Improvements in MOLES Trojan Analysis and FPGA Implementation

CEAS-EECS | Senior Design 2020

University of CINCINNATI

XILINX VHDL python

Purpose

The purpose of this project is to improve the analysis of the MOLES trojan circuit. Focus will be on improving the overall efficiency the actual attack, with potential improvements in a FPGA implementation of the circuit.

Hypotheses

The efficiency of a MOLES-based side channel attack can be improved with machine learning. This would mean that a MOLES-based SCA could be achieved with less data and/or less processing time to determine the secret key.

Additionally, using the fact that high capacitance can be implemented in a FPGA via high fanout logic switched at high clock speeds, it is possible to simulate the MOLES capacitance with a long series of LUTs.

Background Information

The separation of design and fabrication of ICs has led to vulnerabilities originating in untrusted circuit foundries which could implant malicious hardware Trojans into a genuine design. One such Trojan circuit proposed in the literature is the 'Malicious Off-chip Leakage Enabled by Side-channel's (MOLES). This small trojan is designed to leak secret multi-bit information off-chip through power side-channels. (See Figure 1 below) This leakage occurs at low enough levels as to appear to be part of the noise generated by the power consumption of the host IC in order to evade detection

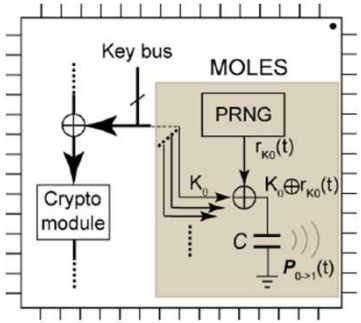

Figure 1: Diagram of MOLES Trojan

However, extracting the leaked data difficult as at least 10000 of data sets are required. These ae very time consuming to collect, organize and analyze.

Challenges

Machine Learning

- As we began to better understand the nature of the MOLES trojan, scope creep became a serious issue. Because of the high level of entropy of the system, significant pre-processing would be required to have the model learn associations between keys and the signal. This coupled with the fact that the key generated signal is intended to have a pseudo-random distribution for the length of its period meant that our initial plan to have the model learn to make predictions based on.
- The modeling process posed itself as a bottleneck for the analysis. To determine the secret key via MOLES trojan, the output power trace must be analyzed, and the composite LFSR/secret key pattern isolated. However, extracting this signal requires a significant amount of data, which we did not have as there are no existing data sets (that we're aware of) for our problem space. Therefore we needed to generate our own data.

Due to the issues surrounding the implementation of machine learning, we rescoped to focus on implementing internal FPGA capacitance.

FPGA Implementation

- Due to the move to remote classes resulting from Covid-19, we lost access the oscilloscope we had been using. We managed to borrow an oscilloscope from UC, but it was older. This resulted in some difficulties for us. First was the decreased sampling rate, which reduced measurement precision. Some other minor issues were a USB 1.0 port type for waveform export and that we did not have software to run interact with the oscilloscope from the computer. We ended up writing python scripts for this purpose.
- While working on the internal FPGA capacitance, we had difficulties getting the initial current draw large enough to measure. We determined that we did not have the clock at a high enough frequency. – 5MHz s 100MHz. Once the clock rate was increased, the current draw was significant enough to measure.
- FPGA development tools are designed to optimize code, which has been a problem for us deliberately creating unconnected circuitry that serves no obvious purpose to the tool. More work needs to be done to reliably force the tool to allow the generating of internal capacitance.

FPGA Implementation

Implementing the MOLES trojan on a FPGA is simple, barring the implementation of the capacitors. These capacitors are what cause the power draw that is analyzed to recover the secret key.

External Capacitance: Proof of Concept

In our proof of concept phase we used external capacitors attached to the Zed-Board to test our ability to recover keys through the power measurements. Having a higher capacitor value draws more current and therefore makes the key values easier to recover. Because of this we initially started with very large capacitors 50-10uF. However, having very large capacitors also means that they take a very long time to charge. During tests with a 5MHz clock, large capacitors did not have enough time to charge fully inside of one clock cycle, resulting in bad data.

After experimentation, we realized that the best results came when there were no external capacitors connected to the board while we kept driving the output pins. This is because the capacitance of the pins themselves was enough to give a strong enough signal to recover the keys.

Internal Capacitance

To simulate capacitance in the FPGA, LUTs strung together to create a long line of logic, resulting in fanout. With enough LUTs in series being driven from one to zero and back again will create notable current draw using a power supply. This means that some form of capacitance has been created or increased in the design while it is active. The larger the fanout or the faster the switching of logic i.e. clock speed, will all increase capacitance

This design should be act similarly to the actual circuit, with the output of the MOLE trojan driving the LUT design and should create a noticeable pattern in the power data for power analysis. Most of the current work for the VHDL portion of the project has centered around how many LUTs are necessary and how fast the clock rate needs to be to get a good periodic signal in the power data.

Team Members

Linden Peterson	Heiko Stowasser	Adam Pendergrass	Thomas Kissel	Dr. John Emmert
Computer Engineering	Computer Engineering	Electrical Engineering	Computer Engineering	Technical Advisor

Conclusion

Due to unforeseen complications based on new revelations we determined that machine learning would not serve as an appropriate method of analysis as we initially hypothesized.

The implementation of the MOLES trojan on FPGA was initially geared towards obtaining additional data sets to use for improving the data analysis portion. However, once we realized the extent of the issues involved in implementing machine learning withing our time frame, we decided to focus on the FPGA implementation due to the broader effect on the hardware security community. Implementing internal capacitance would allow the implementation of other trojans on an FPGA, thereby expanding the ability of other researchers to test them in a real, unsimulated environment. This would also decrease the financial and time costs of performing hardware tests in comparison to creating a custom ASIC for each testing phase.

For the internal capacitance, LUTs were used to simulate capacitors inside the FPGA. An overall power consumption increase is noticeable with the LUTs present – which is what the MOLES trojan requires. We encountered difficulties in key extraction because our analysis model expects behavior of a capacitor. This appears as a short current spike at the beginning of the clock cycle when the capacitor charges. Our model looks specifically at the first 10-20% of the clock cycle to detect this spike, ignoring noise in the rest of the clock cycle. Using LUTs we have not observed the same spiking behavior. More research is required into how the LUTs generate current in order to create a more effective model for key extraction.

Future Work

Research into the internal capacitance implementation aspect of this project will be continued. Depending on future success, there may be a paper.

References

- Anderson, Jason H., and Farid N. Najm. "Interconnect capacitance estimation for FPGAs." ASP-DAC 2004: Asia and South Pacific Design Automation Conference 2004 (IEEE Cat. No. 04EX753). IEEE, 2004.
- Lin, Lang, Wayne Burleson, and Christof Paar. "MOLES: malicious off-chip leakage enabled by side-channels." 2009 IEEE/ACM International Conference on Computer-Aided Design-Digest of Technical Papers. IEEE, 2009.