Synopsis

This document was created after we did the labs in 'Network Security'. The exercises correspond to the numbers. In these labs there were a number of "*Think abouts*", where you had to figure some stuff out. It was not mandatory but could be relevant for the lab review. So, I removed all answers for the concrete exercises and just left the answers for the *Think abouts*.

Exercise 1

rep-3

How could somebody on the defense side (i.e., in charge of the network security) make the most of whois against attackers?

using an attacker's IP address a defender can search with WHOIS for an registrant of the given IP. This would allow a defender to find out where the attack originated from.
Even if this IP is only a bot or a cloud provider like cloudflare a defender can still receive contact information.
Furthermore from a defensive point of view, the WHOIS database might tell you:

- that you are exposing too much information about yourself in your own domain info (owners name, address, phone number, etc.).
- how long domains have been registered so that you can use automated tools to block domains that are very young (and likely to be malicious if suddenly appearing in emails).
- who to contact in the event that a legitimate domain is sending spam or hosting malicious content.

rep-5

Are mail servers hosted by the same company? Depending on the company, the answer to this question can be "yes" or "no". Considering each of these possibilities, does it make sense targeting mail servers as potential vectors for penetration attacks?

“self-hosted”: yes it does make sense, s. Link. when self-hosted, problems are for example, admins have to recognize attacks, report vulnerabilities and upgrade software when patches are available. Further problems include letting users change their passwords, e. g. after a successful exploit and exposed username/password combinations. Until all users have changed their passwords in a large organization, further services could have been compromised.

“cloud-hosted”: yes, there are possibilities of attacks. even if the service is cloud-hosted and has protections against DDOS attacks like Cloudflare, there is the possibility that such servers which host many clients could be a viable target for a large-scale attack when enough services are put under distress.

Potential risks/attacks:

- No SSL (client to server) -> man-in-the middle
- Weak passwords -> if an account gets hijacked, it can be used for e.g. ‘social engineering’, also lots of other accounts might get revealed.
- reuse of username/password combinations: users often reuse passwords for multiple sites, a compromised mail server could lead to further exposure

Protect:

- Configure, Protect, and Analyze Log Files
- Back up Data
- malware scanning and spam filtering
- Security Testing (e.g. vulnerability scanning)

rep-6

Wireshark

Imagine using Wireshark for checking all the traffic passing through an intermediate routing device. Do you think that you could detect hosts performing horizontal scanning? And vertical scanning? Do you consider Wireshark as a suitable tool for analyzing large amounts of network traffic data? Why?

before an attack: NO, I don't think it is appropriate for it.
after an attack: Wireshark is good for after-attack analysis, to find out how an attack happened.
For the concrete filtering, aggregation and possible pattern analysis to prevent attacks on a routing device we would use other tools.

Exercise 2:

Think about it

Do you think that Go-Flows has any advantage compared with tcpdump?

In comparison to tcpdump, we can get aggregated results, which are more easily used in plots and are usable for machine learning. For the process of manual analysis *Go-Flow* is easier to use in comparison to tcpdump. *tcpdump* is mainly used in scraping the network data, whereas Go-Flow is used to aggregate / transform the pcap files in a better suited format.

What are the proportions of TCP, UDP, and ICMP traffic? And traffic that is not TCP, UDP, or ICMP?

Listing of Protocols, with Identifier in parentheses and percent. of traffic:

- ICMP (1) + IPv6-ICMP (58): ~ 39%
- TCP (6): ~ 45%
- UDP (17): ~ 11%
- GRE (47): ~ 4%

How much traffic is related to websites (HTTP, HTTPS)? And DNS traffic?

Websites (port 80,443): 4.319161520033035

DNS (port 53): 0.3857645045815537

In this pcap packet there is actually none which is using the HTTP or HTTPS protocol.

Think about:

Remember that here we have extracted flows within a time-frame of 10 seconds. Can you think about legitimate and illegitimate situations for case (c), i.e., a source sending traffic to many different destinations in a short time?

You can additionally count the number of flows that show TCP, UDP, ICMP, and other IP protocols as "mode" protocol. Do you think that you will get a similar proportion as in [rep-11]? Beyond answering "yes" or "no", think about reasons that might make such proportions similar or different (there are some that are worth considering).

illegitimate reasons: preparation for a later attack, checking (repeatedly!) open ports of active services with (zero-day) vulnerabilities on a large scale

legitimate reasons: scanning for botnets after an attack, checking if bots are responsive to shutdown commands to mitigate attacks

think about Google, building their search engine: they also have to scan the internet for newly available / updated websites, so their traffic would also look like that

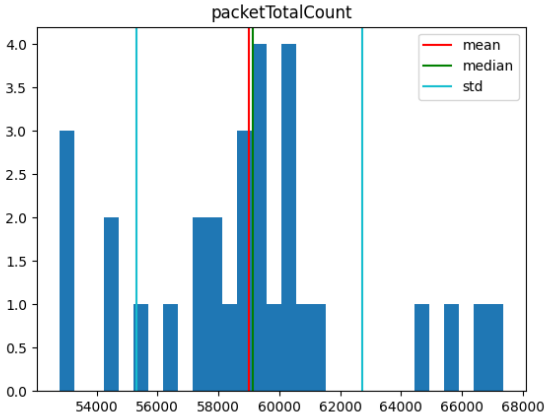
Further, there are scenarios, where global events increase streams from one source to multiple destinations like broadcasts of major sport events or natural disaster news coverage.

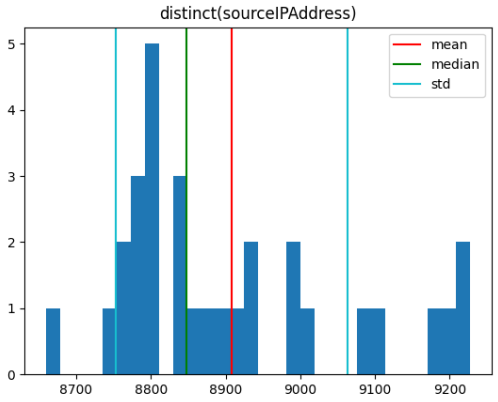
2.4 Think about it

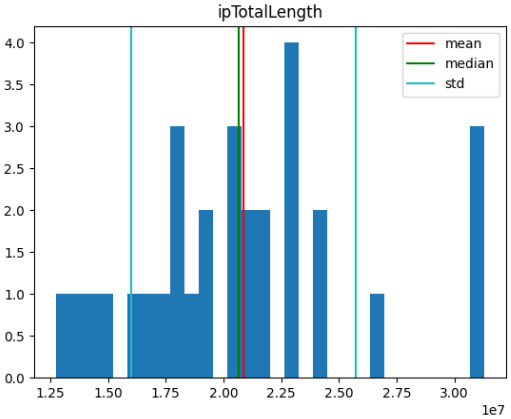
It is obvious that the three explored time series have different order-of-magnitude, but are they correlated? Time series must be plotted, so we encourage you to do that. Depending on the analysis platform (Python, MATLAB, R, etc.), you have commands that evaluate correlations between signals by outputting a numerical value (0: no correlation, 1: maximum direct correlation, -1: maximum inverse correlation). However, whenever possible, we recommend using plots and visual representations. Plot the three time-series. To better assess correlations, you can scale/normalize signals before plotting them.

Additionally, you can assess value distributions by plotting histograms. We recommend also plotting central tendency values (mean, median, standard deviation) superposed on the histograms to check if they are representative of the data. Are they?

For the *totalPacketCount* it is representative, since Median and Mean are near to each other. The same can be said about the *ipTotalLength*. In contrast, the *sourceIpaddresses* are more diffused, so mean and median don't really say that much about it.







Ex 3-2, Think about

Q: Are the results in [rep-14] correlated?

Yes, since the correlation reaches 0.97 in (a) and (b), *pkts*, *unique IP destination* and *bytes* are correlated. Only unique source IP addresses are not that strongly correlated.

The drop in unique source IP addresses did not cause that much effect on the other results since they are not correlated. This is also reflected in the Pearson correlation result of 0.41.

Q: [rep-15], Do results make sense for you?

Yes. The traffic which we see here, was recorded from not registered ip addresses in dark space. A smaller amount of scans of the ip range by fewer devices (unique ip addresses) was done in comparison to the destination address space. This would explain the difference in ratio.

Opinion on Usage of median / mean:

Often, we need both statistical indicators for a data set to really figure out where the “middle” of the data distribution hill is located. E.g., if standard deviation is small, and median and mean are nearly the same, we can assume that the actual middle of the distribution should also be near them. Visual checks are still necessary though. Like in the previous example and as shown in the images above, there is still uncertainty if the data is not actually completely random and distributed far away from our computed mean / median.

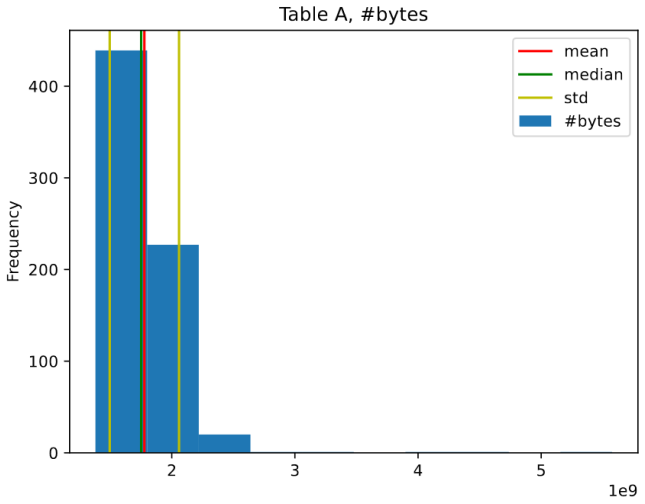
Pros for mean: if the distribution is skewed, we often find that the highest value is better represented by mean than median.
On the contrary, if some values appear much more often on one side of the distribution, but amount to less than the average, than the median could signal a more accurate point for the “middle” of the (distribution) hill.

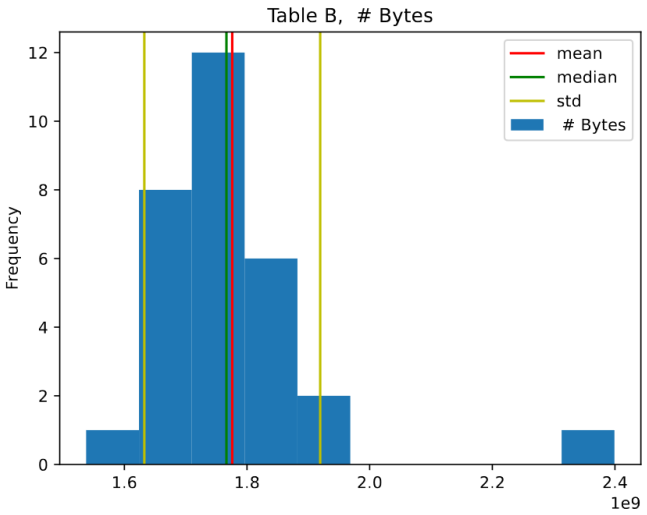
Some of those problems can be reduced by using normalization and scaling but still remain in most problems.

Ex 3-3

Q: does the data in the tables A (hourly averaged bytes) and table B (averaged over day) coincide?

Yes, for the most part, the data seems the same. Table B has some outliers, which moves the mean to the right, but otherwise the distributions look much alike.





Box plots specialty so to say, are to show only the statistical indicators for a data distribution. for the above Bytes example, it shows the outliers at table B. This contrasts the image we get from table A. Also, when only looking at the means, medians and std we find that the two tables differ much more than what we get from looking at the respective histograms.

Q: Did you get negative values in [rep-19]? Can you figure out why? And why not in the case of packets?

Yes, we got negative values for (b) and (c). Reason is that the monthly average of unique ips is lower than the average mean of the three major protocols combined, resulting in a negative value. The same goes for the unique IP destinations. This could be explained with that other protocols are not used that much for example for broadcasting which results in a far lower unique IP source / destination average.

as a formula:

$$< other - sources > \; = \frac{'monthly\ unique\ IP\ sources' - uIP1 - uIP17 - uIP6}{'monthly\ unique\ IP\ sources'}$$

The "*monthly unique IP sources*" is from the file `team??_monthly.csv`
whereas the other data is from the file `team??_protocol.csv`

For packets, the margin was nearly 0. for packets, it seems, there isn't that much difference between the average packets size for one of the major protocols in comparison to others. Therefore, the percentage seems closer to the real usage of such protocols.

Ex 3-4

Yes, the signals show periodicities, in particular ~~(a) repeats itself once a day and (b) repeats itself every 30 days, so each month.~~

- #pkts/hour:
  k=1 means it repeats once in the given timespawn (which actually is no repetition), hence I'd say it doesn't repeat

- #uIPs/hour: k=30 means: in a total timespan of 720 hours it repeats 30 times which results in a period time of: 720/30 = 24h which is once per day.

Ex 4-2

It makes sense that most TCP flags in use are the SYN flags, since they are used to look for open (TCP) ports and only when a connection is accepted, an ACK (TCP-) flag is returned. Therefore, SYN will be naturally used more often than ACK.

Q: Does the TTL plot show mountain-like shapes? If so, can you figure out why?

Yes, there are mountain-like shapes, at around 60, 90, 120, 180 and 240 sec TTL. [ACK] seem to be send more often with 60 or 90 sec. to live, while e.g. [SYN] is sent completely spread out. This has something to do with how SYN packets are built by different OS. As we researched this on the web, the following posts explained it in more depth., see this link and this here. Further, each router decreases the TTL counter and when a router receives a packet with TTL 0, then a packet will be dropped.

Remark: the SYN TTL spread can be explained further by how a packet is sent a long the hierarchy up until one router knows the target IP. Until then, the TTL is reduced by each router along the way. In comparison, ACK will be sent more directly and will possibly avoid unnecessary hops. Therefore, the TTL is more steady / less often reduced.

Ex 4-3

The first flow in [rep-23] should be a DDOS attack, using the ICMP flood pattern. Since the TTL is not the same for most of the requests, it seems that a distributed botnet was used for the attack (different flows, different amount of hops). So yes, this flow seems to be malicious. The target is the address 203.74.52.109, since it is pinged first from all the others.

The second flow seems to be a normal server. The used port is 80 (HTTP) and the packet rate is not out of the norm for this particular flow. Also, the total amount of packets sent from or to this ip address is not out of the ordinary.