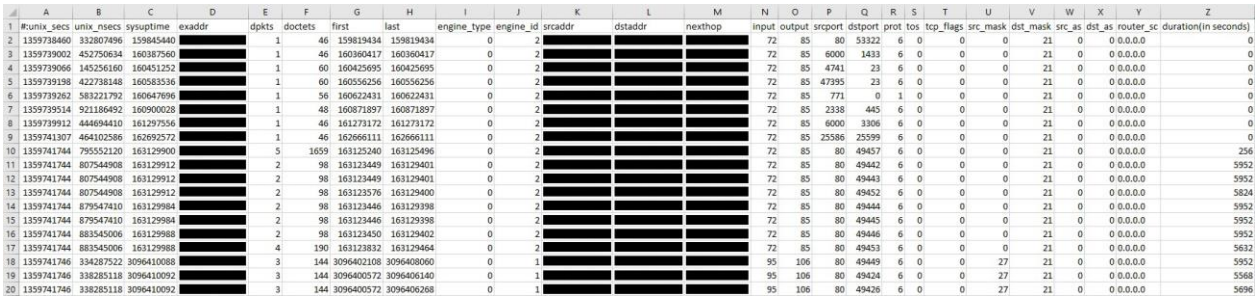
Profiling Internet Users

The source of data for this project is Cisco NetFlow version 5, which is one of the most popular technologies to collect IP traffic. Many parameters can be extracted from the source data including; Packets, Octets, beginning and ending of each flow, source and destination port numbers, source and destination IP addresses and many other variables which are included in the following figure.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	#unix_secs	unix_nsecs	sysuptime	exaddr	dpkts	doctets	first	last	engine_type	engine_id	srcaddr	dstaddr	nexthop	input	output	srcport	dstport	prot	tos	tcp_flags	src_mask	dst_mask	src_as	dst_as	router_sc	duration(in seconds)
2	1359738460	332807496	159845440	[redacted]	1	46	159819434	159819434	0	2	[redacted]	[redacted]	[redacted]	72	85	80	53322	6	0	0	0	21	0	0	0.0.0.0	0
3	1359739002	452750634	160387560	[redacted]	1	46	160360417	160360417	0	2	[redacted]	[redacted]	[redacted]	72	85	6000	1433	6	0	0	0	21	0	0	0.0.0.0	0
4	1359739066	145256160	160451252	[redacted]	1	60	160425695	160425695	0	2	[redacted]	[redacted]	[redacted]	72	85	4741	23	6	0	0	0	21	0	0	0.0.0.0	0
5	1359739198	422738148	160583536	[redacted]	1	60	160556256	160556256	0	2	[redacted]	[redacted]	[redacted]	72	85	47395	23	6	0	0	0	21	0	0	0.0.0.0	0
6	1359739262	583221792	160647696	[redacted]	1	56	160622431	160622431	0	2	[redacted]	[redacted]	[redacted]	72	85	771	0	1	0	0	0	21	0	0	0.0.0.0	0
7	1359739514	921186492	160900028	[redacted]	1	48	160871897	160871897	0	2	[redacted]	[redacted]	[redacted]	72	85	2338	445	6	0	0	0	21	0	0	0.0.0.0	0
8	1359739912	444694410	161297556	[redacted]	1	46	161273172	161273172	0	2	[redacted]	[redacted]	[redacted]	72	85	6000	3306	6	0	0	0	21	0	0	0.0.0.0	0
9	1359741307	464102586	162692572	[redacted]	1	46	162666111	162666111	0	2	[redacted]	[redacted]	[redacted]	72	85	25586	25599	6	0	0	0	21	0	0	0.0.0.0	0
10	1359741744	795552120	163129900	[redacted]	5	1659	163125240	163125496	0	2	[redacted]	[redacted]	[redacted]	72	85	80	49457	6	0	0	0	21	0	0	0.0.0.0	256
11	1359741744	807544908	163129912	[redacted]	2	98	163123449	163129401	0	2	[redacted]	[redacted]	[redacted]	72	85	80	49442	6	0	0	0	21	0	0	0.0.0.0	5952
12	1359741744	807544908	163129912	[redacted]	2	98	163123449	163129401	0	2	[redacted]	[redacted]	[redacted]	72	85	80	49443	6	0	0	0	21	0	0	0.0.0.0	5952
13	1359741744	807544908	163129912	[redacted]	2	98	163123576	163129400	0	2	[redacted]	[redacted]	[redacted]	72	85	80	49452	6	0	0	0	21	0	0	0.0.0.0	5824
14	1359741744	879547410	163129984	[redacted]	2	98	163123446	163129398	0	2	[redacted]	[redacted]	[redacted]	72	85	80	49444	6	0	0	0	21	0	0	0.0.0.0	5952
15	1359741744	879547410	163129984	[redacted]	2	98	163123446	163129398	0	2	[redacted]	[redacted]	[redacted]	72	85	80	49445	6	0	0	0	21	0	0	0.0.0.0	5952
16	1359741744	883545006	163129988	[redacted]	2	98	163123450	163129402	0	2	[redacted]	[redacted]	[redacted]	72	85	80	49446	6	0	0	0	21	0	0	0.0.0.0	5952
17	1359741744	883545006	163129988	[redacted]	4	190	163123832	163129464	0	2	[redacted]	[redacted]	[redacted]	72	85	80	49453	6	0	0	0	21	0	0	0.0.0.0	5632
18	1359741746	334287522	3096410088	[redacted]	3	144	3096402108	3096408060	0	1	[redacted]	[redacted]	[redacted]	95	106	80	49449	6	0	0	27	21	0	0	0.0.0.0	5952
19	1359741746	338285118	3096410092	[redacted]	3	144	3096400572	3096406140	0	1	[redacted]	[redacted]	[redacted]	95	106	80	49424	6	0	0	27	21	0	0	0.0.0.0	5568
20	1359741746	338285118	3096410092	[redacted]	3	144	3096400572	3096406268	0	1	[redacted]	[redacted]	[redacted]	95	106	80	49426	6	0	0	27	21	0	0	0.0.0.0	5696

To preserve the privacy, all the IP addresses are removed from the data. 54 Excel files are included in the project which each file corresponds to one user. Data is captured for a month long period. In average, the number of flows for each subject over a week worth of data is more than 7000.

You may download the files from the link below:

https://drive.google.com/drive/folders/1yoCnGBqO9mw9IMo3fTUKkpagqc7FB6ci

Parameter to use for profiling: In this project, we want to demonstrate if the Internet usage of each subject is statistically indistinguishable when compared to the Internet usage of the same subject over time, while simultaneously being statistically distinguishable when compared to Internet usage of other subjects. Subsequently, we want to study how the time window chosen for profiling affects the answer to the above problem. You can implement a profile for each user based on many criteria; but, we suggest to use **octets/duration**. Duration can be obtained from "first" and "last" columns, and Octets can be obtained from "doctets" field in the Excel sheets.

Computing Flow Durations: You should write a program in any language that you are familiar in order to get network data as the input and do the statistical analysis to find the distinguishability or indistinguishability between subjects. Each file should be opened and compared with rest of the files. It is important to mention that each file should be split in parts. For example, you could split into groups of duration as small as 10 seconds or as high as 24 hours. But, for this project, you should compare three time windows of **10 seconds**, **227 seconds**, and **5 minutes** to find out which time window has the least number of users that are statistically distinguishable when compared to Internet usage of other subjects while simultaneously statistically indistinguishable when compared to the Internet usage of the same subject. Each time window has several flows, you may find the average for the variable octets/duration in each window. Note that some flows have a duration of 0 which due to the reason that the granularity is too short, the duration is 0 millisecond. Since you need to divide octets over duration and dividing by zero is undefined, you may not

consider flows with the duration of 0. For flows with durations longer than 10 seconds, or 227 seconds, or five minutes, simply average the octets to create smaller flows of durations of 10 seconds, or 227 seconds, or five minutes.

Data Splitting per Day: Data splitting can be done only based on the column that is named "Real First Packet" and included in the excel files. You do not need to consider the "Real End Packet" column. The "Real First Packet" column shows the initial date and time of each flow in epoch format. You may convert this value to a human readable data and time. You may use the following link to find a method in most of the programming languages for this conversion.

https://www.epochconverter.com/

Statistical Computations: You may do some initial calculations on the Excel files like calculating the ratio of octets/duration. However, it is recommended to write all the tasks in the programing that you are familiar. In this way, it is much easier to keep track of the data from input to the output.

For the statistical analysis part, you may use the flowing steps [1].

1. After splitting the data into parts as explained before, you need to find the correlation between them. Since you are comparing correlations across weeks, you should split the months' worth of Internet usage data into four groups each for four weeks for all subjects. A brief snapshot of two weeks data for two subjects across time is shown in the following figure. In the following sample, window of 227 seconds is chosen. Column on the left is showing for a sample *"User A"* and the column on the right is showing a sample data for *"User B"*. Each row represents a window. For an instance, first row represents data for Monday from 00:00:00am to 00:03:47am which is a 227-second window with the value of 6.3972 for the parameter of octets/duration. Similar procedure was done until Friday 11:56:13pm to 00:00:00am that is the last window in the week.

For this project, you can choose to derive these windows between 8:00am and 5:00pm on Weekdays (no Saturday or Sunday) to speed up computations without losing the meaning of the project.

[1] If you see NetFlow record that is highly anomalous, it could be due to processing errors (since router data are heavily sampled and processed). Feel free to ignore such anomalous data, and mention a note in your report.

	User A		User B		
	Time	Octets/Duration	Time	Octets/Duration	
Week 1	Monday (00:00:00am-00:03:47am)	6.3972	Monday (00:00:00am-00:03:47am)	0.0302	Week 1
	⋮	⋮	⋮	⋮	
	Monday (11:56:13pm-00:00:00am)	4.9369	Monday (11:56:13pm-00:00:00am)	13.7590	
	Tuesday (00:00:00am-00:03:47am)	5.0646	Tuesday (00:00:00am-00:03:47am)	1.4598	
	⋮	⋮	⋮	⋮	
	Tuesday (11:56:13pm-00:00:00am)	4.2846	Tuesday (11:56:13pm-00:00:00am)	0.7783	
	Wednesday (00:00:00am-00:03:47am)	5.7988	Wednesday (00:00:00am-00:03:47am)	2.6305	
	⋮	⋮	⋮	⋮	
	Wednesday (11:56:13pm-00:00:00am)	2.3436	Wednesday (11:56:13pm-00:00:00am)	6.2205	
	Thursday (00:00:00am-00:03:47am)	2.4772	Thursday (00:00:00am-00:03:47am)	0.0000	
	⋮	⋮	⋮	⋮	
	Thursday (11:56:13pm-00:00:00am)	3.1775	Thursday (11:56:13pm-00:00:00am)	0.0000	
	Friday (00:00:00am-00:03:47am)	4.8082	Friday (00:00:00am-00:03:47am)	9.1049	
	⋮	⋮	⋮	⋮	
	Friday (11:56:13pm-00:00:00am)	5.0530	Friday (11:56:13pm-00:00:00am)	0.0000	
Week 2	Monday (00:00:00am-00:03:47am)	6.4694	Monday (00:00:00am-00:03:47am)	2.0793	Week 2
	⋮	⋮	⋮	⋮	
	Monday (11:56:13pm-00:00:00am)	4.3542	Monday (11:56:13pm-00:00:00am)	36.1807	
	Tuesday (00:00:00am-00:03:47am)	8.2608	Tuesday (00:00:00am-00:03:47am)	4.2334	
	⋮	⋮	⋮	⋮	
	Tuesday (11:56:13pm-00:00:00am)	8.1370	Tuesday (11:56:13pm-00:00:00am)	4.3147	
	Wednesday (00:00:00am-00:03:47am)	12.6390	Wednesday (00:00:00am-00:03:47am)	4.8411	
	⋮	⋮	⋮	⋮	
	Wednesday (11:56:13pm-00:00:00am)	12.6685	Wednesday (11:56:13pm-00:00:00am)	3.4661	
	Thursday (00:00:00am-00:03:47am)	11.6330	Thursday (00:00:00am-00:03:47am)	14.3444	
	⋮	⋮	⋮	⋮	
	Thursday (11:56:13pm-00:00:00am)	14.2283	Thursday (11:56:13pm-00:00:00am)	1.1753	
	Friday (00:00:00am-00:03:47am)	13.3379	Friday (00:00:00am-00:03:47am)	7.6747	
	⋮	⋮	⋮	⋮	
	Friday (11:56:13pm-00:00:00am)	17.3506	Friday (11:56:13pm-00:00:00am)	10.0920	

2. At this step you need to calculate the correlation coefficient values. There are three main type of correlation coefficients. In this project, it is recommended to use the Spearman's correlation coefficient. You may calculate it in Excel or even you can find the formula and implement it in your code. You need to find three correlation values of r_{1a2a}, r_{1a2b} and r_{2a2b}. Numbers are showing the weeks and characters are showing the subjects. For example r_{1a2a} denotes the Spearman's correlation coefficient between Internet usage of *"Subject a"* for week 1 with Internet usage of *"Subject a"* for week 2. Similarly, r_{1a2b} denotes the Spearman's correlation coefficient between Internet usage of *"Subject a"* for week 1 with Internet usage of *"Subject b"* for week 2 and r_{2a2b} denotes the Spearman's correlation coefficient between Internet usage of *"Subject a"* for week 2 with Internet usage of *"Subject b"* for week 2. For the calculation, you may use the following formula in your code:

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)}$$

d_i: The difference between the ranks of corresponding variables

n: Number of observations (Number of windows in a week)

If you are not familiar with Spearman's correlation you may learn from the link below:

https://www.wikihow.com/Calculate-Spearman%27s-Rank-Correlation-Coefficient

3. Based on the correlation values which are calculated in the previous step, the main part of this project can be done. For the statistical framework of this project, Meng, Rosenthal, and Rubins Z Test Statistic (MRR-Z test) can be employed to find the value of Z. Required formulas are included below. Correlation coefficients that are calculated in the last step can be imported in the Z formula.

$$Z = [Z_{1a2a} - Z_{1a2b}] * \frac{\sqrt{[N-3]}}{2 * [1 - r_{2a2b}] * h}$$

$$Z_{1a2a} = \frac{1}{2} log \frac{1 + r_{1a2a}}{1 - r_{1a2a}}$$

$$Z_{1a2b} = \frac{1}{2} log \frac{1 + r_{1a2b}}{1 - r_{1a2b}}$$

$$h = \frac{1 - [f * rm^2]}{1 - rm^2}$$

$$f = \frac{1 - r_{2a2b}}{2 * [1 - rm^2]}$$

$$rm^2 = \frac{r_{1a2a}^2 + r_{1a2b}^2}{2}$$

N: Sample size of the data set (Number of windows in a week)

4. Based on the Z value calculated from the previous part, the corresponding P-value can be computed as follows:

$$P = 1 - \Phi(Z)$$

where $\Phi(Z)$ is the cumulative distribution function of standard normal distribution. You may use the following function in your code to find the P-value. This function is written in C# but you may modify it to use in any other languages.

```
static double PFunction(double z)
{
    double p = 0.3275911;
    double a1 = 0.254829592;
    double a2 = -0.284496736;
    double a3 = 1.421413741;
    double a4 = -1.453152027;
    double a5 = 1.061405429;

    int sign;
    if (z < 0.0)
        sign = -1;
    else
        sign = 1;

    double x = Math.Abs(z) / Math.Sqrt(2.0);
    double t = 1.0 / (1.0 + p * x);
    double erf = 1.0 - (((((a5 * t + a4) * t) + a3)
      * t + a2) * t + a1) * t * Math.Exp(-x * x);
    return 0.5 * (1.0 + sign * erf);
}
```

5. Finally, based on the value that calculated from the previous step, you can decide that two users are distinguishable or indistinguishable from each other. When $P \leq 0.05$ means that correlation

coefficient calculated for Internet usage patterns for an unknown subject (say b) is significantly smaller than that for a known subject (say a) and as such *"subject b"* will be identified as a subject distinct from *"subject a"*. On the contrary, when $P > 0.05$, indicates that correlation coefficient calculated for Internet usage patterns for an unknown subject (say b) is not significantly smaller than that for a known subject (say a), and as such *"subject b"* will be identified as indistinguishable from *"subject a"*.

To finish, you need to write a report and briefly explain the procedure and include a table, which shows three time windows of **10 seconds**, **227 seconds**, and **5 minutes** that used in the project and the average number of matches (average number of *P* values that are greater than 0.05 across all the users) for each window and state which window is better in terms of authentication. By better authentication, what we mean is that, you need to find out at what time window, it happens that a) each user's data in one week is statistically indistinguishable from the same user's data across another week; and b) the number of other users whose data is statistically indistinguishable from a particular user is minimum (ideally 0).

For this project, you will report results by comparing data across Week 1 and Week 2

You will submit all code, and a report as a Zip file on Canvas. Deadline is announced on Canvas. Earlier submissions are encouraged, so that grader can verify if all files are there.

Also, note that if a particular value of r (either, r_{1a2a}, r_{1a2b} and r_{2a2b}) is equal to 1, feel free to use 0.99 instead, so that you don't run into division by zero problems.

Addendum: Please note that you will need to submit a report with three Tables first. Each Table will look like the one below, where each cell lists a p-value. For instance, in the Table below, the diagonal entries are p-values that compare Week 1 Internet data for ONE user with Week 2 data of the SAME user. Non-diagonal entries should make sense also. For instance, Row 1 Column 2 (shaded) will have p-value for User 1's Week 1 data with User 2's Week 2 data. Similarly, Row 4 Column 3 will have p-value for User 4's Week 1 data with User 3's Week 2 data. You will generate three tables though like the one below for three time windows of comparison: **10 seconds**, **227 seconds**, and **5 minutes.** Each cell in each table will have one p-value. **Mention Time Window and Weeks used in Table Caption.**

Please contact me if you have questions on this representation.

Then, you will need to write a small (a paragraph or two) report indicating what you see from the tables. You could write on the degree to which a single user's data exhibits repeatability over time, and also the degree of distinguishability across weeks between different users. Any anomalies you see, or other interesting insights can be mentioned. Write about the impact of time window size on profiling.

Then, in the report, write clearly how the grader needs to execute your code. Specify which weeks are used for Table generation, and how the grader can see the final p-value for all tables. The grader will actually execute your code to see if entries in Table you entered match the p-values

computed by the grader (for a few random entries) for different window sizes. Code will be checked for plagiarism.

For this project, it is enough if you generate the Table below for any two weeks (preferably Weeks 1 and 2).

You will generate three tables though for three time windows **10 seconds**, **227 seconds**, and **5 minutes.**

Weeks 1 &2	User 1	User 2	User 3	User 4	……	User 54
User 1						
User 2						
User 3						
User 4						
……..						
User 54						