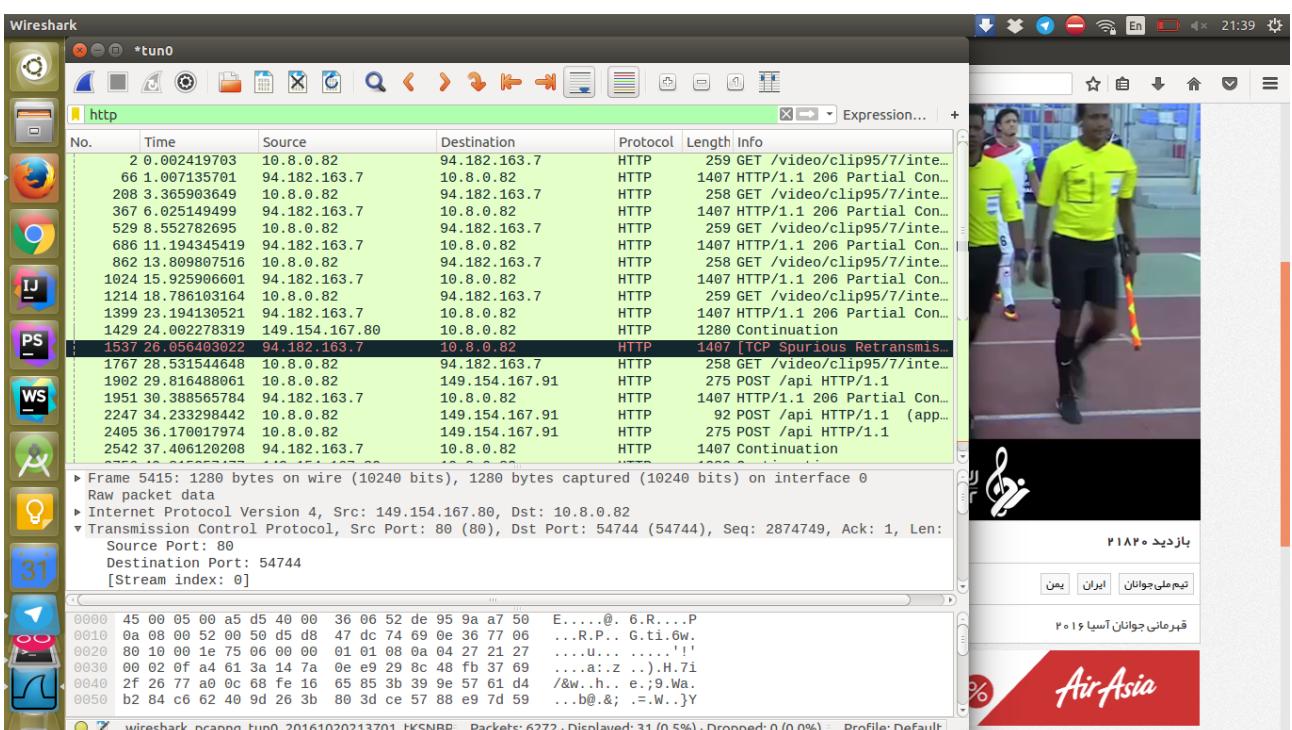


اما وقتی از دانلود منیجر دانلود میکنیم (از xtreme download manager) این فایل را به چند قطعه میشکند و موازی آنها را سعی میکند دانلود بکند. بسته های رد و بدل شده در این حالت به صورت زیر است:





همانطور که میبینید در تصویر بالا جزئیات بسته TCP رد و بدل شده را مشاهده میکنید.



همانطور که در بالا میبیند در ابتدا چندین درخواست HTTP داده شده است که از متده استفاده شده است.

### تمرین سوم :

این سایت ها از Apache server استفاده میکند . این نرم افزار توسط Apache Software Foundation تولید شده است و تحت لایسنس همین شرکت میباشد . آخرین ورژن آن ۲.۴.۲۳ stable میباشد . این نرم افزار همچنین دارای ویژگی های زیر میباشد :

Basic access authentication	-
Digest access authentication	-
SSL/TLS https	-
Virtual hosting	-
CGI , FCGI , SCGI	-
Run in user space	-
Administration console	-
IPv6	-
http ۲	-
windows , OS X , linux . solaris پشتیبانی از	-

همچنین ویژگی های زیر را ندارد :

java servlet	-
msn.com , Microsoft.com	-۲

این سایت ها از Microsoft IIS استفاده میکنند. توسط مایکروسافت تولید شده و رایگان نمیباشد. دارای ویژگی های زیر است :

Basic access authentication	-
Digest access authentication	-
SSL/TLS https	-
Virtual hosting	-
CGI , FCGI , SCGI	-
Run in user space and kernel space	-
Administration console	-
IPv6	-

http ۲ -

windows پشتیبانی تنها از -

ویژگی های زیر را ندارد :

Java servlet -

-۳

این سایت از Nginx استفاده میکند. توسط شرکت Nginx تولید شده و دارای ویژگی های زیر است :

Basic access authentication -

Digest access authentication -

SSL/TLS https -

Virtual hosting -

FCGI , SCGI -

Run in user space -

Administration console -

IPv۶ -

http ۲ -

windows , linux , OS X , solaris پشتیبانی از -

ویژگی های زیر را ندارد :

Java servlets -

CGI -

soft۹۸.ir -۴

این سایت از LiteSpeed Technologies استفاده میکند . توسط LiteSpeed Technologies تولید شده و رایگان نمیباشد . دارای

ویژگی های زیر میباشد :

Basic access authentication -

Digest access authentication -

SSL/TLS https -

Virtual hosting -

FCGI , SCGI , CGI	-
Run in user space	-
Administration console	-
IPv۶	-
http ۲	-
linux , OS X , solaris	- پشتیبانی از
ویژگی های زیر را ندارد :	
Java servlets	-
عدم پشتیبانی از windows	-
Oracle.com , java.com	-۵
این سایت ها از oracle http server استفاده میکنند . توسعه شرکت oracle تولید شده و رایگان نمیباشد . دارای	
ویژگی های زیر است :	
Basic access authentication	-
Digest access authentication	-
SSL/TLS https	-
Virtual hosting	-
FCGI , SCGI , CGI	-
Run in user space	-
Administration console	-
IPv۶	-
windows , linux , solaris	- پشتیبانی از
ویژگی های زیر را ندارد :	
Java servlets	-
عدم پشتیبانی از OS X	-
حال با استفاده از نرم افزار postman چندین درخواست را به این سایت ها میزنیم و جواب آنها را بررسی میکنیم :	
ابتدا به سایت apache.com که از apple.com استفاده میکند درخواست میدهیم :	

The screenshot shows the Postman application interface. On the left is a sidebar with a tree view of recent requests and collections. The main area shows a request to `http://apple.com` via GET. The Headers tab is selected, showing one entry: `Content-Type: application/json`. Other tabs include Body, Cookies, and Tests. The status bar at the bottom right indicates `Status: 200 OK Time: 1141 ms`.

همانطور که میبینید هدر سرور برابر apache قرار گرفته است

حال به سایت [microsoft.com](http://microsoft.com) در خواست میدهیم

The screenshot shows the Postman application interface. The sidebar lists requests to `http://microsoft.com` and `http://apple.com`. The main area shows a request to `http://microsoft.com` via GET. The Headers tab is selected, showing 21 entries. These include standard HTTP headers like `access-control-allow-credentials`, `access-control-allow-methods`, and `cache-control`, as well as specific Microsoft headers like `p3p` and `server`. The status bar at the bottom right indicates `Status: 200 OK Time: 1955 ms`.

The screenshot shows the Postman application interface. On the left, there's a sidebar with icons for different tools like terminal, browser, and file manager. The main area has tabs for 'Runner' and 'Import'. The 'Builder' tab is active, showing a request to 'http://microsoft.com' using a 'GET' method. The 'Headers' tab is selected, displaying 21 header entries. The response status is '200 OK' with a time of '1955 ms'. The response body is partially visible, showing standard Microsoft headers.

همانطور که میبینید هدر های بسیار بیشتری نسبت به apache رد و بدل میشود.

سپس به [wordpress.com](https://wordpress.com) درخواست میزنیم:

This screenshot shows a similar setup in Postman, targeting 'http://wordpress.com'. The 'Headers' tab is active, showing one entry for 'Content-Type' set to 'application/json'. The response status is '200 OK' with a time of '4181 ms'. The response body is partially visible, showing headers typical for an nginx server.

همانطور که میبینید مقدار هدر سرور برابر x است.

حال به [soft98.ir](http://soft98.ir) درخواست میزنیم :

The screenshot shows the Postman application interface. On the left is a sidebar with various icons for different tools and collections. The main area has a 'History' tab selected, showing a list of recent requests. A specific request to `http://soft98.ir` is highlighted. The request details show a `GET` method, URL, and headers. The 'Headers' tab is active, displaying 12 headers. The response status is `200 OK` and time is `6068 ms`.

علاوه بر اینکه مقدار سرور برابر `litespeed` است زبان استفاده شده نیز که برابر `php` است آورده شده.

و در آخر هم به `oracle.com` درخواست زدیم که به طور زیر شد :

This screenshot of Postman shows a similar setup to the previous one, with the 'History' tab selected in the sidebar. A request to `http://oracle.com` is selected. The 'Headers' tab is active, showing 14 headers. The response status is `200 OK` and time is `6311 ms`.

تمرین چهارم :

با مرورگر `google chrome` به سایت `digikala.com` متصل شدم و سپس لگین کردم :

کوکی های سنت شده عبارت بودند از .DKAUTH , DK-Client , NID ... که در شکل زیر مشخص است .

همچنین کوکی DK-Client در پاسخ برگردانده شد. با کلیک بر روی یک محصول و باز کردن لینک آن مرورگر

کوکی را ارسال میکند

Name	Value	Domain	Path	Expires...	Size	HTTP	Secure	SameSite
.DKAUTH	3314BCC80281189D89BA3FCA2EC... CWUU,3a9308d9-ce75-419b-bcc2... N/A N/A N/A 1020							
DK-Client	CWUU,3a9308d9-ce75-419b-bcc2... N/A N/A N/A 53							
_asc	2c24ccb157dcc817e75b0466be N/A N/A N/A 35							
_auc	fc51c56215797ff5b1ff7zeab N/A N/A N/A 33							
_ga	GA1.2.1668600967.1475849748 N/A N/A N/A 33							
_gat	1 N/A N/A N/A 8							
dk-guid	true N/A N/A N/A 14							
scarab.profile	%22165865%7C1475849937%22 N/A N/A N/A 42							
scarab.visitor	%227901417B5DDA6D9%22 N/A N/A N/A 38							
DK-Client	CWUU,3a9308d9-ce75-419b-bcc2... .digikala.com / 2017... 130 ✓							

## تمرین پنجم :

الف ) در ابتدا باید فایل htpasswd را بازسازیم که رد آن اطلاعات یوزرنیم و پسورد کاربر ذخیره می شود برای اینکار

کافیست از دستور زیر استفاده کنیم :

```
sudo htpasswd -c /etc/apache2/.htpasswd test
```

بعد پسورد مربوطه را وارد میکنیم . سپس با استفاده از فایل htaccess برای دایرکتوری مورد نظر میتوان سطح دسترسی در

نظر گرفت به اینصورت که در فolder Basic که از قبل ساخته ایم یک فایل به نام htaccess درست میکنیم و در آن

محتویات زیر را قرار میدهیم :

```
AuthType Basic
```

```
"AuthName "Restricted Content"
```

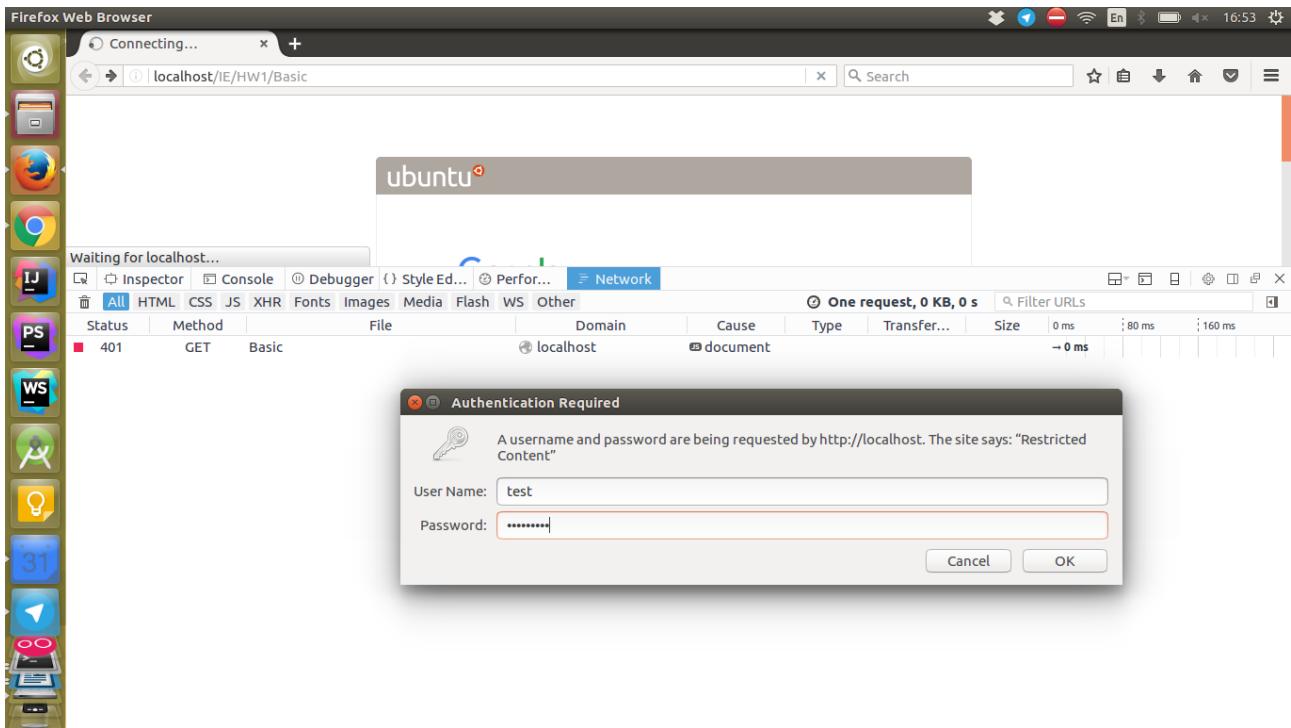
AuthUserFile /etc/apache2/.htpasswd

Require valid-user

این کار برای basic authentication میباشد. سپس باید سرور را ری استارت کنیم که از دستور زیر استفاده میکنیم :

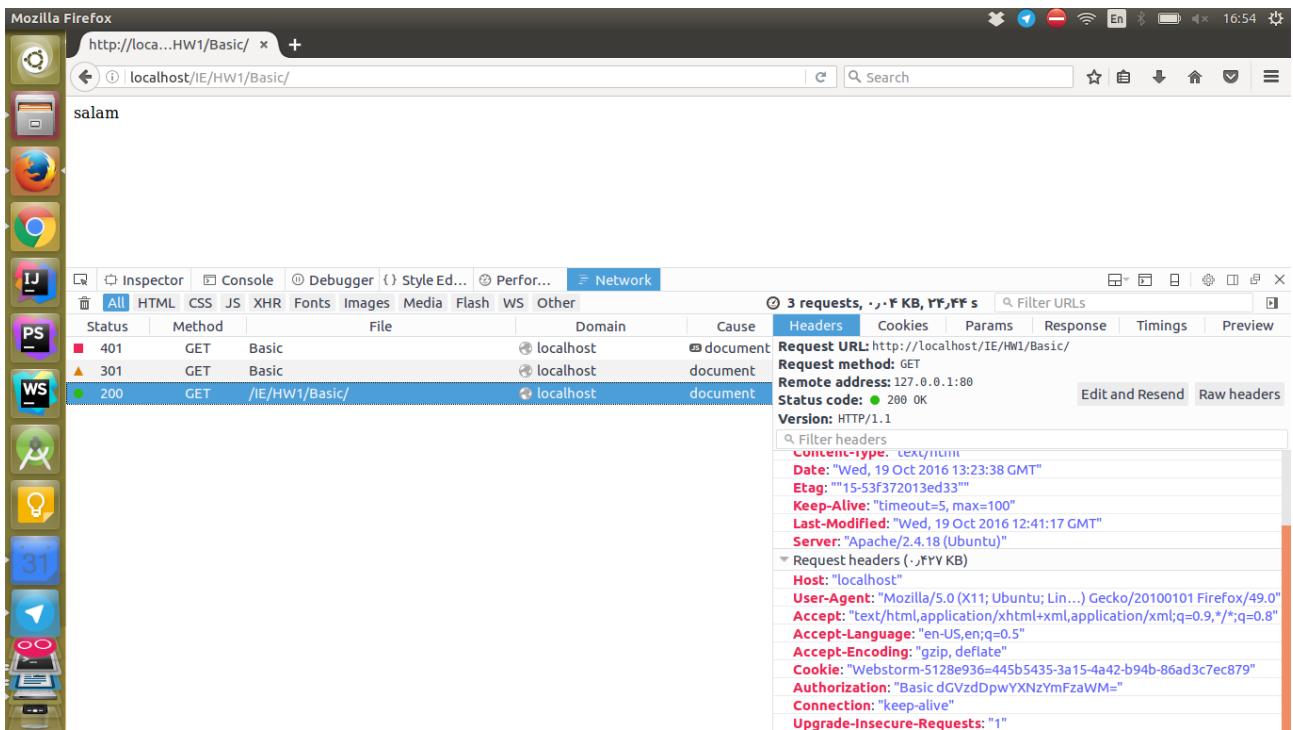
sudo service apache2 restart

بعد با باز کردن مرورگر فایر فاکس و وارد کردن لینک <http://localhost/IE/HW1/Basic> با پنجره زیر روبرو میشویم :



سپس با وارد کردن یوزرنیم و پسورد پنجره زیر به نمایش در میآید :

همانطور که میبینید هدر authentication با عبارت basic شروع شده که به معنای استفاده از حالت basic میباشد.



برای اینکه بتوان از الگوریتم digest استفاده کرد باید مراحل زیر را طی نمود :

ابتدا باید یک فایل پسورد digest بسازیم که اینکار با دستور زیر قابل انجام است :

```
htdigest -c .passwd password_requiered test
```

سپس باید فایل htaccess را درون دایرکتوری مورد نظر ایجاد کنیم و محتویات آن را برابر مقادیر زیر قرار دهیم :

```
AuthType Digest
```

```
"AuthName "test
```

```
AuthUserFile /etc/apache2/.passwd
```

```
AuthDigestDomain /var/www/html/IE/HW1/Digest http://127.0.0.1/IE/HW1/Digest
```

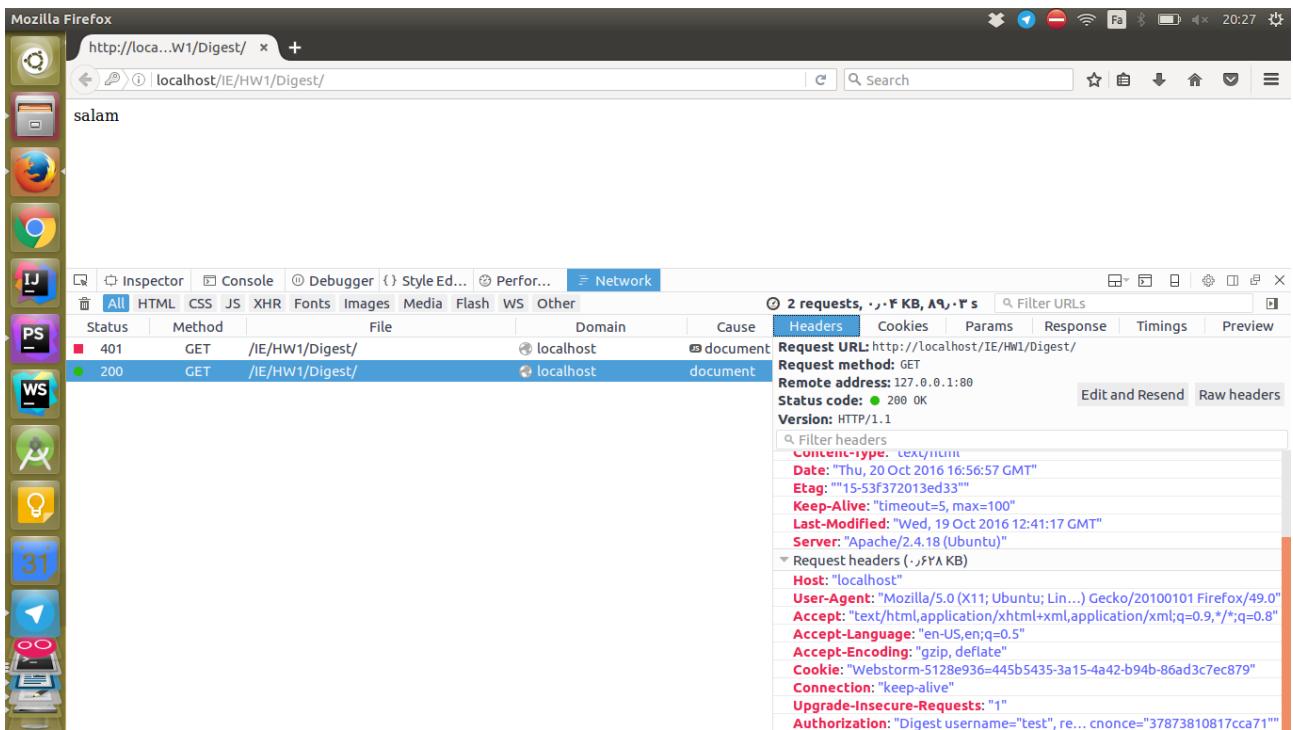
```
Require valid-user
```

برای پسورد خواندن این الگوریتم باید آدرس همان فایل پسورد ایجاد شده را بدهیم.

حال وقتی در مرورگر فایرفاکس درخواست دسترسی به localhost/IE/HW1/Digest را میدهیم با پنجره زیر روبرو خواهیم شد :

می‌شویم :

حال با وارد کردن پسورد مورد نظر پنجره زیر به نمایش در می‌آید :



همانطور که میبینید هدر authentication با مقدار digest شروع شده است که به معنی این است که از این الگوریتم برای

استفاده گردیده است :

ب ) این قسمت در بخش الف انجام شد و تصاویرش نیز گویا می باشد.