CS4150: TEMPLATE ATTACK

Bas van't Spijker
1497944

Yuhang Tian
5219728

May 27, 2021

1 Model Description

The template attack is to build the probabilistic model for the input features and output classes. To establish this model, three things should be figured out - what are the features, what are the labels, and how to construct the distributive model.

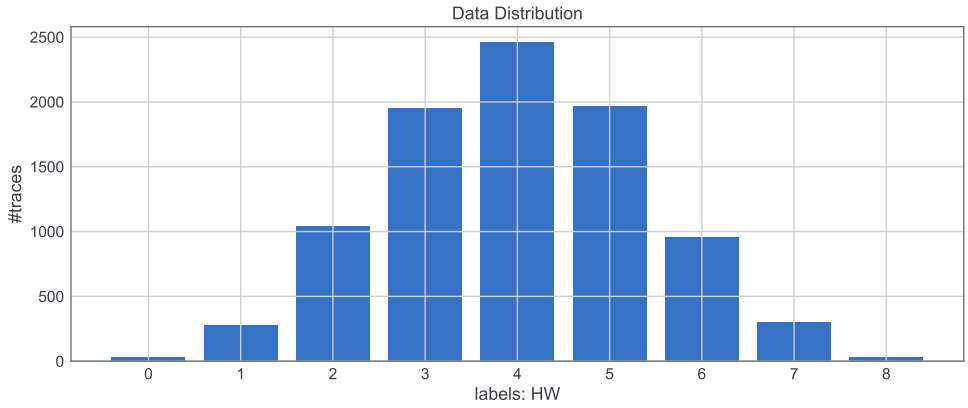

Figure 1: Hamming Distance Distribution

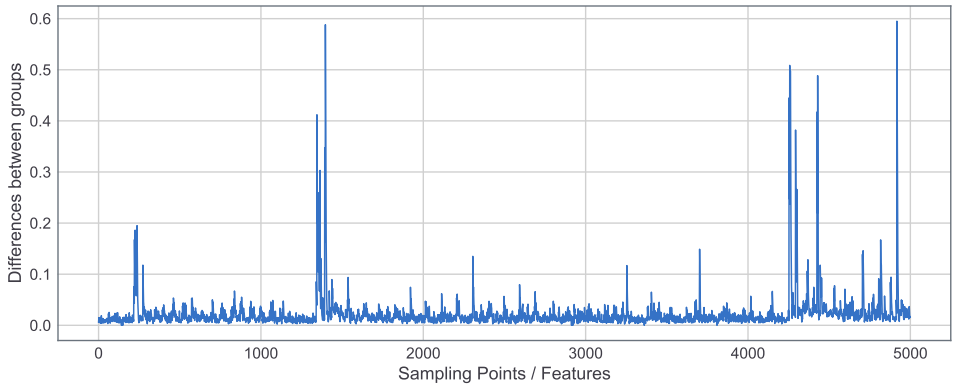

Figure 2: Points that generate the most significant differences

The input features or the points of interest are extracted from the power leakage of traces which are measured by an oscilloscope. Similarly, we still used Hamming distance to build the relationship between the key after being xor with plaintext and their power leakage and such relation would be reversely used for finding the key. In terms of Hamming distance of 8-bit digits, there are 8 different classes ranging from 0 to 8. We selected the first 9000 traces and computed the Hamming distance for each of them. The result is shown in fig. 1. It shows

that the more center the value is, the more frequently it occurs because it is less difficult to combine a value in the middle for the hamming weight. This would cause a problem when training. If the number of training traces is insufficient, some classes may contain fewer points than the features or even miss from that figure, which will make the covariance matrix become singular. However, this dilemma can be overcome by using pooled template attack that will be demonstrated later on.

In the data set, for each trace, it records 5000 points. If directly training this raw data for the model - 5000 points as the input features, it is time-consuming. In order to lower the feature dimension, we carried out feature selection. Since it assumes there is a relation between hamming distance and trace power leakage, we subtracted the means between different hamming classes and summed them up. The accumulative subtraction reveals which points are mainly for generating the differences. The result is shown in fig. 2. The higher the spike, the more important the feature is. The feature owning a higher spike is more deterministic and powerful for the label decision.

It should be noticed that the points are captured by the oscilloscope with a very high sampling rate - the oscilloscope commonly owns a higher sampling rate than the original signal) - the neighbors of a point may still stagnate at that point. We set a window to only select a point in a local region and the result has been shown in fig. 3 where the red points denote the points of interest.

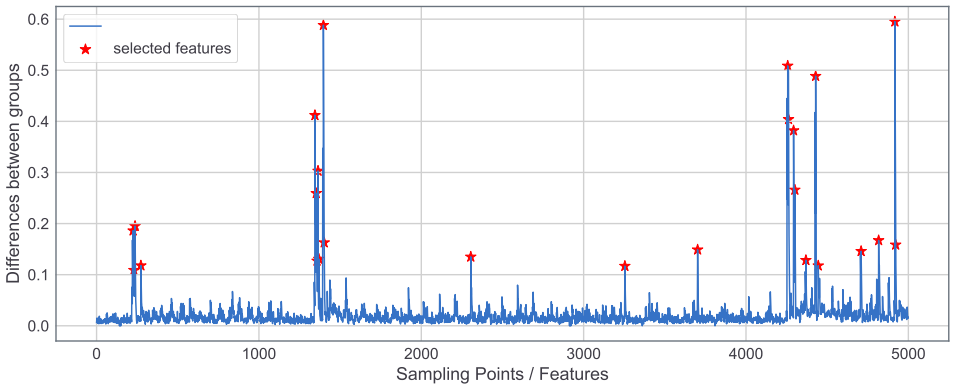

Figure 3: Selected Features

To build up the probabilistic model of hamming classes and points of interest, the two-variate Gaussian model will be applied which is like eq. 1. Besides, we applied Bayes in eq. 2 to find the poster-prior probability.

$$p(X = x|Y = y) = \frac{1}{\sqrt{(2\pi)^D|\Sigma_y|}} e^{-\frac{1}{2}(x-\overline{x_y})^T \Sigma_y^{-1}(x-\overline{x_y})} \tag{1}$$

$$P(Y = y|X = x) = \frac{p(Y = y)p(X = x|Y = y)}{p(X = x)} \tag{2}$$

2 Template Attack

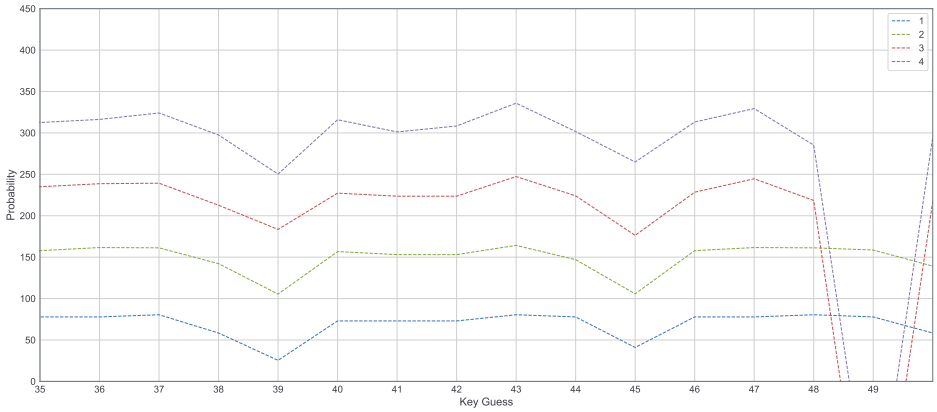

Figure 4: Probabilistic Curve: an example of sub-key 0

To carry out template attack, the number of training traces is set to 9000, the number of points of interest is set to 25. In this setting, it uses 11 testing traces to find the correct key in the template model.

In fig. 4, it shows the changing of post-prior probability of subkey_0. When the probabilistic model observes a new point, it will fit the original model and calculate the current probability of all possible values for that key from 0 to 255. When it sees one more, it will accumulate the calculated log-likelihood and adjust the current curve. The correct value's probability grows larger when the feeding traces are ample and becomes the predominant one.

3 Pooled Template Attack

As mentioned, the main drawback of a template attack is when the training number of traces is limited, the only way to avoid a singular matrix is to reduce the number of points of interest. However, this reduction makes the sampling points not fully used. For instance, when the above setting of template attack is changed in which the number of training decreases to 5000, 25 features lead the matrix to be singular.

The disadvantage of template attack can be improved by using pooled template attack. The difference is shown in eq. 3 where the covariance matrix is replaced by the mean of the original one - the average of 9 classes. However, in pooled template attack, the substitution makes it become less powerful since it loses the information of noise for each specific class which is now estimated less precisely. In order to compensate for this, it may require more testing traces to figure out the correct key.

$$p(X = x|Y = y) = \frac{1}{\sqrt{(2\pi)^D|\Sigma|}} e^{-\frac{1}{2}(x-\overline{x_y})^T\Sigma^{-1}(x-\overline{x_y})} \tag{3}$$

If the same setting is applied to template attack, it requires 2 more traces to find the whole key. The performance is still not bad. In addition, it can be trained by fewer training traces, for instance, 5000. In this condition, it requires 5 more traces to find the whole key, but the template cannot use 5000 training traces for 25 selected features. In conclusion, if the training traces are not enough, a pooled template attack can be a useful alternative method to break the key.