Difficulties of Applying Timing Analysis

January 13, 2019

1 Traditional Timing Analysis

One of the difficulties in timing analysis is characterizing timing responses $t = T(k,m)$ given a key message pair (k,m) and the corresponding timing measurements. The value of t could be as simple as a real number or as complicated as some insane probability distribution. This does matter since different attacking schemes require different types of inputs. The following paragraphs describe some interesting properties of this HSM's measurements and how these features help itself to resist traditional timing attacks.

1.1 Double-Concentration Property

A common assumption in all of the traditional timing cryptanalysis is the timing responses $t = T(k,m)$ is a simple real number. That is, timing measurements of a fixed key message pair should concentrate on a single horizontal line. However, this HSM does not behave this way. The timing responses we measured concentrate on two different horizontal lines instead of one. Figure 1 shows 2000 samples of timing response using a single private key to sign a text file containing only two characters "77". It can be easily seen from the figure that most of the timing responses are either close to 5.27945 or 5.27960. The timing difference between the two lines is about 0.00015

There are several naive options to cast this kind of timing measurements to a real number such as using average of all samples and either one of the concentrations. Average seems to be better at the first glance since we don't know whether the distribution of timing responses will remain consistent. There might be the case that more responses locate at the upper concentration when signing one message while more responses locate at the lower concentration when signing another message. Using either one of the concentration fails to characterize their weights. However, there could still be some problems with using average of all samples. For example, when the difference between two concentrations d is not a constant, a timing response with a large d may result in the same value as a timing response with a small d. It seems that neither of the approaches are perfect without any further information. This motivates us to do the second experiment, testing whether the distribution will remain the same and whether the difference of the two concentration is constant when the

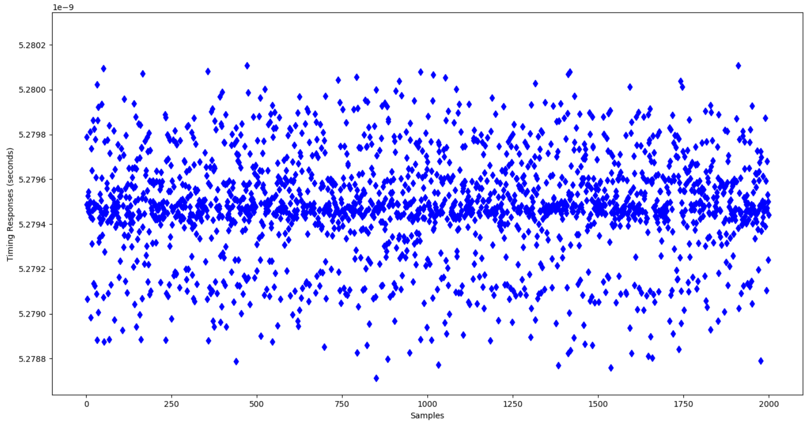

Figure 1: M1-K1 Timing Measurement

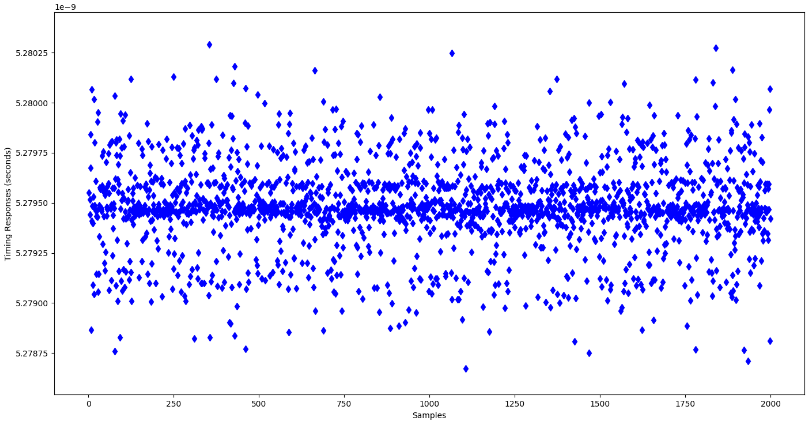

Figure 2: M2-K1 Timing Measurement

key is fixed. In this experiment, we used another file containing ”66” as our new input message and measure the timing responses again. Figure 2 shows that using different messages still results in similar distribution, and it can be seen that the timing difference between these two concentrations remains the same. It seems that all of the approaches aforementioned are suitable. For simplicity, we will use the average of all samples so that we don’t need to deal with the problem of clustering these data.

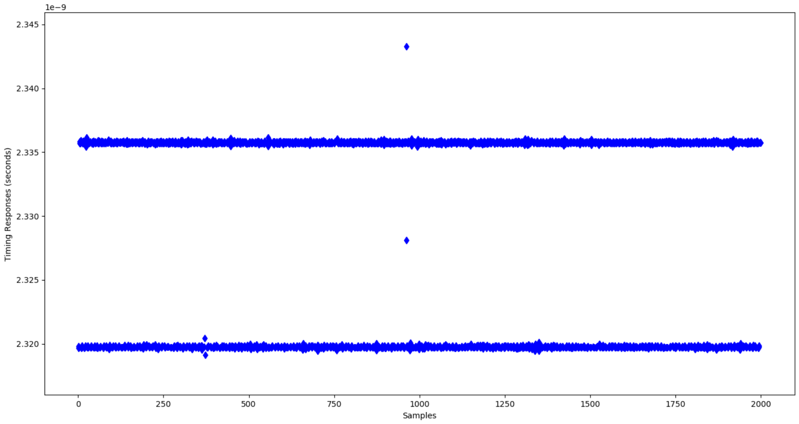

Figure 3: M1-K2 Timing Measurement

1.2 Small and Large Concentrations

Suspecting that double-concentration property may not show up when signing with a different key, we sign the file containing ”77” (M1) again using a different key (K2) 2000 times (see figure 3). This time, the double-concentration characteristic shows up again. More specifically, it shows up thrice in this experiment. The first one separated the timing responses into two large concentrations L_u, L_l, then the second and the third further divided these two clusters into four small concentrations $S_{uu}, S_{ul}, S_{lu}, S_{ll}$ (see figure 4 and 5). From the figures, we can see that the timing differences of S_{uu}, S_{ul} and S_{lu}, S_{ll} are both about 0.00015; while the timing difference of L_u, L_l is about 0.015, which is about 100 times of the smaller ones. In addition, the differences between S_{uu}, S_{ul} and S_{lu}, S_{ll} are of the same magnitude order as the one produced by the first key (K1). It seems that small double concentration property will always occur no matter which key is used, while the occurrence of large double concentration only shows up conditionally.

Since this phenomenon did not occur in the previous experiments, to make sure it is reproducible, we repeated this experiment several times and tried several different keys. We found that the occurrence of such a phenomenon is almost unpredictable. It is so capricious that it may show up in one experiment while mysteriously disappear in the next one. Additionally, it might even expose itself in the middle of the experiment The most typical example can be shown in Figure 6. For the first few hundreds of samples, there is no clue that the second concentration will occur.

This finding challenges the approach of using average of all samples. When observing a single large concentration, no one knows whether the seen one is the upper large concentration or the lower large concentration or even the com-

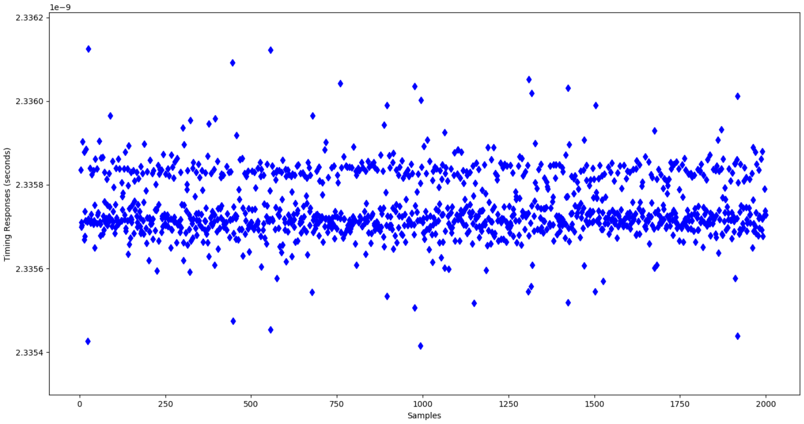

Figure 4: M1-K2 Timing Measurement: Upper Large Concentration

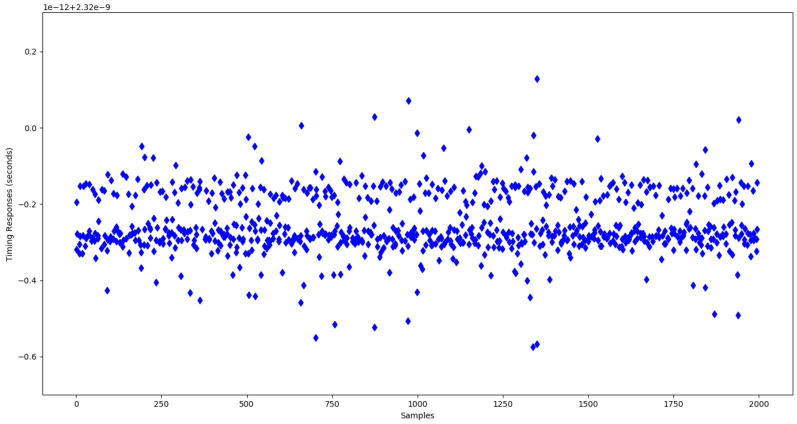

Figure 5: M1-K2 Timing Measurement: Lower Large Concentration

bination of both. This means measurement error can be dozens of milliseconds depending on the guess of whether it is the upper one or the lower one.

1.3 Convergence of Average

In addition to all issues of assuming $t = T(km)$ to be a real number aforementioned, another one is the number of samples required to achieve a specific accuracy. If the average of all samples was chosen to characterize the timing

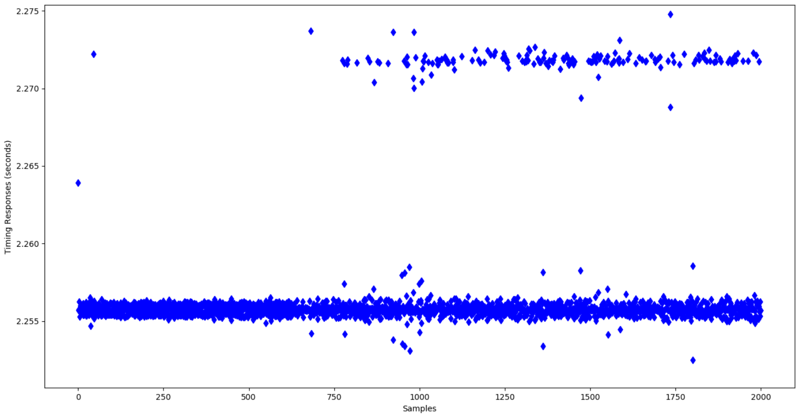

Figure 6: M1-K3 Timing Measurement

measurements, it should at least converge to microsecond precision since the timing difference between the two clusters is about several hundred microseconds as we have shown in the first experiment. To know how many samples are needed, we signed a message $10,000$ times and calculate the temporary average. That is, we calculated $a_i = \frac{s_i}{i+1}$, where $s_i = \sum_{k=0}^{k=i} t_k$ is the sum of all timing responses t_k up to i_{th} responses. Figure 7 shows the value of current average a_k when k iterates from 0 to 9999. The blue line represents a_k while the yellow line serves as a reference of the final value. It can be seen that the blue line was still fluctuating after $4,000$ samples because of several outliers. As long as these outliers exist, even $10,000$ samples are not enough to achieve 5 digit precision. This makes the average approach impractical since a signing usually take several seconds.

1.4 Dominance of Keys

Another interesting property is the overwhelming influence of keys. In the first experiment, it can be easily seen that timing responses generated by one key differ from timing responses generated by another dramatically. The offset in figure 1 is 5.279 while the offset in figure 3 is 2.320. Contrasting to the timing difference when signing different files, the timing difference caused by using different key is gigantic. It seems that it is the key that dominates the timing response. To examine this hypothesis, we signed a single message using 500 randomly generated keys. Figure 8 shows the resulting timing responses. They are uniformly distributed over $(2, 8)$, which confirms keys always dominate timing responses.

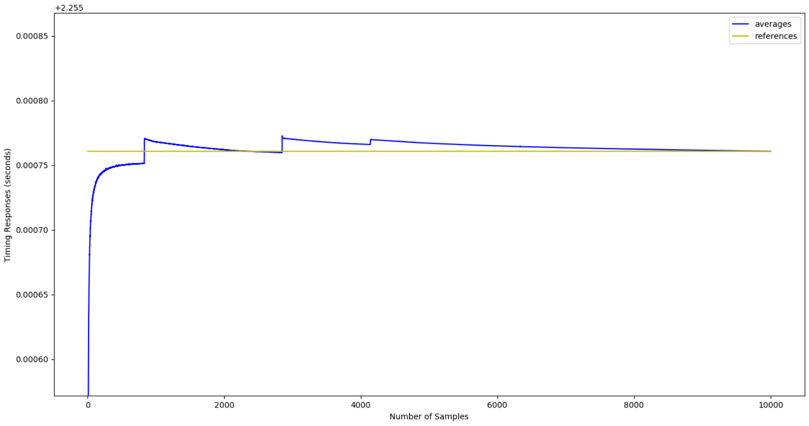

Figure 7: Convergence of Average: Original Data

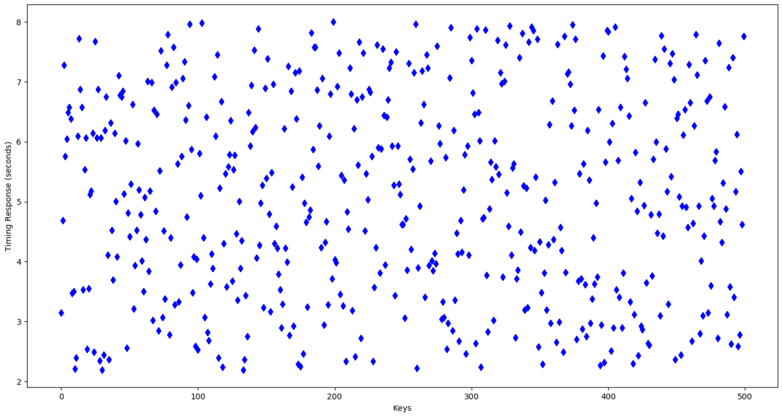

Figure 8: Timing Measurements of a 500-sized Key Set

1.5 Bootstrapping Influence

This property was discovered by an accident in an experiments. Once we were measuring timing responses for other purposes, the power supply of the HSM was cut off by some other students. Since the HSM has to set its master key whenever it is restarted, and private keys written into the HSM when logging in with one master key cannot be used when logging in with another, we have to delete all keys in it and rewrite all of them into the HSM. After all these were done, we started to measure the timing responses again. Based on our previous experiment, we expected that the timing responses of key (K1) should be about 5.27. However, the offset of timing responses is now about 2.25 while the timing difference between two concentrations stays similar at around 0.00015 (see Figure 9). We tested other keys and found that all of the keys now result in different timing offset but same difference in two concentrations. Fortunately, all other properties remain unchanged. We still saw double-concentration property, the timing difference in two concentrations stays similar, key was still dominating the timing responses, and we still have to drop some data points in order for the average to converge.

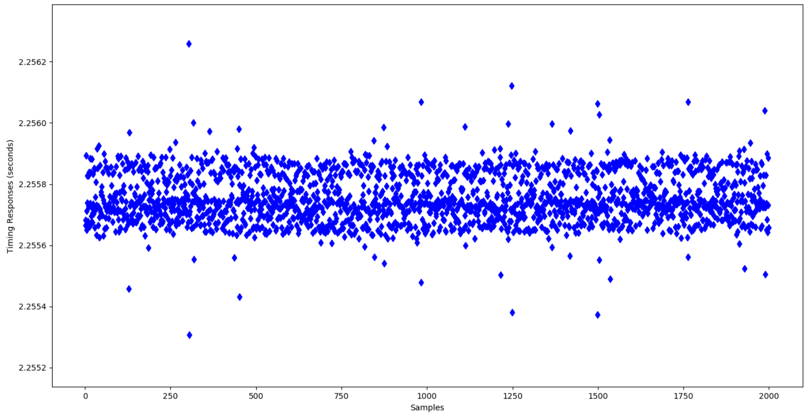

Figure 9: Re-bootstrapped M1-K1 Timing Measurement

This experiment brought us both bad news and good news. It told us measurements taken under different master keys cannot be taken into consideration simultaneously. However, it confirms our proposition that the offset is independent of the key message pair.

2 Machine Learning Approach

2.1 Regression

In our experiments, we tried hundreds of different hyper-parameters, including changing the number of hidden layers from 1 to 20, the number of neurons in each layer from 40 to 200, modifying different methods of weight initialization and different combination with other influencing factors.

However, even with our current best performance, the average error reached 1.4 seconds in the process of actually applying the model for prediction. This result is better than randomly guess the answer (average error being 1.95s) and only a little better than the strategy of choosing the average time as the prediction.

In figure 10, we could see that the basis of each layer continues changing to fit the excepted value, but the weight changes very little after the first 4,000 steps. That means the learning ability become pretty low in the second half of the training. And the error shows that the model is not very strong, which means the key information may not has a strong relation with the running time.

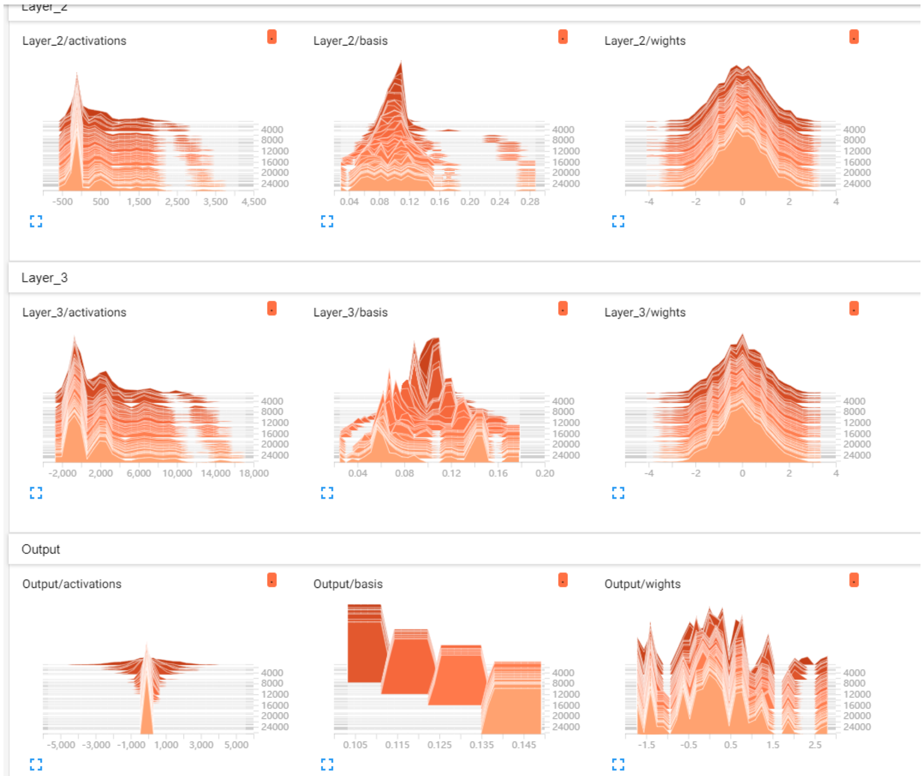

Figure 10: Loss function value curve(red: N, blue: d)

2.2 Classification

In the classification experiment, we tried both random forest and neural network methods and divided the dataset into 7 different categories.

Figure 11 shows the training process of the neural network. We could see that the loss function converges after 10,000 steps, but the accuracy is only about 20.15% which is only a little better than randomly choose. Meanwhile, the random forest model did not perform well too, it overfitted after running about 100 steps and the accuracy on the validation set is 18.3%. This answer demonstrates the result that our current design input vector doesn't have a strong relation with the output label, which is the same as the regression model.

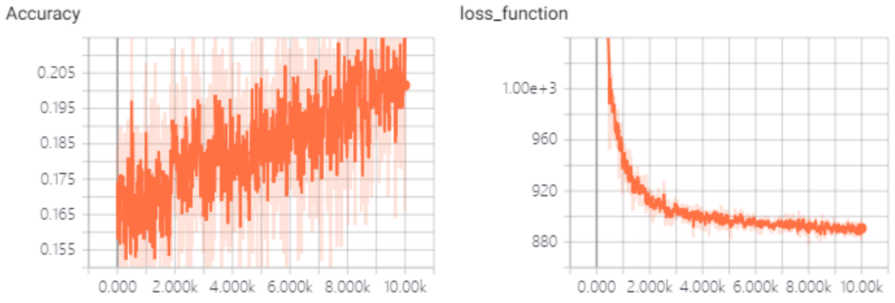

Figure 11: Loss function value and accuracy curve

From all those models we built and trained, it not show a strong relation between only the key and the signing time. The potential attack need to consider more factors(hardware setting and implementation).

2.3 Conclusion

In the first experiment, we found that timing responses concentrate on two different values. We also found that this phenomenon sometimes happens in both small and large scales. The second experiment showed us it is necessary to drop anomaly data points in order to make its average converge to a specific precision within a reasonable number of samples. In the third experiment, we discovered it is the key that dominates the timing responses. In the fourth experiment, we unearthed the truth that the large timing variation caused by switching keys may be closely related to the HSM bootstrapping process. In traditional timing attacks, researcher often relies on the assumption that timing response of a given key message pair is just a constant. With these experiments, we presented that this HSM does not behave this way and there are a lot of difficulties applying those existing timing attacks on it.