Cartography

When acquiring EM signals, the location of the EM probe above the device under test has great importance to the quality of the captured signals. Two popular metrics to evaluate the quality of the captured signals are the Signal-to-Noise Ratio (SNR) and Test Vector Leakage Assessment (TVLA). Typically, the leakage analysis process is executed within the context of extracting a cryptographic key from a device [1]. In our settings we are not interested in extracting a key, but rather in monitoring the executed code on the device, requiring us to use a different metric to evaluate the quality of the captured signals. Our leakage profiling process is composed of two stages: identifying the EM bandwidth where the leakage occurs, and then identifying the location that provides the best leakage.

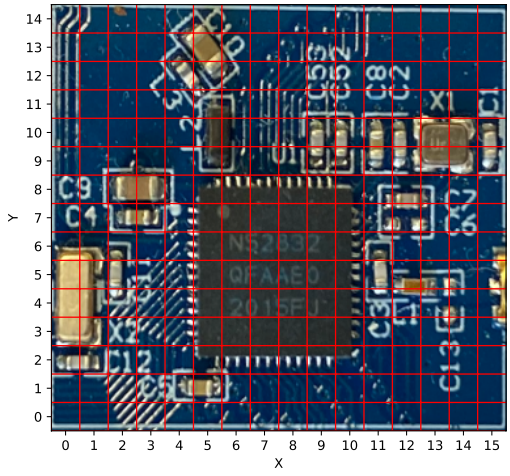
1) Identifying Electromagnetic Bandwidth: When using spectrum analyzer, there are two key parameters, the center frequency which refers to the specific frequency which the analyzer is configured to capture the spectral content of the signal, and the capture bandwidth (or span) which refers to the range of frequencies that the analyzer will capture. For example, an analyzer that is configured to a center frequency of 10Mhz and a span of 500 kHz, will capture the signals in the range of 9.75Mhz to 10.25Mhz. Over the course of this study, it became clear that the EM leakage does not occur in a specific frequency, but rather in a range of frequencies. This is due to the fact that the EM leakage is a result of the switching of the transistors in the device, and the switching frequency is not constant. Reducing the capture bandwidth to the minimum required, will reduce the amount of data that needs to be processed, and will reduce the amount of noise in the captured signals. Identifying the EM bandwidth where the leakage occurs included several steps. First we created a program with two phases, the first phase induced heavy computational loads on the processor for a limited amount of time, followed by a phase during which the loads were minimized for equivalent period. For example, the computational intensive phase can be simulated by applications like AES encryption, and the second phase can be simulated by executing NOP instructions consecutively. Running this two-phased program on the device under test will cause the captured EM signals to contain two distinct patterns, one for the computationally-intensive phase, and one for the idle phase, patterns that we will further utilize to minimize the effective bandwidth. The key idea around creating the two phase program is that as long as the capture bandwidth is in the correct range, the two phases will be distinguishable.

To minimize the capture bandwidth, we first configure the spectrum analyzer with an arbitrary center frequency and a wide capture bandwidth. We then execute the two-phased program that was explained previously on the device under test and capture the EM signals. The next step is to iteratively reduce the acquisition bandwidth. For each iteration, we reduce the span by a reduction factor which initially equals to $\frac{1}{2}$, then capture the signals again. After each capture we calculate the mean of each phase (e.g. intense and idle) in our signal. If the difference between the means is greater than a threshold, meaning the phases are distinct, we continue to the next iteration. Otherwise, we conclude that the range that was filtered out in the previous iteration includes the range of interest. In that case, we restore the span to include it, then try to reduce it with a lower reduction factor. We perform this reduction process twice, each time anchoring one of the edges of the span. Following the previous example, with a center frequency of 10Mhz and a span ranging between 9.75Mhz to 10.25Mhz, the first reduction will result a center frequency of 9.875MHz and a span ranging between 9.75Mhz to 10Mhz (assuming lower limit anchor). We stop the process when any further reduction of the span will result in indistinguishable patterns. A high level pseudocode of the process is presented in Algorithm 1.

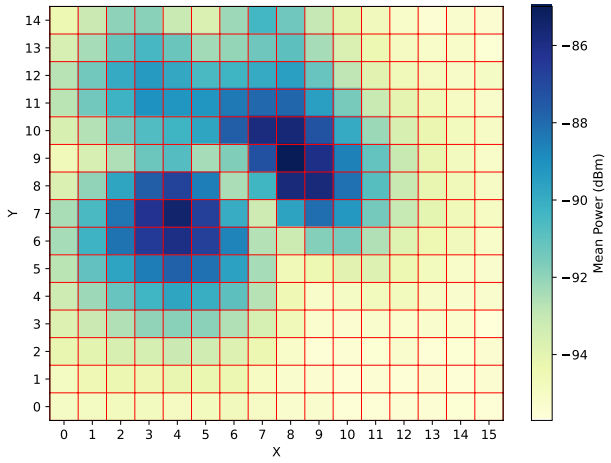
Algorithm 1

$Center\_Frequency \leftarrow Arbitrary\_Frequency$
$Span \leftarrow Max\_Span$
$Reduction\_Factor \leftarrow 0.5$
while $Reduction\_Factor < 0.95$ **do**
  $Signal \leftarrow Capture\_Signal()$
  $Intense_{mean} \leftarrow Mean(Signal_{intense})$
  $Idle_{mean} \leftarrow Mean(Signal_{idle})$
  if $Intense_{mean} - Idle_{mean} > threshold$ **then**
    $Span, Center\_Frequency \leftarrow Reduce\_Span()$
  else
    $Span, Center\_Frequency \leftarrow Restore\_Span()$
    $Reduction\_Factor \leftarrow Reduction\_Factor + 0.05$
  end if
end while

2) Mapping the Device: After identifying the EM bandwidth where the leakage occurs, we need to identify the device's "sweet spot", i.e. the location that provides the best leakage. To do so, we defined our areas of interest and divided our device under test into a grid of n by m cells, the area we focused on is presented in fig. 1a and includes the SoC and its surroundings. To measure the quality of the signal emitted in each cell, we executed the two-phased program again and captured the frequency spectrum (rather than the time domain signals). Next, we calculated the mean power at the captured frequency band. The mean power of the frequency band is a good indicator since the executed instruction has direct influence on the switching of the transistors, and therefore on the power consumption, influence that is also reflected in the EM leakage. We repeated this process 30 times for each cell and averaged the results. The experiment was done using a Secure-IC Analyzr XYZ table with 3 Newport SMC100 motion steppers. The results are presented as a heatmap in fig. 1b which is aligned with fig. 1a. The heatmap represents the mean power in each cell, the with darker cells indicating stronger leakage. The heatmap shows two areas ($(x, y) = (4, 7)$ and $(x, y) = (8, 9)$) where the leakage is notably stronger than the rest of the grid. Surprisingly, these areas are located in



(a) The area of search of the device under test



(b) Heatmap of the mean power values after full scan

Fig. 1: The scanned area its respective power values

the surroundings of the SoC, around capacitors and copper traces rather on top of the SoC, possibly because of the SoC packaging that is effectively reducing the EM leakage.

To confirm the results we made another visual analysis comparing the time domain signals of high power cells against low power cells. The visual inspection confirmed the results, as cells with high power showed the two distinct patterns clearly, while cells with low power showed no patterns at all. Finally, to verify that the identified bandwidth and sweet-spots that were found earlier are not specific to one program, we repeated the process again with a different program.

This time, the intensive phase was simulated by executing matrix multiplication instead of the AES encryption algorithm. The results were similar: the minimized bandwidth spread around the same frequencies, and the Pearson correlation coefficient between the two heatmaps was 0.95 with a p-value of 2.09^{-127}, making it practically impossible to observe such results by chance. Since the two tested programs are different in nature, the similar bandwidth and similar sweet spots indicates that they are not program specific, and can be used regardless of the executed program.

References

[1] Josef Danial, Debayan Das, Santosh Ghosh, Arijit Raychowdhury, and Shreyas Sen. SCNIFFER: low-cost, automated, efficient electromagnetic side-channel sniffing. *IEEE Access*, 8:173414–173427, 2020. doi: 10.1109/ACCESS.2020.3025022. URL https://doi.org/10.1109/ACCESS.2020.3025022.