Hardware Trojan: AES-T2200

Version	Author	Date	Comments
1.0	Rahul Vittal	4/20/2017	Initial Version
Fill in details for future versions			

- **Trojan Description:** This trojan is derived from AES-T1300. AES-T1300 had a flaw when it came to synthesizing it. The synthesis tool would remove the trojan circuitry as it didn't have a primary output. As a workaround, we have synthesized the trojan circuitry separately and fused it with the regular AES circuit (aes_synth_opt.v). We also created another workaround solution by adding a dummy output to the trojan circuitry before synthesizing it(aes_synth.v). This prevented the synthesis tool from removing the trojan circuitry.
  The idea of this trojan is to artificially introduce leaking intermediate states in the key schedule that depend on known input bits and key bits, but that naturally would not occur during regular processing of the cipher. The Trojan leaks one byte of the AES round key for each round of the key schedule. The leakage circuit (LC) is a 16-bit shift register and loaded it with an initial alternating sequence of zeros and ones. The shift register is only enabled in case the input to the leakage circuit is one, which results in an additional dynamic power consumption [1].

- **How to Trigger:** Make state == 128'h00112233_44556677_8899aabb_ccddeeff
- **Trojan Taxonomy:**

  Insertion phase: Design
  Abstraction level: Gate-level Netlist
  Activation mechanism: Triggered Externally
  Effects: Leak Information
  Location: Processor
  Physical characteristics: Functional

- **Attack model:** The attack will be performed by either an untrusted 3PIP provider or a rogue employee at the system integration level.

Running the benchmark: This needs you to have access to Synopsys 90nm SAED library and Synopsys VCS

1. *export SAED90nm_PATH=<Path to synopsys SAED 90nm Verilog file>/saed90nm.v*
2. *cd src/TjIn*
3. Execute one the run scripts : Execute "*source run_tj*" on the command line to run the testbench and the trojan vcs(needs Synopsys VCS license loaded). To see the behavior of trojan free circuit execute "*source run_tjfree*"

References : [1] L. Lin, M. Kasper, T. Güneysu, C. Paar and W. Burleson, "Trojan Side-Channels: Lightweight Hardware Trojans through Side-Channel Engineering," 11th International Workshop Cryptographic Hardware and Embedded Systems (CHES), pp.382-395, 2009.

Please send your concerns/questions to Administrator at admin@trust-hub.org