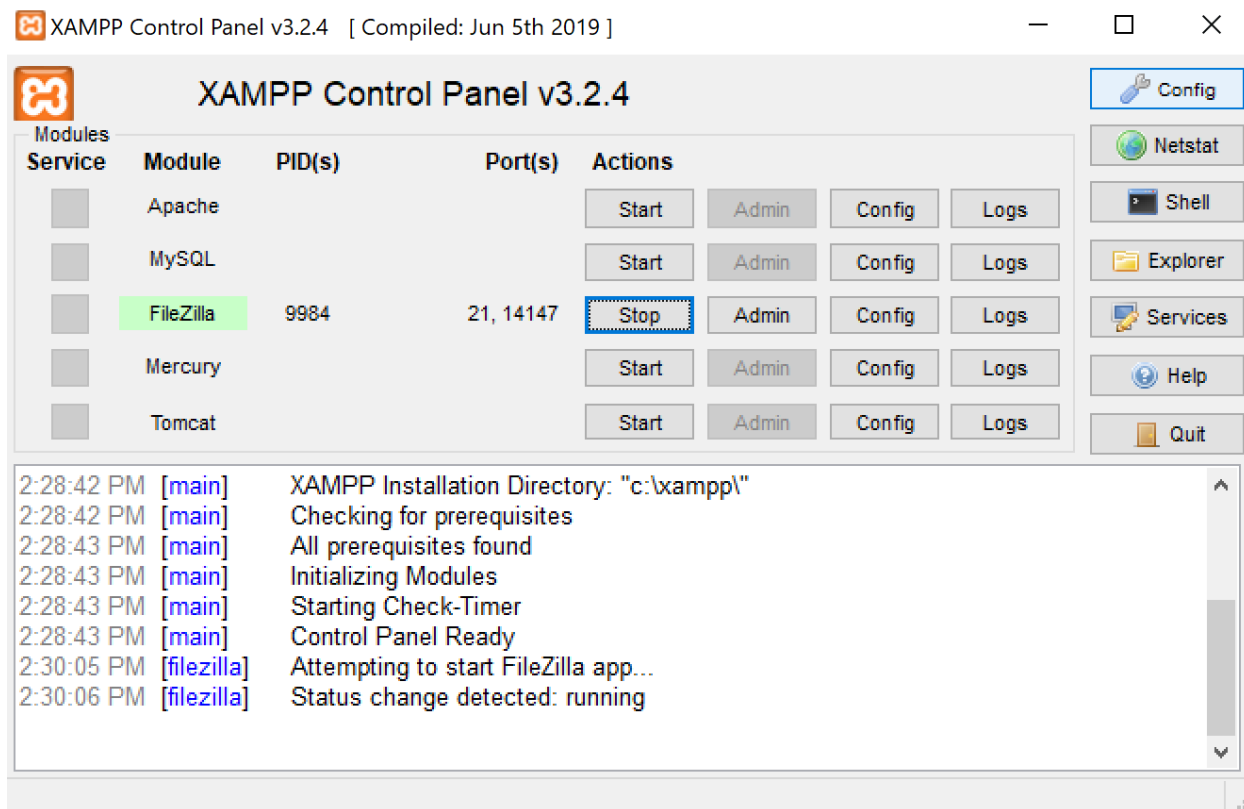
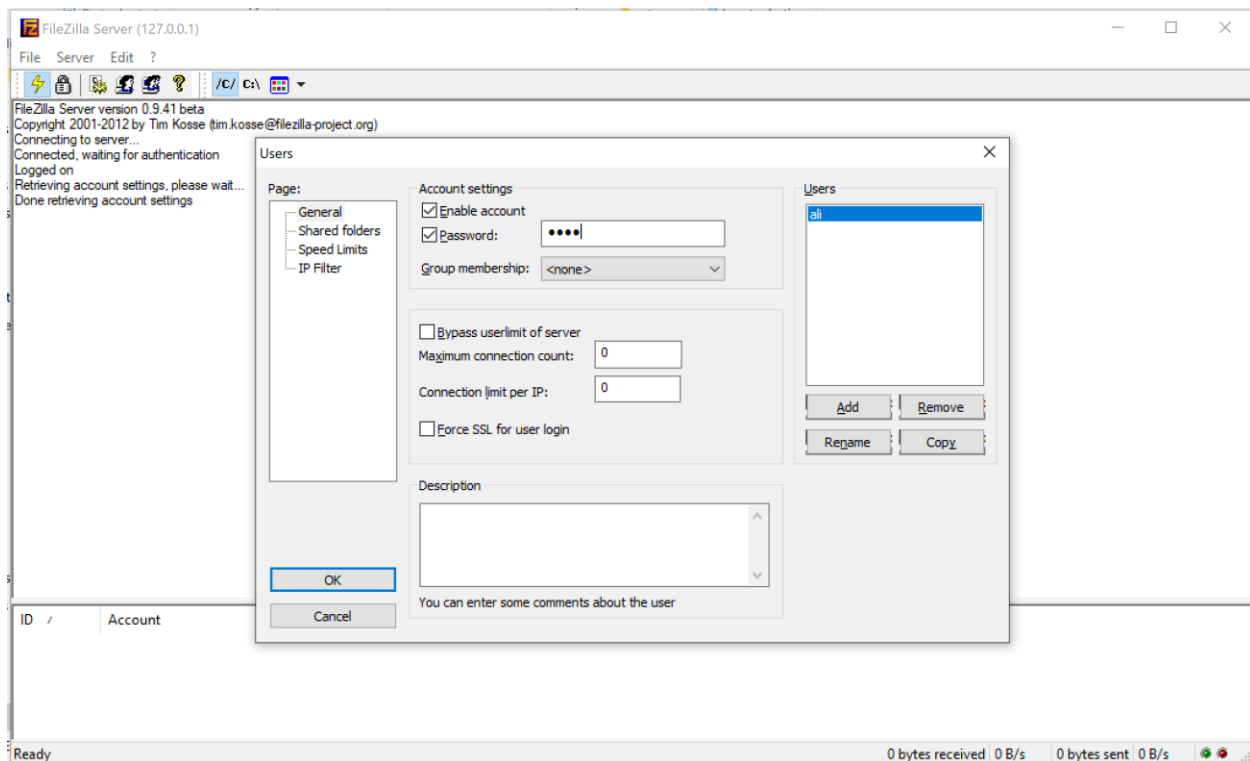


تمرین عملی سری دوم

برای نصب و راه اندازی سرویس ftp با استفاده از XAMPP سرویس FileZilla را start می کنیم:

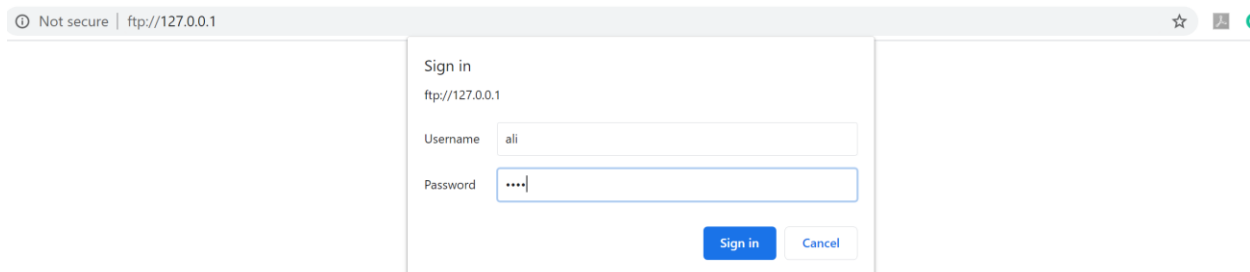


سپس در محل xampp در پوشه FileZillaFTP فایل FileZilla Server Interface.exe را روی آدرس 127.0.0.1 باز می کنیم و از منوی edit و بخش users به صورت زیر یک کاربر با نام ali و پسورد 1111 اضافه می کنیم (دایرکتوری که بعد از لاگین شدن باید نمایش داده شود را نیز مشخص می کنیم) مثل :((Desktop



به این صورت یک سرور ftp راه اندازی شده و یک کاربر هم به آن اضافه شده است.









حال برای لاگین کردن در مرورگر آدرس <ftp://127.0.0.1:21> را وارد می کنیم که اطلاعات کاربر را می خواهد :



بعد از وارد کردن نام کاربری و پسورد دایرکتوری مشخص شده را نمایش می دهد:

← → ↻ ⓘ Not secure | ftp://127.0.0.1

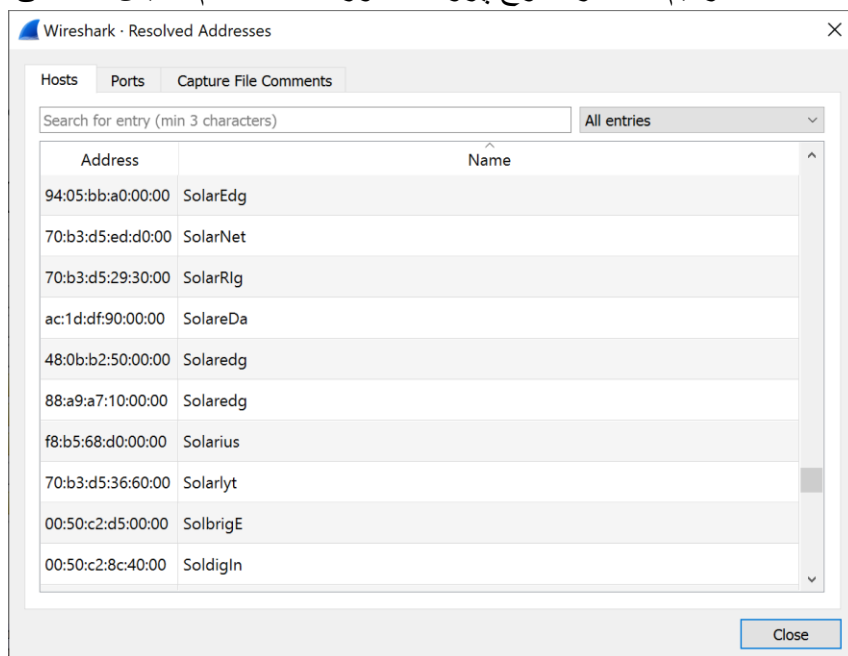
Index of /

	Name	Size	Date Modified
	1.JPG	464 kB	7/14/18, 4:30:00 AM
	2.txt	8.4 kB	3/2/19, 3:30:00 AM
	9631013_Alireza_Pirhadi_3.zip	858 kB	6/9/18, 4:30:00 AM
	AnalogInput/		11/27/19, 3:30:00 AM
	AP_Lab2/		2/22/20, 3:30:00 AM
	Blink/		11/27/19, 3:30:00 AM
	bush.png	46.9 kB	7/11/18, 4:30:00 AM
	Button/		10/22/19, 3:30:00 AM

حال اگر قبل از لاگین کردن با wireshark شنود را شروع کرده باشیم(روی واسط شبکه loopback) می بینیم که یکی از بسته های با پروتکل ftp نام کاربری و دیگری password را نشان می دهد:

1034	116.125042	127.0.0.1	127.0.0.1	FTP	94 Request: USER ali
1065	122.112233	127.0.0.1	127.0.0.1	FTP	115 Response: 331 Password required for ali
1074	122.114257	127.0.0.1	127.0.0.1	FTP	95 Request: PASS 1111
1092	128.109795	127.0.0.1	127.0.0.1	FTP	99 Response: 230 Logged on

۱. در بخش resolved addresses لیستی از تمام آدرس های IP و اسم DNS هایی که در وب گردی از آن ها استفاده کرده ایم را می بینیم هم چنین در بخش دیگری از آن شماره پورت های آدرس هایی که استفاده کردیم به همراه نوع پروتکل مورد استفاده هم نمایش داده می شود.



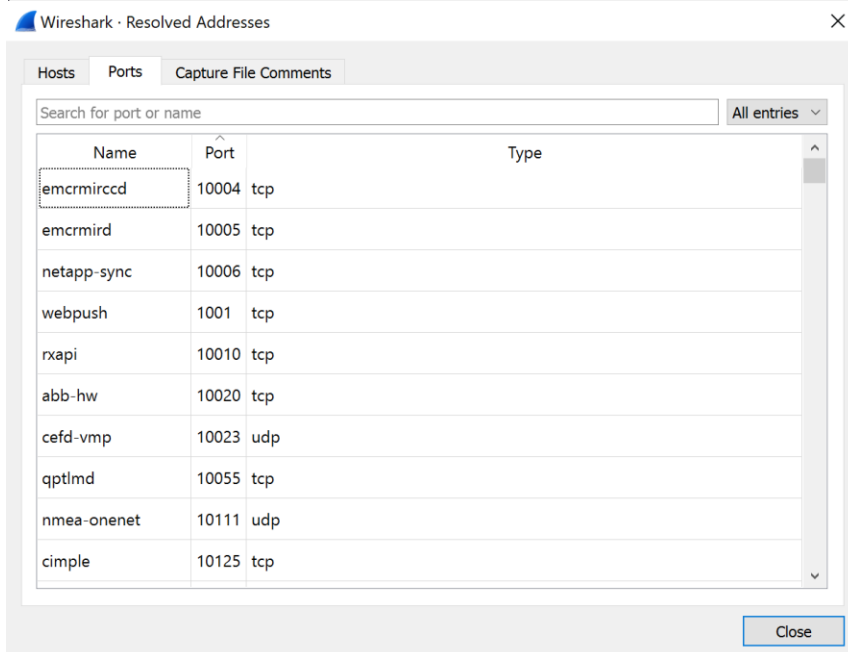
Wireshark - Resolved Addresses

Hosts Ports Capture File Comments

Search for entry (min 3 characters) All entries

Address	Name
94:05:bb:a0:00:00	SolarEdg
70:b3:d5:ed:d0:00	SolarNet
70:b3:d5:29:30:00	SolarRIg
ac:1d:df:90:00:00	SolareDa
48:0b:b2:50:00:00	Solaredg
88:a9:a7:10:00:00	Solaredg
f8:b5:68:d0:00:00	Solarus
70:b3:d5:36:60:00	Solarlyt
00:50:c2:d5:00:00	SolbrigE
00:50:c2:8c:40:00	SoldigIn

Close



Wireshark - Resolved Addresses

Hosts Ports Capture File Comments

Search for port or name All entries

Name	Port	Type
emcirmirccd	10004	tcp
emcirmird	10005	tcp
netapp-sync	10006	tcp
webpush	1001	tcp
rxapi	10010	tcp
abb-hw	10020	tcp
cefd-vmp	10023	udp
qptlmd	10055	tcp
nmea-onenet	10111	udp
cimple	10125	tcp

Close

۲. در بخش protocol hierarchy می‌توانیم ببینیم چند درصد بسته‌ها یا بایت‌های منتقل شده از طریق یک پروتکل منتقل شده است. مثلاً مشاهده می‌کنیم که ۱۰۰ درصد بسته‌ها روی بستر IPv4 منتقل شده‌اند و ۹۳/۶ درصد بسته‌ها از طریق TCP و روی بستر IPv4 منتقل شده‌اند.

Wireshark - Protocol Hierarchy Statistics - Wi-Fi

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
▼ Frame	100.0	13570	100.0	8373990	1269 k	0	0	0
▼ Ethernet	100.0	13570	2.3	189980	28 k	0	0	0
▼ Internet Protocol Version 4	100.0	13568	3.2	271376	41 k	0	0	0
▼ User Datagram Protocol	6.4	862	0.1	6896	1045	0	0	0
Simple Service Discovery Protocol	1.1	153	0.6	46965	7118	153	46965	7118
Domain Name System	1.0	136	0.1	8059	1221	136	8059	1221
Data	4.2	573	3.5	291814	44 k	573	291814	44 k
▼ Transmission Control Protocol	93.6	12702	90.3	7558812	1145 k	9892	5187403	786 k
Transport Layer Security	20.5	2785	63.5	5316696	805 k	2710	4749688	719 k
Malformed Packet	0.2	30	0.0	0	0	30	0	0
▼ Hypertext Transfer Protocol	0.2	30	0.3	20958	3176	10	2504	379
Online Certificate Status Protocol	0.1	20	0.1	10428	1580	20	10428	1580
Data	0.3	40	0.6	48445	7342	40	48445	7342
Internet Group Management Protocol	0.0	4	0.0	32	4	4	32	4
Address Resolution Protocol	0.0	2	0.0	56	8	2	56	8

No display filter.

Close Copy Help

۳. در بخش conversations می توانیم جزئیات دقیق تمام ارتباطات به تفکیک بستر ها و پروتکل ها را ببینیم. مثلا با پروتکل TCP اگر ارتباطی که بین دو آدرس A و B به وجود آمده می توانیم بفهمیم روی چه port هایی بوده، تعداد بسته ها و بایت ها و بیت های منتقل شده در هر سمت و در مجموع و زمان شروع و مدت ارتباط چه بوده است.(یا اطلاعات مشابه برای همه ارتباطات روی IPv4) مانند شکل زیر:

Wireshark · Conversations · Wi-Fi

Ethernet · 8		IPv4 · 52		IPv6		TCP · 198		UDP · 117					
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
54.229.134.170	8282	192.168.1.6	1648	2	121	1	66	1	55	27.222861	0.1243	4246	3538
74.125.140.188	5228	192.168.1.6	1697	4	242	2	132	2	110	5.189582	45.3014	23	19
192.168.1.6	34708	93.184.216.34	80	7	402	5	282	2	120	0.319728	0.0617	36 k	15 k
192.168.1.6	34709	93.184.216.34	80	7	402	5	282	2	120	0.617240	0.0582	38 k	16 k
192.168.1.6	34710	93.184.216.34	80	7	402	5	282	2	120	0.617564	0.0585	38 k	16 k
192.168.1.6	13396	185.4.104.224	443	133	10 k	66	3732	67	6365	0.780540	51.9991	574	979
192.168.1.6	34711	93.184.216.34	80	7	402	5	282	2	120	1.136292	0.0691	32 k	13 k
192.168.1.6	34712	93.184.216.34	80	7	402	5	282	2	120	2.038723	0.0587	38 k	16 k
192.168.1.6	34713	93.184.216.34	80	7	402	5	282	2	120	3.429135	0.0647	34 k	14 k
192.168.1.6	13064	86.104.45.141	443	26	1908	13	738	13	1170	4.286163	45.1991	130	207
192.168.1.6	34184	52.222.157.212	443	6	1209	3	644	3	565	4.838508	8.1307	633	555
192.168.1.6	25109	162.159.130.234	443	8	599	4	318	4	281	5.296822	41.4087	61	54
192.168.1.6	34714	93.184.216.34	80	7	402	5	282	2	120	5.321035	0.0595	37 k	16 k
192.168.1.6	34715	93.184.216.34	80	8	456	5	282	3	174	5.621198	0.0697	32 k	19 k
192.168.1.6	34716	93.184.216.34	80	7	402	5	282	2	120	5.621903	0.0687	32 k	13 k
192.168.1.6	34717	93.184.216.34	80	7	402	5	282	2	120	6.136414	0.0606	37 k	15 k
192.168.1.6	34718	93.184.216.34	80	7	402	5	282	2	120	7.041179	0.0767	29 k	12 k
192.168.1.6	34469	185.143.233.5	443	8	476	5	271	3	205	7.078436	27.8644	77	58
192.168.1.6	34719	94.182.183.34	443	1,179	891 k	576	51 k	603	840 k	7.881370	13.2077	31 k	508 k
192.168.1.6	34720	94.182.183.34	443	328	223 k	172	24 k	156	198 k	7.881695	5.3771	35 k	296 k
192.168.1.6	34721	94.182.183.34	443	14	4489	8	641	6	3848	7.912955	0.1210	42 k	254 k
192.168.1.6	34722	94.182.183.34	443	14	4501	8	653	6	3848	7.914341	0.1334	39 k	230 k
192.168.1.6	34723	23.58.223.136	80	7	1580	4	482	3	1098	8.110744	0.0695	55 k	126 k
192.168.1.6	34724	23.58.223.136	80	7	1580	4	482	3	1098	8.110744	0.0662	58 k	132 k
192.168.1.6	34725	93.184.216.34	80	8	456	5	282	3	174	8.428915	0.2733	8253	5092
192.168.1.6	34726	94.182.183.34	443	124	74 k	67	13 k	57	60 k	8.563918	4.6948	23 k	103 k
192.168.1.6	34727	94.182.183.34	443	193	127 k	103	17 k	90	109 k	8.564452	4.6943	30 k	186 k
192.168.1.6	34728	94.182.183.34	443	592	430 k	298	34 k	294	395 k	8.565999	12.1895	22 k	259 k
192.168.1.6	34729	94.182.183.34	443	244	162 k	129	20 k	115	142 k	8.566880	4.6918	34 k	242 k
192.168.1.6	34730	185.147.178.24	443	63	28 k	34	4346	29	24 k	8.695069	26.2471	1324	7435

☐ Name resolution☐ Limit to display filter☐ Absolute start time

Conversation Types ▾

CopyFollow Stream...Graph...CloseHelp

۴. برای مشخص کردن یک نشست از همان بخش conversations می توانیم استفاده کنیم مثلا در شکل بالا می بینیم که یک نشست بین مبدا با آدرس 192.168.1.6 و پورت 34708 با مقصد با آدرس 93.184.216.34 و پورت 80 ایجاد شده است.

192.168.1.6 34708 93.184.216.34 80 7 402 5 282 2 120 0.319728 0.0617 36 k 15 k

۵. در بخش endpoints تعداد و اطلاعات endpoint ها(مبدا ها یا مقصد ها) به تفکیک پروتکل ها و بستر ها آمده است. مثلا در شکل زیر می بینیم که ۵۲ تا IPv4 endpoint در این مدت capture شده است و اطلاعات هر کدام مثل تعداد بسته ها و بایت های دریافت شده(Received packets(Rx)) و ارسال شده(transmitted packets(Tx)) مشخص شده است.

Wireshark · Endpoints · Wi-Fi

Ethernet · 7		IPv4 · 52		IPv6		TCP · 234		UDP · 99					
Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes	Country	City	AS Number	AS Organization			
13.224.226.63	85	32 k	37	26 k	48	6064	—	—	—	—			
23.58.222.25	21	1263	12	777	9	486	—	—	—	—			
23.58.223.136	76	23 k	34	18 k	42	4526	—	—	—	—			
45.94.255.23	65	21 k	29	16 k	36	5037	—	—	—	—			
45.156.184.4	63	32 k	29	29 k	34	3048	—	—	—	—			
46.224.1.220	75	7694	37	4667	38	3027	—	—	—	—			
46.224.1.221	61	6077	30	3624	31	2453	—	—	—	—			
52.222.157.212	6	1209	3	565	3	644	—	—	—	—			
54.229.134.170	2	121	1	66	1	55	—	—	—	—			
72.247.161.167	18	1008	7	414	11	594	—	—	—	—			
74.125.140.188	4	242	2	132	2	110	—	—	—	—			
86.104.45.141	26	1908	13	1170	13	738	—	—	—	—			
93.184.216.34	463	26 k	137	8190	326	18 k	—	—	—	—			
94.182.132.65	3,485	2490 k	1,703	2346 k	1,782	143 k	—	—	—	—			
94.182.163.52	166	89 k	76	82 k	90	6960	—	—	—	—			
94.182.163.91	36	14 k	14	11 k	22	2285	—	—	—	—			
94.182.163.181	45	19 k	18	16 k	27	2649	—	—	—	—			
94.182.163.211	77	29 k	30	25 k	47	4149	—	—	—	—			
94.182.163.231	75	26 k	29	22 k	46	4044	—	—	—	—			
94.182.183.34	4,715	3402 k	2,333	3158 k	2,382	244 k	—	—	—	—			
104.31.77.193	440	233 k	215	211 k	225	21 k	—	—	—	—			
104.66.65.152	16	888	6	348	10	540	—	—	—	—			
147.75.33.229	64	20 k	28	16 k	36	3508	—	—	—	—			
147.75.101.5	164	95 k	80	89 k	84	6251	—	—	—	—			
148.72.152.18	14	1068	7	546	7	522	—	—	—	—			
151.139.128.14	14	3114	6	2196	8	918	—	—	—	—			
162.159.130.234	8	599	4	281	4	318	—	—	—	—			
172.64.107.3	165	58 k	78	50 k	87	8219	—	—	—	—			
172.64.138.32	1,769	1200 k	883	1122 k	886	78 k	—	—	—	—			
172.67.174.163	107	34 k	48	24 k	59	10 k	—	—	—	—			

☐ Name resolution ☐ Limit to display filter Endpoint Types ▾

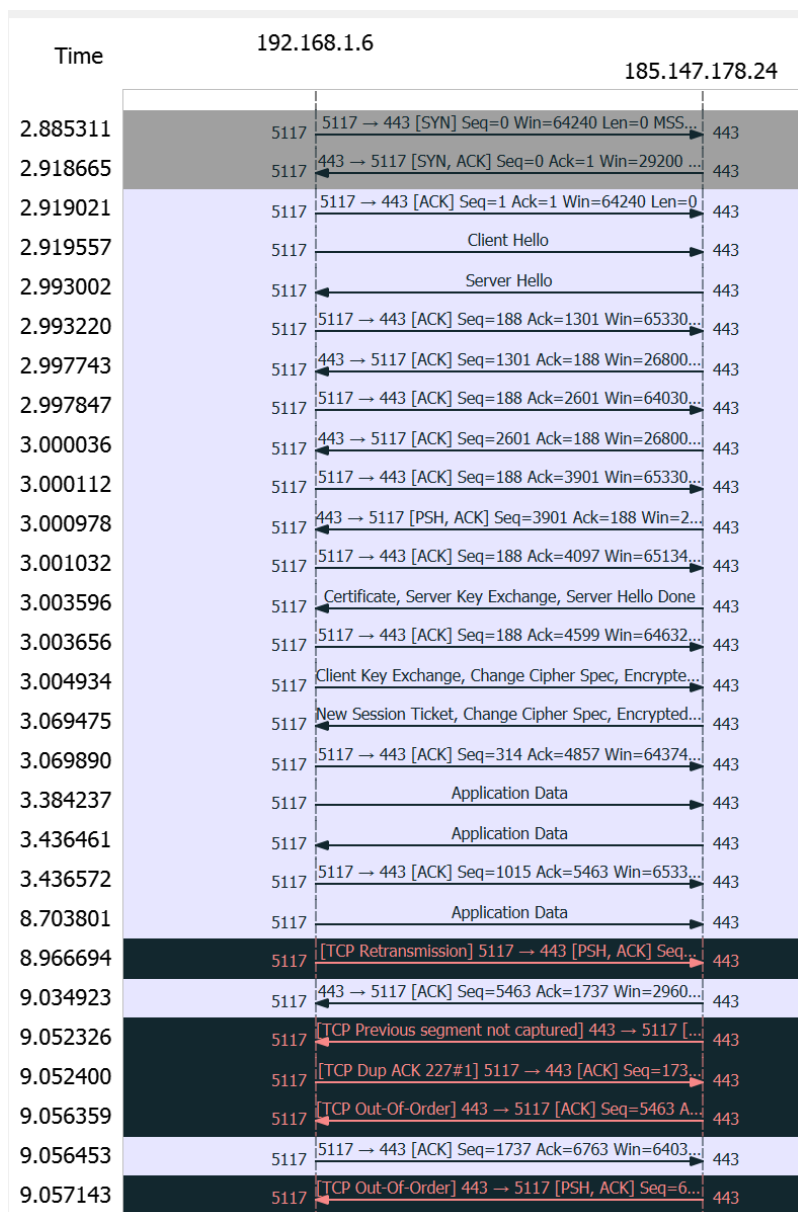
Copy ▾ Map ▾ Close Help

با مرتب کردن endpoint ها بر اساس آدرس ها و پورت ها در tab مخصوص TCP می بینیم که روی سیستم ما یعنی با آدرس 192.168.1.6 از پورت 1603 تا پورت 34921 برای ارتباطات TCP استفاده شده است پس مرورگر با چندین فرآیند و به صورت multi processing اتصالات را ایجاد می کند.

برای تشخیص Default gateway می بینیم دو تا آدرس بیشتر از بقیه بسته منتقل کرده اند که یکی از آن ها مودم و دیگری کامپیوتر ما است برای تشخیص مودم (default gateway) اونی که بایت کمتری دریافت کرده (Rx کمتری دارد) را انتخاب می کنیم چرا که بیشتر داده ها توسط کامپیوتر دریافت می شود. پس Default gateway = 28:3b:82:5e:0d:b8

Ethernet · 7		IPv4 · 52		IPv6	TCP · 234		UDP · 99	
Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes		
00:16:eb:45:fa:0b	13,381	8316 k	6,962	691 k	6,419			7625 k
01:00:5e:00:00:fb	1	46	0	0	1			46
01:00:5e:00:00:fc	1	46	0	0	1			46
01:00:5e:7f:ff:fa	154	53 k	0	0	154			53 k
28:3b:82:5e:0d:b8	13,442	8337 k	6,553	7672 k	6,889			664 k
d0:6f:4a:bb:78:0e	120	34 k	55	9948	65			24 k
ff:ff:ff:ff:ff:ff	41	6150	0	0	41			6150

۶. حال یک نشست را از conversation انتخاب می کنیم و با زدن tcp stream follow آن را فیلتر می کنیم (که من نشست کامپیوتر با 185.147.178.24 را انتخاب کردم) سپس در flow graph با زدن limit to display filter ارتباط بین شان را بررسی می کنیم که به صورت زیر است :



برای ایجاد ارتباط از 3 way handshake استفاده شده است که ابتدا کلاینت برای ایجاد ارتباط TCP یک بسته با Seq=0 و با فلگ SYN می فرستد که در آن Len هم صفر است یعنی داده ای ارسال نمی شود و همچنین اندازه پنجره کلاینت win=64240 قرار داده شده. سپس سرور در پاسخ seq=0 و ack=1 را با فلگ های SYN و ack می فرستد و اندازه پنجره را 29200 قرار می دهد و در ادامه کلاینت seq=1 (همانطور که سرور انتظار دارد) و ack=1 را با فلگ ack می فرستد و اندازه پنجره را تغییر نمی دهد. به این صورت ارتباط بین شان ایجاد شده است.

سپس برای handshake اول TLS کلاینت یک hello فرستاده که سرور هم hello را در پاسخ فرستاده است.

در ادامه می بینیم درخواست های کلاینت با موفقیت انجام شده و seq هایی که سرور می فرستد برابر Ack های کلاینت در بسته قبل است. در بسته های ۱۳ و ۱۵ و ۱۶ هم key exchange اتفاق افتاده است. سپس می بینیم که انتقال داده با application data مشخص شده است (به دلیل https بودن ارتباط جزئیاتش مخفی است)

در ادامه packet loss هم رخ داده است که در آن کلاینت در اولین خط سیاه مشخص شده بسته ای با ack=5463 می فرستد و چون داده کمتری دریافت شده، یک بسته (که در سومین خط سیاه مشخص شده) به عنوان tcp dup ACK می فرستد و در جواب باقی بسته را دریافت می کند.