

Otto-Friedrich-University of Bamberg

Professorship for Computer Science,
Communication Services, Telecommunication
Systems and Computer Networks



Foundation of Internet Communication

Assignment 1

Submitted by:

Group X

Moktahid Al Faisal
Abdullah Al Mosabbir
Mohammed Mehedi Hasan
Kazi Sayef Shawgat
Sheikh Jumon Ahmed

Supervisor: Prof. Dr. Udo Krieger

Bamberg, May 13, 2020
Summer Term 2020

Contents

1	Taking Wireshark for a Test Run	1
1.1	1
1.2	1
1.3	2
2	Wireshark Handling	3
3	Wireshark Packet Filtering	5
4	Taking Kathará for a Test Run	9
4.1	Experiment	9
5	Basic Linux network administration	11
5.1	11
5.2	11
5.3	12
5.4	12

List of Figures

1	1.1	1
2	1.2	1
3	1.3	2
4	Response time to get Okay response	3
5	Fau destination address	4
6	Denic destination address	4
7	ping google	5
8	ICMP Display Filter	5
9	Packet sent by host to caida site	5
10	Host MAC address	6
11	Packets filtered by MAC address sent by host to caida site . .	6
12	Packets recieved by host ip address	7
13	Tcp dump to source IP address 192.168.0.159	7
14	Tcp Dump between two hosts 10.0.0.3 and 10.0.0.12	8
15	Pakcet containing Ip on host 10.0.0.3	8
16	Tcp Dump for Port 22	8
17	Filter Command	8
18	Kathará installation verification	9
19	Kathará commands	9
20	Kathará container creation	10
21	Kathará container network interfaces	10
22	Kathará container tear down	10
23	Create emulated pc with kathara command	11
24	Assign network interfaces with emulated pc	11

25	Setting up ip addresses in emulated pc	12
26	Create bridge and configure switch with brctl	12
27	Ping and check if connection works	12
28	CP dump to check ICMP message	13

1 Taking Wireshark for a Test Run

1.1

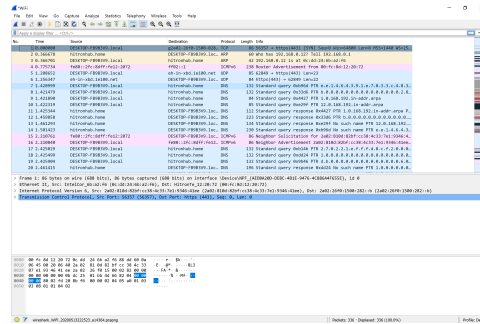


Figure 1: 1.1

1.2

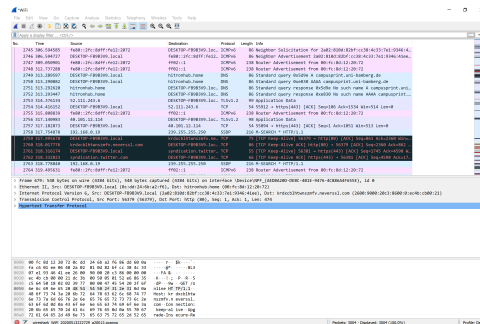


Figure 2: 1.2

1.3

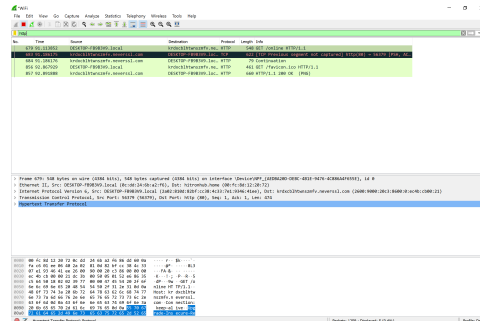


Figure 3: 1.3

2 Wireshark Handling

1. After starting wireshark, I entered in wifi interface and captured packet. While running wireshark I visited neverssl.com and it showed http message exchange with my web server in . Then I added http filter in display filter option.
2. the period between the sending of the HTTP GET message until the receipt of the HTTP OK reply.

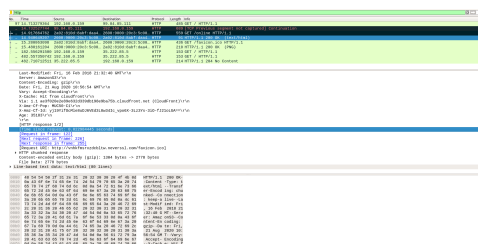


Figure 4: Response time to get Okay response

3. IP addresses of the servers `www.fau.de` and `www.denic.de`

Index	Name	Description	Parent - object ID
1	Root		
2	Root		
3	Root		
4	Root		
5	Root		
6	Root		
7	Root		
8	Root		
9	Root		
10	Root		
11	Root		
12	Root		
13	Root		
14	Root		
15	Root		
16	Root		
17	Root		
18	Root		
19	Root		
20	Root		
21	Root		
22	Root		
23	Root		
24	Root		
25	Root		
26	Root		
27	Root		
28	Root		
29	Root		
30	Root		
31	Root		
32	Root		
33	Root		
34	Root		
35	Root		
36	Root		
37	Root		
38	Root		
39	Root		
40	Root		
41	Root		
42	Root		
43	Root		
44	Root		
45	Root		
46	Root		
47	Root		
48	Root		
49	Root		
50	Root		
51	Root		
52	Root		
53	Root		
54	Root		
55	Root		
56	Root		
57	Root		
58	Root		
59	Root		
60	Root		
61	Root		
62	Root		
63	Root		
64	Root		
65	Root		
66	Root		
67	Root		
68	Root		
69	Root		
70	Root		
71	Root		
72	Root		
73	Root		
74	Root		
75	Root		
76	Root		
77	Root		
78	Root		
79	Root		
80	Root		
81	Root		
82	Root		
83	Root		
84	Root		
85	Root		
86	Root		
87	Root		
88	Root		
89	Root		
90	Root		
91	Root		
92	Root		
93	Root		
94	Root		
95	Root		
96	Root		
97	Root		
98	Root		
99	Root		
100	Root		
101	Root		
102	Root		
103	Root		
104	Root		
105	Root		
106	Root		
107	Root		
108	Root		
109	Root		
110	Root		
111	Root		
112	Root		
113	Root		
114	Root		
115	Root		
116	Root		
117	Root		
118	Root		
119	Root		
120	Root		
121	Root		
122	Root		
123	Root		
124	Root		
125	Root		
126	Root		
127	Root		
128	Root		
129	Root		
130	Root		
131	Root		
132	Root		
133	Root		
134	Root		
135	Root		
136	Root		
137	Root		
138	Root		
139	Root		
140	Root		
141	Root		
142	Root		
143	Root		
144	Root		
145	Root		
146	Root		
147	Root		
148	Root		
149	Root		
150	Root		
151	Root		
152	Root		
153	Root		
154	Root		
155	Root		
156	Root		
157	Root		
158	Root		
159	Root		
160	Root		
161	Root		
162	Root		
163	Root		
164	Root		
165	Root		
166	Root		
167	Root		
168	Root		
169	Root		
170	Root		
171	Root		
172	Root		
173	Root		
174	Root		
175	Root		
176	Root		
177	Root		
178	Root		
179	Root		
180	Root		
181	Root		
182	Root		
183	Root		
184	Root		
185	Root		
186	Root		
187	Root		
188	Root		
189	Root		
190	Root		
191	Root		
192	Root		
193	Root		
194	Root		
195	Root		
196	Root		
197	Root		
198	Root		
199	Root		
200	Root		
201	Root		
202	Root		
203	Root		
204	Root		
205	Root		
206	Root		
207	Root		
208	Root		
209	Root		
210	Root		
211	Root		
212	Root		
213	Root		
214	Root		
215	Root		
216	Root		
217	Root		
218	Root		
219	Root		
220	Root		
221	Root		
222	Root		
223	Root		
224	Root		
225	Root		
226	Root		
227	Root		
228	Root		
229	Root		
230	Root		
231	Root		
232	Root		
233	Root		
234	Root		
235	Root		
236	Root		
237	Root		
238	Root		
239	Root		
240	Root		
241	Root		
242	Root		
243	Root		
244	Root		
245	Root		
246	Root		
247	Root		
248	Root		
249	Root		
250	Root		
251	Root		
252	Root		
253	Root		
254	Root		
255	Root		
256	Root		
257	Root		
258	Root		
259	Root		
260	Root		
261	Root		
262	Root		
263	Root		
264	Root		
265	Root		
266	Root		
267	Root		
268	Root		
269	Root		
270	Root		
271	Root		
272	Root		
273	Root		
274	Root		
275	Root		
276	Root		
277	Root		
278	Root		
279	Root		
280	Root		
281	Root		
282	Root		
283	Root		
284	Root		
285	Root		
286	Root		
287	Root		
288	Root		
289	Root		
290	Root		
291	Root		
292	Root		
293	Root		
294	Root		
295	Root		
296	Root		
297	Root		
298	Root		
299	Root		
300	Root		
301	Root		
302	Root		
303	Root		
304	Root		
305	Root		
306	Root		
307	Root		
308	Root		
309	Root		
310	Root		
311	Root		
312	Root		
313	Root		
314	Root		
315	Root		
316	Root		
317	Root		
318	Root		
319	Root		
320	Root		
321	Root		
322	Root		
323	Root		
324	Root		
325	Root		
326	Root		
327	Root		
328	Root		
329	Root		
330	Root		
331	Root		
332	Root		
333	Root		
334	Root		
335	Root		
336	Root		
337	Root		
338	Root		
339	Root		
340	Root		
341	Root		
342	Root		
343	Root		
344	Root		
345	Root		
346	Root		
347	Root		
348	Root		
349	Root		
350	Root		
351	Root		
352	Root		
353	Root		
354	Root		
355	Root		
356	Root		
357	Root		
358	Root		
359	Root		
360	Root		
361	Root		
362	Root		
363	Root		
364	Root		
365	Root		
366	Root		
367	Root		
368	Root		
369	Root		
370	Root		
371	Root		
372	Root		
373	Root		
374	Root		
375	Root		
376	Root		
377	Root		
378	Root		
379	Root		
380	Root		
381	Root		
382	Root		
383	Root		
384	Root		
385	Root		
386	Root		
387	Root		
388	Root		
389	Root		
390	Root		
391	Root		
392	Root		
393	Root		
394	Root		
395	Root		
396	Root		
397	Root		
398	Root		
399	Root		
400	Root		
401	Root		
402	Root		
403	Root		
404	Root		
405	Root		
406	Root		
407	Root		
408	Root		
409	Root		
410	Root		
411	Root		
412	Root		
413	Root		
414	Root		
415	Root		
416	Root		
417	Root		
418	Root		
419	Root		
420	Root		
421	Root		
422	Root		
423	Root		
424	Root		
425	Root		
426	Root		
427	Root		
428	Root		
429	Root		
430	Root		
431	Root		
432	Root		
433	Root		
434	Root		
435	Root		
436	Root		
437	Root		
438	Root		
439	Root		
440	Root		
441	Root		
442	Root		
443	Root		
444	Root		
445	Root		
446	Root		
447	Root		
448	Root		
449	Root		
450	Root		
451	Root		
452	Root		
453	Root		
454	Root		
455	Root		
456	Root		
457	Root		
458	Root		
459	Root		
460	Root		
461	Root		
462	Root		
463	Root		
464	Root		
465	Root		
466	Root		
467	Root		
468	Root		
469	Root		
470	Root		
471	Root		
472	Root		
473	Root		
474	Root		
475	Root		
476	Root		
477	Root		
478	Root		
479	Root		
480			

Figure 5: Fau destination address

```

164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
```

Figure 6: Denic destination address

3 Wireshark Packet Filtering

1. ping to check the connection between your host and another in the network and apply display filter.

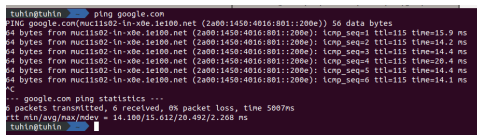


Figure 7: ping google

Ping sends Internet Control Message Protocol (ICMP) Request to a interface on the network and waiting for a reply.

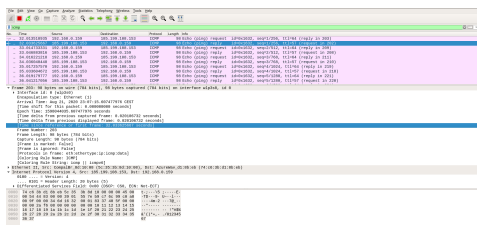


Figure 8: ICMP Display Filter

2. packets sent by your host while visiting the URL <http://www.caida.org/tools/visualization/map> and filter with destination address

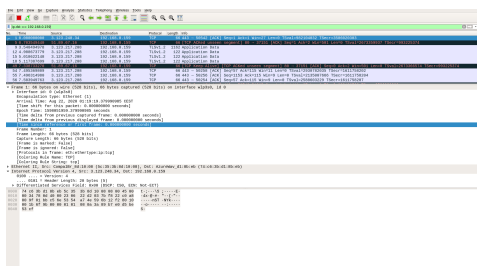


Figure 9: Packet sent by host to caida site

3. Host MAC address is:

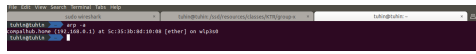


Figure 10: Host MAC address

Packets sent by host while visiting the URL <http://www.caida.org/tools/visualization/mapnet> and filter with source MAC address

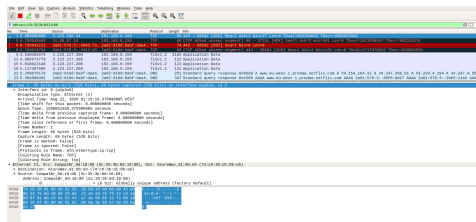


Figure 11: Packets filtered by MAC address sent by host to caida site

4. Difference between MAC and IP address:

MAC address or media access control address is a six byte hexadecimal address provided by NIC Card's manufacturer which ensures the physical address of a computer whereas IP address or Internet protocol address is either four byte IPv4 or six byte IPv6 address provided by the internet service provider or ISP which represents the logical address of a computer.

5. Yes. To communicate into LAN network directly rather than through router.

6. Packets that have been received by host:

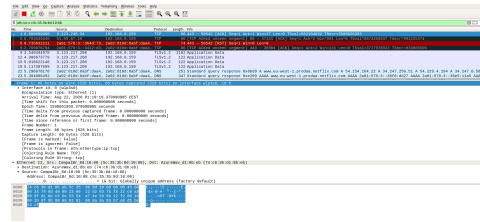


Figure 12: Packets received by host ip address

Host ip address is **192.168.0.159**

7. Tcpdump or capture filter expression that captures packets containing IP datagrams with a source or destination IP address equal to my IP address which is 192.168.0.159

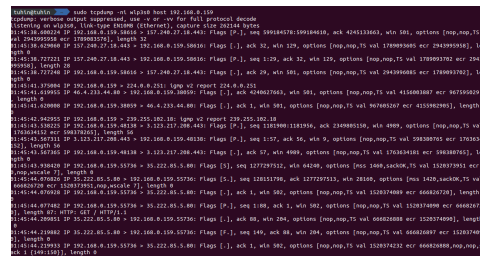


Figure 13: Tcp dump to source IP address 192.168.0.159

8. Syntax of a tcpdump or capture filter expression

```
tuhing@tuhin ~$ sudo tcpdump -ni wlp3s0 host 10.0.0.3 or 10.0.0.12
[sudo] password for tuhin:
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on wlp3s0, link-type EN10MB (Ethernet), capture size 262144 bytes
```

Figure 14: Tcp Dump between two hosts 10.0.0.3 and 10.0.0.12

```
tuhing@tuhin ~$ sudo tcpdump -ni eth0 host 10.0.0.3
tcpdump: eth0: No such device exists
SIOCGIFHWADDR: No such device)
tuhing@tuhin ~$ nml device status
DEVICE TYPE STATE CONNECTION
wlp3s0 wlan connected 10.0.0.12
enp2s0 ethernet unavailable --
lo loopback unmanaged --
tuhing@tuhin ~$ sudo tcpdump -ni eth0 host 10.0.0.3
tcpdump: eth0: No such device exists
SIOCGIFHWADDR: No such device)
tuhing@tuhin ~$ sudo tcpdump -ni wlp3s0 host 10.0.0.3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on wlp3s0, link-type EN10MB (Ethernet), capture size 262144 bytes
^C
0 packets captured
0 packets received by filter
0 packets dropped by kernel
tuhing@tuhin ~$
```

Figure 15: Paket containing Ip on host 10.0.0.3

9. Tcpdump or capture filter expression that captures TCP packets using port number 22

```
tuhing@tuhin ~$ sudo tcpdump -ni wlp3s0 tcp port 22
[sudo] password for tuhin:
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on wlp3s0, link-type EN10MB (Ethernet), capture size 262144 bytes
^C
0 packets captured
0 packets received by filter
0 packets dropped by kernel
tuhing@tuhin ~$
```

Figure 16: Tcp Dump for Port 22

10. Filter shows IP with destination IP equal to 192.168.178.1 and frame greater than 350 bytes

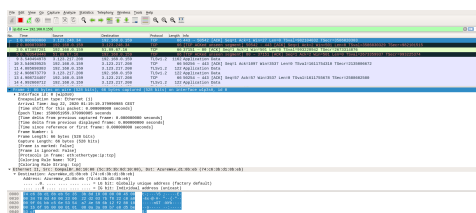


Figure 17: Filter Command

4 Taking Kathará for a Test Run

4.1 Experiment

After installing Kathará verifying it by following command:

```
tuhi@tuhi:~$ kathara check
* Current Manager is: Docker (Kathara)
* Manager version is: 19.03.12
* Python version is: 3.6.9 (default, Apr 18 2020, 01:56:04) - [GCC 8.4.0]
* Kathara version is: 2.2.3
* Trying to run 'hello world' container...
Deploying links...
Deploying machines... [#####] 1/1
* Container run successfully.
Deleting machines... [#####] 1/1
Deleting links...
tuhi@tuhi:~$
```

Figure 18: Kathará installation verification

1. All the functionalities can be shown by following command:

```
tuhi@tuhi:~$ kathara help
usage: kathara [-h] [-v] <command> [<args>]

Possible Kathara commands are:

vstart  Start a new Kathara machine
vclean  Stop a single Kathara machine
vconfig Attach network interfaces to a running Kathara machine
lstart  Start a Kathara Lab
lclean  Stop a Kathara Lab
linfo   Show information about a Kathara Lab
lrestart Restart a Kathara Lab
ltest   Test a Kathara Lab
lconfig Attach network interfaces to a running Kathara machine in a Kathara Lab
connect Connect to a Kathara machine
wipe    Delete all Kathara machines and links, optionally also delete settings
list    Show all running Kathara machines of the current user
settings Show and edit Kathara settings
check   Check your system environment

A network emulation tool.

positional arguments:
  command      command to run.

optional arguments:
  -h, --help    Show an help message and exit.
  -V, --version Print the current Kathara version.

tuhi@tuhi:~$
```

Figure 19: Kathará commands

2. A new container is created by following command:

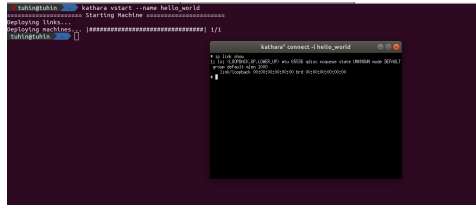


Figure 20: Kathará container creation

3. Network Interfaces for kathara container is displayed by following command:

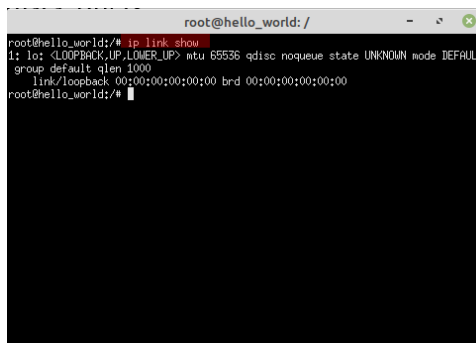


Figure 21: Kathará container network interfaces

4. Created kathara container is teared down by following command:

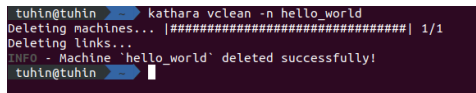


Figure 22: Kathará container tear down

5. Successfully first single node network is being set up.

5 Basic Linux network administration

5.1

```
tuhing@tuhin ~$ kathara vstart -n pc1 --eth 0:A
===== Starting Machine =====
Deploying links... |#####| 1/1
Deploying machines... |#####| 1/1
tuhing@tuhin ~$ kathara vstart -n pc2 --eth 0:B
===== Starting Machine =====
Deploying links... |#####| 1/1
Deploying machines... |#####| 1/1
tuhing@tuhin ~$ kathara vstart -n pc3 --eth 0:C
===== Starting Machine =====
Deploying links... |#####| 1/1
Deploying machines... |#####| 1/1
tuhing@tuhin ~$ kathara vstart -n switch --eth 0:A 1:B 2:C
===== Starting Machine =====
Deploying links... |#####| 3/3
Deploying machines... |#####| 1/1
tuhing@tuhin ~$
```

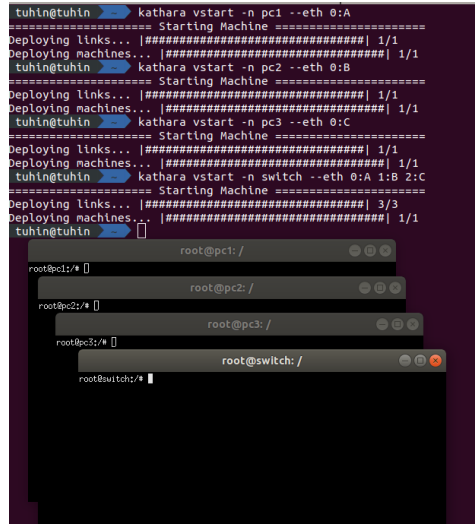


Figure 23: Create emulated pc with kathara command

```
tuhing@tuhin ~$ kathara vstart -n pc1 --eth 0:A
===== Starting Machine =====
Deploying links... |#####| 1/1
Deploying machines... |#####| 1/1
tuhing@tuhin ~$ kathara vstart -n pc2 --eth 0:B
===== Starting Machine =====
Deploying links... |#####| 1/1
Deploying machines... |#####| 1/1
tuhing@tuhin ~$ kathara vstart -n pc3 --eth 0:C
===== Starting Machine =====
Deploying links... |#####| 1/1
Deploying machines... |#####| 1/1
tuhing@tuhin ~$ kathara vstart -n switch --eth 0:A 1:B 2:C
===== Starting Machine =====
Deploying links... |#####| 3/3
Deploying machines... |#####| 1/1
tuhing@tuhin ~$
```

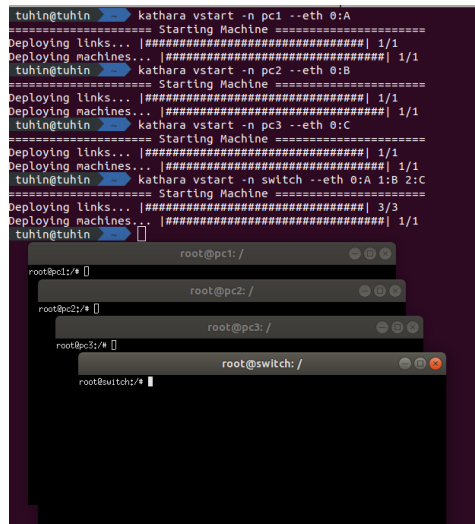


Figure 24: Assign network interfaces with emulated pc

5.2

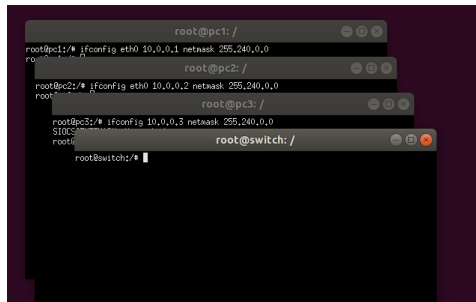


Figure 25: Setting up ip addresses in emulated pc

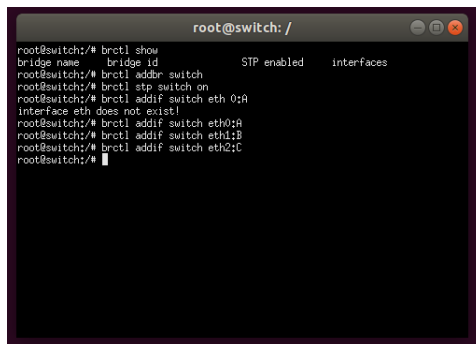


Figure 26: Create bridge and configure switch with brctl

5.3

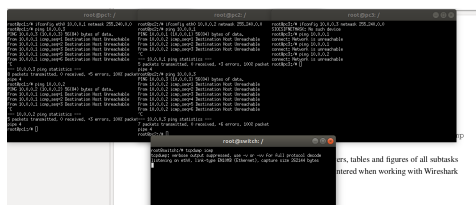


Figure 27: Ping and check if connection works

5.4

