Survey on Hardware Trojan Detection Techniques

[1]Mrs G K Sandhia

ABSTRACT*— Hardware Trojan (HT) is a malevolent alteration of the circuitry of an IC. These are harmful to the security of the systems which are built on such malicious integrated circuits. Owing to the trend of globalization in the Integrated Circuit Industry, companies often outsource the design and fabrication of ICs to third-party vendors. The consequence of this trend is the potential existence of unlawful stealthily inserted Hardware Trojans which has resulted in a great security concern, and demands identification and detection of vulnerable HTs possible in different components of machine such as memory/cache system. The several types of intentional modification of the circuits result in the outcome of adversarial effects on the computer architecture. In this paper we discuss the various problems that are faced when detecting such Trojans and the solutions that have been created in order to counter such obstacles as well as their convenience. The methods that have been surveyed are applied at different stages of production of the ICs. The techniques used together with their limitations have also been surveyed.*

Keywords *– Hardware Trojans, Integrated Circuits, Security.*

I. INTRODUCTION

Owing to the trend of globalization in the Integrated Circuit Industry, companies often outsource the design and fabrication of the integrated circuits to third-party vendors. This trend continues to be popular and is frequently employed. However the lack of company supervision results the ICs being vulnerable to the tampering of the vendors to whom the task of fabrication has been outsourced. ICs are involved in almost without exception in every electronic circuit board, as well as embedded systems and numerous electronic projects.

The pervasive nature of the IC have made them indispensable in our daily lives. This very pervasiveness proves to be dangerous when we consider the ramifications of usage of unsafe ICs such as when introduced in high criticality systems such as in circuits that are used in the purpose of aerospace, health, military or finance. There are multiple security threats that compromised ICs result in such as use of subpar components, counterfeiting, and malicious alterations of the electric circuits. These Trojans can be inserted to introduced to achieve different objectives such as reducing the system performance, result in denial of service and leakage of sensitive information.

The Trojan hardware has two components mainly the trigger and the payload. The former is the circuit which is capable of activating the Trojan under specific conditions to enable it to carry out its function. The latter is the circuit itself which carries out the malevolent tasks that have been assigned to it. These circuits can be installed into the ICs at different stages of the manufacturing process. Due to the high level of threat that these Trojans pose the research community is investing significant efforts into conceiving ideas to rectify or detect them.

[1] *AP, SRMIST, Nandini Banerjee, Student, SRMIST, KTR*

Received: 21 Dec 2019 | Revised: 18 Jan 2020 | Accepted: 05 Feb 2020

The Hardware Trojan detection comprises essentially in creating techniques that can detect the alterations of malicious nature in the circuits. These techniques target various types of Hardware Trojans and find solutions to address the issues caused by them particularly. The methodologies can in turn be applied to search for these Trojans at different stages of the manufacturing process such as at the design phase, presilicon phase, postsilicon phase, fabrication phase, post-fabrication phase or even at runtime phase.

In this paper we have surveyed the different techniques that have emerged in identifying such Hardware Trojans including their competence and their drawbacks and also the specific problem that they aim to target. The Section II summarizes the different different techniques employed to detect the Trojans based upon the stages at which they are used into the architecture.

II. HARDWARE TROJAN DETECTION TECHNIQUES

Hardware Trojan detection is crucial that it be done at an early stage. The identification of such Trojans early leads to their isolation to ensure that they are not used for commercial purposes. The detection techniques vary depending upon the inputs used by them as well as the different phases at which they are employed.

A. *Design Phase*

Chances of HTs getting inserted at the design phase are significantly greater than at the manufacturing phase. The detection techniques based on identification of unused or nearly-unused circuits were static, inaccurate and unable to detect all types of HTs. HTs are classified into parasitic and bug based. Parasite based Trojans exist alongside the actual circuit and do not cause any loss of its original functionalities. Bug based Trojans cause changes in the circuits that results in the circuit losing its original functionalities partly. VeriTrust employed by Zhang et al[8] identifies dedicated trigger inputs, i.e., inputs used in the condition under which the HTs are activated; the parasite-based HTs are detected. The VeriTrust technique is used to uncover parasite-based HTs. In an RTL VeriTrust identifies Hardware Trojans by searching for the inputs that are redundant. It does this by changing all dormant entries during the course of testing to don’t care values. Essentially VeriTrust can be regarded as a method that employs the concept of “unused input identification”. The comprehensive framework of VeriTrust consists of two parts, namely, the tracer and the checker. The strength that VeriTrust has over its contemporaries is that it is unbothered by the style of implementation of the Hardware Trojans and hence simple HT modifications cannot inhibit VeriTrust. However, the limitation is VeriTrust is incapable of detecting those Trojans whose trigger is on all the time. VeriTrust detects Trojans with functional-level malevolent behavior. VeriTrust is unable to detect bug-based HTs but threat due to them is very small.

Another design phase detection technique takes into account that detection of Trojans by manual inspection of the design layout or the hardware code is a strenuous activity. Identification of Trojan in a circuit without a copy for checking for equality is a difficult task. The solution proposed by Reece and Robinson[7] in this paper is to contrast two design of untrustworthy nature which are alike but not of identical functionality to enable validating each other. Trojans are unlikely to be activated by their triggers during normal testing/usage. Hence if one Trojan is less likely to be activated by coincidence then the chance of two Trojans to be triggered in the same manner for different payloads is even lesser. The method used for HT detection is called design comparison. To effectively note the similarity between two designs the processing steps used are the wrapping of two designs to match their functionality and unrolling the internal state components. Outputs from the two circuits are compared using the Boolean satisfiability (SAT) solver which identifies the probable inputs that cause the two circuits to produce distinguishable outputs. The strength of this method is that it is very effective when applied to multiple Trojan benchmarks. It is efficient with respect to both the detectability and the quickness of the test. Also it is a low cost solution. However, its drawback is that the tools used such as SMV greatly restricts the circuits to be tested and is not efficient with respect to circuits of large size. The Cadence RTL compiler optimizes space and time required even when not instructed to do so but this makes detection difficult. The unroll depth which is defined allows for Trojans with extremely lengthy time for activation to remain concealed by surpassing the test window.

Trojans are of two types: either they are always active or they need to be triggered. The triggering event is a rare condition that makes detection of Trojans hard by trying to estimate the exact time of triggering. Shen at al[11] have proposed a design phase detection method called LMDet which derives from the principle of recognizing the naturalness of a circuit. Clean circuits are associated with being natural and circuits infested with Trojans are seen as unnatural or unusual. Being natural derives from the NLP field indicating regularness and repetition whereas unnatural is indicated as being an outlier. The quality of naturalness with respect to circuits is seen as gate sequences that are frequently encountered in the training data. Any circuit that has gate sequences with unusual gate sequences can be deemed as a malevolently altered circuit such as bugs or Trojans. Integrated Circuit Designers should be made aware of such alterations. A gate sequence base statistical model is developed in order to locate the unnatural circuits. This is done by using a circuit graph that has been derived from a design at gate level. The statistical model is then trained on the data provided to it. The sequences of gate in the circuit are examined in accordance to their probability in the model built. Any gate sequences that are seen of low probability are marked as suspicious. LMDet is advantageous as it does not need a reference of a golden chip, its execution time is brief and it is applicable industrially as there are plenty of training sets available for the model to ensure effective statistical results. There is no need for functional testing because the model is also static. The effectualness of the method depends essentially on the quality and quantity of the training data provided. However, an assumption that this method works on is that low observability gates are associated with the payloads and low controllability with triggers. Trojans that do not satisfy this assumption are not detectable by this method such as the always active Trojans.

B. *Presilicon Phase*

The existing detection methods are unable to guarantee high efficiency and accuracy simultaneously. The method suggested by Chen et al[2] uses the principle of feature analysis but for Trojans where the HT trigger features are by structural analysis extracted and added to the database to make the process of detection scalable. The method proposed by the paper is a multilevel fast trustiness verification framework (ML-FASTrust) which is constructed on the principle of feature analysis. The strength of this method is that it has low time costs and scalability and the HT coverage. This method is capable of detecting implicitly triggered HTs which are Trojans whose trigger logic are strewn over many pipeline stages. However, its weakness is that sometimes it is hard to differentiate between the Trojans and the benign components multipliers in circuits. To ensure accurate detection in depth knowledge is needed but this requires huge storage space whereas less knowledge cannot detect the HTs precisely.

Another method to detect Trojans in the presilicon stage has been proposed by Haider et al[4]. It deals with the fact that contemporary HT detection methods such as VeriTrust and FANCI can be circumvented by Trojans because typically these techniques provide assured identification for a known set of Hardware Trojan benchmarks. On designing even a slightly disparate HT, it can bypass these detection techniques. Detection techniques against a bigger class of Trojans are required. This paper takes into account several advanced properties of deterministic HTs and not just targeting known Trojans. These properties define the stealth of the HTs thus help to gain an idea of the design principles that the attacker may use to design Trojans to circumvent the currently available HT detection mechanisms. The algorithm applied is called the HaTCh algorithm to detect HTs. This paper provides an impressive HT detection algorithm called Hardware Trojan Catcher (HaTCh). Its working is in two stages namely the learning phase and the tagging phase. The phase including the logic testing of the IP core to record untrustworthy transitions is the learning phase and the phase performing extra logic with respect to the IP core in order to enable tracking the untrustworthy transitions is the tagging phase. Unlike the other contemporary algorithms available the strength of this algorithm lies in the fact that it provides absolute coverage and provides security for Trojans of multiple classes, whether known or unknown. Nonetheless it still has the drawback that for pipelined and non-pipelined circuitries an overhead of 8:34% and 4:18% is seen respectively. In case of few benchmarks, there is considerably higher overhead than some because of the randomly taken pattern of inputs which do not cover some benchmarks.

*C. **Postsilicon Phase***

Increasing complexity of current ICs and dearth of controllability or observability regarding the it's internals, result in the contemporary tests being ineffective.

The methods used in reverse engineering are invasive, slow, damaging and costly. In this paper Nowroz et al[6] proposes a noninvasive method for detection of Trojans. The use of infrared waves emitted from behind the silicon die results in thermal characterization, which is utilized for obtaining detailed spatial power maps. 2-D principle component analysis (2DPCA) is used to obtain high dimensional thermal and power maps. Applying 2DPCA, a training data set for the supervised threshold approach is needed and none necessary for the unsupervised clustering approach. The strength of this method is that it's an easily scalable procedure. The PVs of gates is taken into account. The method is capable of detecting and locating Trojans of very small sizes as well as small power consumption efficiently. The method has tremendous potential however it does have a limitation since a more general framework needs to be developed that can be applied to a larger set of Trojans. Also this method needs to be extended for detection of pure leakage power. Also the impact of PVs needs to be decreased so as to reduce false positives.

D. Fabrication Phase

The currently available techniques search for HTs in inactive and extreme nets in circuits during the test phase. However the HT can be triggered in the function mode as well, maybe not intentionally however it is a fair enough reason for designers to look for in function mode as well. In this paper Zou et al [10] employ a method that looks for inactive and extreme nets as they're the best candidates for insertion of HTs in the test mode as well. The method is used at the fabrication process. The method finds inactive and extreme nets for a circuit-under-test (CUT) by calculating its switching probabilities as well as its state probabilities. The fast heuristic method reduces total states and the transition tables to speed up the process. This method is advantageous because it has low complexity, high precision, and tested on large circuits as well as popular benchmarks. It makes circumventing the method by attackers difficult as they run the risk of accidental triggering. There is need for future work to be done to compute joint multiple net transition probability.

Postsilicon HT detection uses the process of identifying HTs in fabricated chips, which use side-channel analysis for which procuring a golden chip is necessary. Obtaining such a chip is costly and it's an invasive process. Also process variations (PVs) are a major challenge. In the case of a golden chip-free solution, the gate profiling applied is over determined for the linear system and it incurs a huge cost for large circuits. The solution offered by Chen et al[3] for formulating linear systems for each chip is using sparse gate profiling. Then a Bayesian inference based method for the purpose of calibration of the process variation distribution for every chip is employed. To obtain a PV distribution by a maximum-*a-posteriori* (MAP) this solution puts together the previous knowledge and the necessary measurements for recovering the MAP. The underdetermined linear systems are solved. Detection of HTs is done by analysis of the solutions obtained. The Monte Carlo (MC) simulation is used to obtain prior distribution of the scaling factors. For calibrating the distribution of PV, embedded into the system are PV monitors. Using the simultaneous orthogonal matching pursuit (S-OMP) algorithm, the underdetermined linear systems are solved. The strength of the proposed method is that it has high HT detection rate with low overhead costs and detects HTs in fabricated chips. However, the method used is limited to ensure detection only during fabrication stage only and not during the other stages like design, testing, packaging and use stages.

E. Post Fabrication Phase

The detection of HTs is difficult because of the rare conditions under which they get triggered. The methods used for detection are characterized under two classes, namely logic testing and side channel analysis. However both have limitations. The former is unable to take into account PVs and noise. The latter approach incurs large overhead. In this paper

Zhou et al[9] proposes a method to improve the transition probability of nets by inserting 2-to-1 MUX as test points to and thus activate the HTs. Since the transition probability of one net affects another's in a fan-out cone, those which influence maximum nets with minimum transitional probability are the selected candidates. Increasing transition probabilities helps detect HTs. The theoretical analysis of transition probability is done for improvement. Application of weighted random pattern to determine probability distribution of input patterns. There is usage of selection algorithm to update the next list after insertion of MUXs and WSPs for each input. Also there is determination of threshold of transition probability. This method is advantageous as it is efficient, low cost and incurs a low hardware overhead. Since the proposed approach exploits the logical connection relationships for determining the insertion points therefore any other methods based on the insertion of circuits is compatible with this method. However, this method does have limitations as it encounters minor delay and area overhead on its application.

F. Runtime Phase

Most of the existing HT identification approaches enable detection at test-time. Test time methods fail for well located HTs which do not get triggered at the time of testing and thus the HTs may get deployed. Through this paper Bao et al[1] proposes a runtime detection scheme. It utilizes the link between temperature and power. The change in power consumption of the activated HT reflects in the integrated chip's thermal profile. The leakage power is also taken into account and Kalman Filter is applied. The layout and power information is derived by simulation, synthesis and placing the design in Cadence SimVision, RTL compiler and encounter tools. Using RC thermal model the link between temperature and power is illustrated. The advantage of using this method is that since it is a runtime detection method it can be used throughout the IC's lifetime. Runtime detections usually have large overheads however this method has low overhead. This method avails the already present thermal sensors in electronic systems. However, currently Trojans that do not significantly affect temperature before deployment cannot be detected for which prior test-time approaches need to be applied separately.

Since most HTs target digital circuits there is an assumption that HT should itself be comprised of digital logic which limits the competency of the existing detection methods. Also another limitation of the methods is that they cannot detect very small Trojan's whose effect can be masked by the PVs and noise. In this paper Hou et al[5] take into the account the existence of analog Trojans. It presents a detection method on-chip R2D2 which targets analog Trojans, which are activated by wire flops of high frequency in the processor. The essence of this approach is to secure concerned software controlled registers. The hardware interrupt will be caused in the case of abnormal toggling events in the secured registers. This method overcomes a considerable drawback in existing approaches. This process guarantees low overhead. A runtime Trojan detection method is also used. Nevertheless there is scope for continuing research in this direction by exploring topics such as optimal parameter tuning, post-fabrication configuration using ReRAMs, and split manufacturing.

III. PROPOSED WORK

The objective of this paper was to survey the various methodologies to detect the Hardware Trojans inserted into the circuits, which is instrumental as a step towards the work to be undertaken, which is the insertion of an undetectable Hardware Trojan in the AES algorithm for key extraction. The Trojan to be inserted needs to be checked against the very limitations of these Hardware Trojan detection methods to ensure its undetectability. The success of a Hardware Trojan lies in its ability to remain hidden and uncaught. If the Trojan of the proposed work manages to thus remain hidden, then its success can be ensured.

IV. CONCLUSION

This paper surveyed the various methodologies to detect the Hardware Trojans inserted into the circuits. For each of the techniques discussed we have considered the issue they set out to solve, the solution itself, the technique employed, the advantage(s) of that solution as well as their limitations.

REFERENCES

Received: 21 Dec 2019 | Revised: 18 Jan 2020 | Accepted: 05 Feb 2020

1. M. Abramovici and P. Bradley. Integrated circuit security: New threats and solutions. In Proc. 5th Annu. Workshop Cyber Security Inf. Intell. Res. Cyber Security Inf. Intell. Challenges Strategies (CSIIRW), Knoxville, TN, USA, 2009, pp. 1–3.
2. C. Bao, D. Forte, A. Srivastava. Temperature Tracking: Toward Robust Run-Time Detection of Hardware Trojans. In Computer-aided Design of Integrated Circuits and Systems, Vol. 34, No. 10, October 2015, pp. 1577-1585
3. R. S. Chakraborty, S. Narasimhan, and S. Bhunia. Hardware Trojan: Threats and emerging solutions. In Proc. IEEE Int. High Level Design Validation Test Workshop (HLDVT), San Francisco, CA, USA, Nov. 2009, pp. 166–171.
4. X. Chen, Q. Liu, S. Yao, J. Wang, Q. Xu, Y. Wang, Y. Liu, H. Yang. Hardware Trojan Detection in Third-Party Digital Intellectual Property Cores by Multilevel Feature Analysis. In Computer-aided Design of Integrated Circuits and Systems, Vol. 37, No. 7, July 2018, pp. 1370-1383
5. X. Chen, L. Wang, Y. Wang, Y. Liu, H. Yang. A General Framework for Hardware Trojan Detection in Digital Circuits by Statistical Learning Algorithms. In Computer-aided Design of Integrated Circuits and Systems, Vol. 36, No. 10, October 2017, pp. 1633-1646
6. X. Cui, K. Ma, L. Shi, and K. Wu. High-level synthesis for run-time hardware Trojan detection and recovery. In Proc. 51st ACM/EDAC/IEEE Design Autom. Conf. (DAC), San Francisco, CA, USA, Jun. 2014, pp. 1–6.S.
7. K. Haider, C. Jin, M. Ahmad, D. M. Shila, O. Khan, M. Dijk. Advancing the State-of-the-Art in Hardware Trojans Detection. In Dependable and Secure Computing, Vol. 16, No.1, January 2017, pp. 18-32
8. Y. Hou, H. He, K. Shamsi, Y. Jin, D. Wu. On-Chip Analog Trojan Detection Framework for Microprocessor Trustworthiness. In Computer-aided Design of Integrated Circuits and Systems, 2018, pp. 1
9. Y. Jin, N. Kupp, and Y. Makris. Experiences in hardware Trojan design and implementation. In Proc. IEEE Int. Workshop Hardw. Orient. Security Trust (HOST), San Francisco, CA, USA, Jul. 2009, pp. 50–57.
10. A. N. Nowroz, K. Hu, F. Koushanfar, S. Reda. Novel Techniques for High-Sensitivity Hardware Trojan Detection Using Thermal and Power Maps. In Computer-aided Design of Integrated Circuits and Systems, Vol. 33, No. 12, December 2014, pp. 1792-1805
11. H. Shen, H. Tan, H. Li, F. Zhang, X. Li. LMDet: A “Naturalness” Statistical Method for Hardware Trojan Detection. In Transactions on Very Large Scale Integration(VLSI) Systems. In IEEE. Vol. 26, No.4, April 2018, pp. 720-732
12. T. Reece and H. Robinson. Detection of Hardware Trojans in Third-Party Intellectual Property Using Untrusted Modules. In Computer-aided Design of Integrated Circuits and Systems, Vol. 36, No. 3, March 2016, pp. 357-366
13. M. Tehranipoor and F. Koushanfar. A survey of hardware Trojan taxonomy and detection. In IEEE., Vol. 27, No. 1. 10–25, Jan./Feb. 2010, pp. 10-25
14. J. Zhang, F. Yuan, L. Wei, Y. Liu, Q. Xu. VeriTrust:Verification for Hardware Trust. In Computer-aided Design of Integrated Circuits and Systems, Vol. 34, No. 7, July 2015, pp. 1148-1161

Received: 21 Dec 2019 | Revised: 18 Jan 2020 | Accepted: 05 Feb 2020

15. B. Zhou, W. Zhang, S. Thambipillai, J.T.K. Jin, V. Chaturvedi, T. Luo. Cost-efficient Accleration of Hardware Trojan Detection Through Fan-Out Cone Analysis and Weighted Random Pattern Technique. In Computer-aided Design of Integrated Circuits and Systems, Vol. 35, No. 5, May 2016, pp. 792-805

16. M. Zou, X. Cui, L. Shi, K. Wu. Potential Trigger Detection for Hardware Trojans. In Computer-aided Design of Integrated Circuits and Systems, Vol. 37, No. 7, July 2018, pp. 1384-1395

Received: 21 Dec 2019 | Revised: 18 Jan 2020 | Accepted: 05 Feb 2020