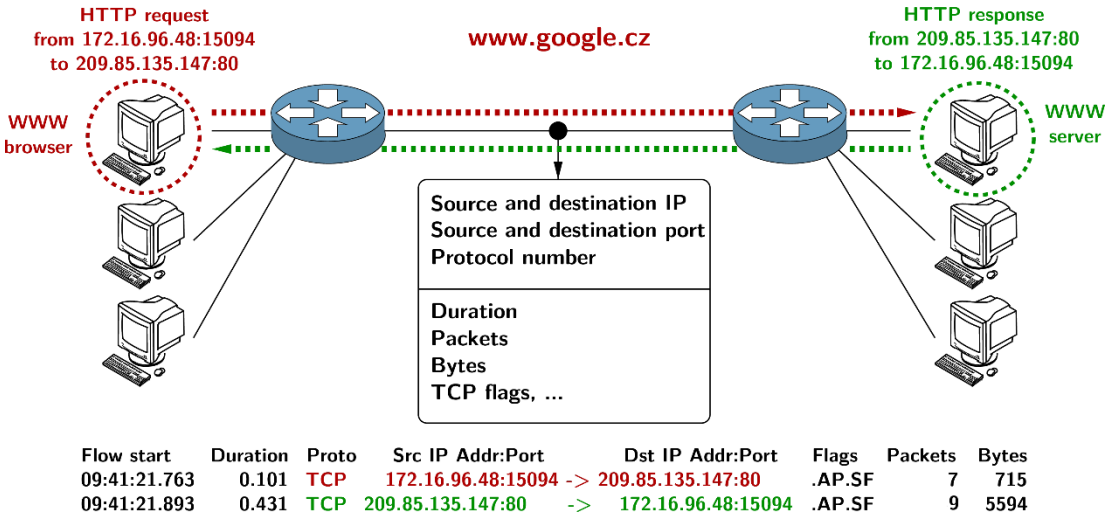
How do cybersecurity teams monitor network traffic?

Capturing packets and subsequently analyzing them is extremely demanding, both computationally and storage-wise. A single computer can generate an enormous amount of packets within a single second! That's why cybersecurity teams, such as CSIRT-MU (https://csirt.muni.cz/), store and analyze only **summary statistics** about the individual network connections, instead of whole packets.

The information about these connections is stored in the form of **network flows**. A simple network flow is composed of packets that share the following **five fields**: *source and destination IP address, source and destination port*, and *protocol*. This quintuple is typically extended with many other statistics, for example, duration of the flow, the total amount of packets, or the number of transferred bytes. (For detailed information, see RFC 3954 and RFC 7011.) While it may seem that this aggregation leads to losing relevant information, it's not true, as flows have **extensive applications** for network monitoring.

As the image below shows, the flow represents a **one-way connection** when TCP is used. The situation is even simpler with UDP, in which each packet forms exactly one flow.



Practical exercise

Now that you understand how network flows are created, will you dare to test your knowledge? You've received strips of paper that represent **individual packets** passing through the network. Each packet contains the following information:

Number	Arrival time	Source address	Destination address	Source port	Destination port	Protocol	Bytes transferred

Can you form **network flows** out of them? (The correct solution is on the other page.)

Solution: The resulting flows

Start time	Duration	Source address:port	Destination address:port	Protocol	Bytes transferred	Packet numbers
0.20	10.20	19.2.3.18:4020	125.14.55.111:80	TCP	440	1, 10, 11, 23
0.60	8.50	122.13.5.71:4000	125.14.55.111:80	TCP	360	2, 9, 21
0.90	0	122.13.5.71:4000	11.15.17.19:20200	UDP	110	3
1.60	8.10	19.2.3.18:22	20.20.40.44:22	TCP	570	4, 6, 16, 18, 22
1.90	0	125.14.55.111:4000	122.13.5.71:80	UDP	150	5
2.60	5.20	125.14.55.111:80	19.2.3.18:4020	TCP	490	7, 8, 17, 19
5.40	6.90	19.2.3.18:22	20.20.40.44:5187	TCP	410	12, 13, 24
6.40	0	122.13.5.71:4000	11.15.17.19:20200	UDP	100	14
6.80	0	125.14.55.111:4000	122.13.5.71:80	UDP	120	15
8.50	0	122.13.5.71:4000	11.15.17.19:20200	UDP	120	20

1.	0.20	19.2.3.18	125.14.55.111	4020	80	TCP	120
2.	0.60	122.13.5.71	125.14.55.111	4000	80	TCP	110
3.	0.90	122.13.5.71	11.15.17.19	4000	20200	UDP	110
4.	1.60	19.2.3.18	20.20.40.44	22	22	TCP	130
5.	1.90	125.14.55.111	122.13.5.71	4000	80	UDP	150
6.	2.20	19.2.3.18	20.20.40.44	22	22	TCP	110
7.	2.60	125.14.55.111	19.2.3.18	80	4020	TCP	120
8.	3.00	125.14.55.111	19.2.3.18	80	4020	TCP	140
9.	3.40	122.13.5.71	125.14.55.111	4000	80	TCP	100
10.	4.20	19.2.3.18	125.14.55.111	4020	80	TCP	100
11.	4.80	19.2.3.18	125.14.55.111	4020	80	TCP	120
12.	5.40	19.2.3.18	20.20.40.44	22	5187	TCP	150

13.	6.00	19.2.3.18	20.20.40.44	22	5187	TCP	150
14.	6.40	122.13.5.71	11.15.17.19	4000	20200	UDP	100
15.	6.80	125.14.55.111	122.13.5.71	4000	80	UDP	120
16.	7.10	19.2.3.18	20.20.40.44	22	22	TCP	100
17.	7.40	125.14.55.111	19.2.3.18	80	4020	TCP	130
18.	7.60	19.2.3.18	20.20.40.44	22	22	TCP	120
19.	7.80	125.14.55.111	19.2.3.18	80	4020	TCP	100
20.	8.50	122.13.5.71	11.15.17.19	4000	20200	UDP	120
21.	9.10	122.13.5.71	125.14.55.111	4000	80	TCP	150
22.	9.70	19.2.3.18	20.20.40.44	22	22	TCP	110
23.	10.40	19.2.3.18	125.14.55.111	4020	80	TCP	100
24.	12.30	19.2.3.18	20.20.40.44	22	5187	TCP	110