Electromagnetic Security Vulnerabilities and Instruction Disassembly of Controller in Adaptive Controllers

Varghese Mathew VAIDYAN
Department of Electrical and Computer Engineering
Iowa State University
Ames, Iowa, USA

Akhilesh TYAGI
Department of Electrical and Computer Engineering
Iowa State University
Ames, Iowa, USA

ABSTRACT [1]

A controller in adaptive control theory is a critical part in mission critical applications in military and computer-controlled systems. An ability to identify and follow the binary instruction execution in the controller part enables fault identification and malware detection in safety critical applications. Electromagnetic field emission based identification of controllers execution state from distance will help ascertain security vulnerabilities early on. machine learning models for instruction identification, Principal Component Analysis (PCA), Adaptive Boosting (AB) and Naïve Bayes (NB) were developed to meet this goal. Our preliminary results of implementation on a 2-stage pipelined controller processor architecture demonstrate that the EM side-channel classification approach identifies a controller execution state in Adaptive control with 93% success rate.

Keywords: Instruction Disassembly, Machine Learning, Hardware Security, IoT Devices, Computer Architecture, Electromagnetics.

1. INTRODUCTION

Adaptive control, specifically self-tuning regulator (STR) is an important strategy for many mission critical systems like computer-controlled systems, missiles, and other defense systems. Practical implementations of self- tuning regulators are mainly realized using microcomputers or microcontrollers, and other processors ([1]-[3]). The controller part is one of the most important parts in adaptive control. Due to the mission critical nature of adaptive control, instruction disassembly and in turn reverse engineering the controller and subsequent security vulnerability identification is crucial. Moreover, malware is emerging as a new battleground in cybersecurity. Recent attacks like Mirai and Moose highlight the need to defend and identify faults early on in these devices [3], [4], [5], [6], [7],[8], [9], [10], [11], [12], [13], [14],[15],[16],[17]. The fundamental challenge in instruction disassembly is precisely disassembling instructions in a black-box environment. Hardware side-channel methods have unique and desirable capabilities in these scenarios as they do not require access to the executing binary. Instruction disassembly of the controller through Electromagnetic(EM) or power side-channel is a significantly challenging problem. Moreover, this needs to be done in a single clock cycle of the order of 0.25ns for a 4GHz processor. Even more, entire disassembly of the controller is a challenge in modern processors. We propose an electromagnetic spectrum-based controller security vulnerability identification by identifying instructions in the pipeline. In particular, our approach is the first to analyze electromagnetic characteristics of controller in Adaptive control design. We were able to identify and estimate the controller state with a high success rate. It also does not pose any requirements in terms of physical access to the device.

Our contributions are summarized as follows:

- We propose a novel at-distance electromagnetic spectrum domain approach of controller disassembly in adaptive control that exposes vulnerabilities and performance problems in critical Adaptive control
- We develop an electromagnetic spectrum domain framework, based on dimensionality reduction and feature selection using Principal Component Analysis for a set of features for the high probability individual instructions in the processor as a training library first. Subsequently, machine learning, classifiers like Adaptive Boosting and Naïve Bayes, are used to identify the controller operation in flight.

[1] We would like to thank Mr. Chris Nuttal of Iowa State University for reviewing and proof-reading the final version of the paper.

- Our experiments on ATMeg328 demonstrate that our technique can disassemble an entire controller operation in flight with 93% accuracy.

The rest of the paper is organized as follows. In Section 2, importance of controller in Adaptive control is discussed. Section 3 details EM Characterization on the device under test. Section 4 presents feature selection and Machine learning approaches. Functional evaluation is presented in section 5. Details of hardware implementation are given in Section 6. Hardware results and evaluation of performance of classifiers are discussed in Section 7. Section 8 summarizes the work.

2. IMPORTANCE OF CONTROLLER AND FAULTLESS OPERATION IN ADAPTIVE CONTROL

The schematic diagram of a Self tuning regulator is represented in Fig.1. Minimum Degree Pole Placement block is an important part of stochastic self-tuning regulator. Consider A and B denoting two polynomials which do not have any common factors in neither differential operator, $p = d/dt$ and forward shift operator q. Also, let A be $monic$.

From the above assumptions, a process can be defined as a single-input, single-output (SISO) system [1],

$$Ay(t) = B(u(t) + v(t)) \qquad (1)$$

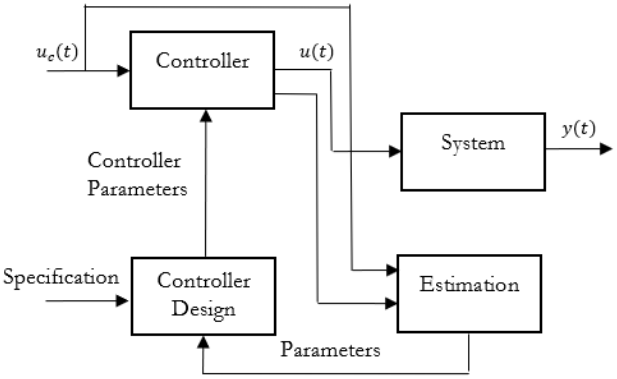

Figure 1 Stochastic Self Tuning Regulator

and, a general linear controller can be given by,

$$Ru(t) = Tu_c(t) - Sy(t), \qquad (2)$$

Here, v is the disturbance, u is the input, $u_c(t)$ is the command signal and, R, S and $T$are polynomials. Then, a transfer operator T/R feed forward and transfer operator $-S/R$ negative feedback is represented by control law. Hence, identifying the faults in the controller early on is very crucial in successful operation and performance capabilities in safety critical applications.

A faulty controller can result from an addition of new lines or tampering of instructions by an adversary and malware. An EM instruction disassembly based fault identification helps in detecting if the controller is tampered with based on a comparison with the fault free model available to us. For an illustration, let the fault free model for the controller consist of instructions $i_1, i_2, i_3, ...$ However, if a malware tampers the code and if the controller instructions become $i_1, i_4, i_3, ...$, it can be detected through EM level execution state identification compared against the fault-free model. Such a comparison against a fault-free golden model is efficient. This fault identification can be performed selectively based on the importance of an instruction to some other controller property for further efficiency enhancements. If we do not have the binaries for legacy controllers for a fault-free model, we still will be able to predict the executing controller instructions based on the available training model of individual instructions. The fault-free model could be developed from multiple legacy controllers through consensus between the EM extracted instructions streams.

3. EM ANALYSIS

Static power consumption of circuits can be defined as: $P_{static} = V_{DD} I_{supply}$. This is primarily due to the leakage from V_{DD} to Ground. In contrast, in the dynamic switching case, charging and discharging of different capacitances is from input switching. And as a consequence, dynamic power consumption is related to the signal frequency. Moreover, for small fall and rise times, dynamic power consumption is entirely related to the energy for charging and discharging of the load capacitances. Dynamic power can be defined as, $P_{dynamic} = C_L V_{DD}^2 f$, where C_L is the total load capacitance whereas f represents signal frequency. Thereby, Dynamic power and consequently Electromagnetic waves related to it are very reliable for identifying the instructions being executed on the processor – in particular because of the correlation from power signature with data value/switching. The EM probe antenna factor can be defined as the ratio of magnetic or electric field of the DUT to the induced voltage of the probe, $AF = H(dB) - V(dB)$. Consider an antenna having radius a, s as the coordinate around the loop perimeter, l as the total loop perimeter and let us assume a is small and also that the loop is symmetrical about axis and has incident field variation, $e^{j\omega t} = e^{jkct}$, we can represent relation $-jk \int_s cB.dS = \oint_s E.ds$ [18]. Here, cB denotes the magnetic field and has the same dimensions of E. Sum of incident field B^i and reradiated field B^τ represent the magnetic field. Using Helmholtz integrals and Ohm's law, and further splitting current to zero phase sequence current and first phase sequence current, we get $I(s) = I^0(s) + I^1(s)$, $I\left(s + \frac{l}{2}\right) = I^0(s) - I^1(s)$. Based on this, the general integral equation for zero phase sequence current case becomes $-jk \iint_s cB^i.dS =$

$\oint_s \ I^0(s)Z^i ds + \frac{j\omega\mu}{4\pi}\oint\oint I^0(s)\frac{e^{-jkR}}{R}ds.ds$. Here, R is the distance between field and source points and $k = \frac{2\pi}{\lambda}, Z^i$ is the internal impedance/unit length. Accordingly, a general case zero-phase sequence current can be inferred to be related to the incident magnetic field, $\boldsymbol{B^i}$.

For a small enough loop, integral of the incident magnetic field becomes B_{z0}^i, which is equivalent of removing even derivatives of Taylor series expansion for $\boldsymbol{B^i}$ at the loop center. Based on that, a first order approximation of zero phase-sequence current can be derived based on constant current I_0. As we know, in constant current scenarios, the low-frequency input admittance of the loop is $Y_0 = \left[\oint Z^i ds + \frac{j\omega\mu}{4\pi}\oint\oint I^0(s)\frac{e^{-jkR}}{R}ds.ds\right]^{-1}$ and we have $h_b = -jkS$.

Based on approximations in last paragraph and $K_B = Y_0 h_b/\boldsymbol{\lambda}$, the unloaded magnetic sensitivity constant K_B can be defined as being dependent on the probe geometry. Accordingly, solution for $I^0(0)$ can be identified with: $I^0(0) \approx I_0 = \boldsymbol{\lambda} K_B(cB_{se}^i)$. Besides, because the electric field on the plane of loop does not enter I_0, I_0 is effectively the magnetic field at loop center.

Let us take the dipole mode which is primarily the first-phase-sequence current $\boldsymbol{I^{(1)}}$, and directly dependent to the electric field. It can be broken to two parts: one which is symmetric across x-axis $I_x^{(1)}$ and other across y-axis $I_y^{(1)}$. Besides, $I_x^{(1)}$ and $I_y^{(1)}$ are further related to E_{y0}^i and E_{z0}^i. Owing to the fact that the loads mainly are restricted to the ones at $s = 0$ or $l/2$ only, $I_x^{(1)}$ can be eliminated while computing load currents. Consequently, the current corresponding to at $s = 0$ can be represented as: $I_y^{(1)}$ (0) = $h_{eI} Y_I E_{y0}^i$, where Y_I, being the input admittance at the center of antenna can be determined by solving the antenna problem directly. Furthermore, h_{eI} can be determined from Rayleigh-Carson reciprocal theorem for a two-port passive system. Implying, $kh_{eI} = F_I$, where F_I denotes the far-zone field factor at each half on the midplane in the broadside direction. Considering I_y to be y component during transmission and I_{y0} is the value at driving point, F_I can be determined as $F_I = \frac{k}{I_{y0}}\int_{-\boldsymbol{l/4}}^{\boldsymbol{l/4}} I_y(s)ds$. That yields unloaded electric sensitivity to be: $h_{eI} = \frac{2}{I_{y0}}\int_0^{l/4} I_y(s)ds$. Furthermore, relation for $I_y^{(1)}$ (0) can alternatively be written as $I_y^{(1)}$ (0) = $\lambda K_E E_{y0}^i$, where K_E denotes $h_{eI} Y_I/\lambda$. In addition, $I_y^{(1)}$ can be used as a measure of parallel component of the electric field because the average magnetic field perpendicular to the plane will not enter $I_y^{(1)}$. From compensation theorem, it is possible to replace load by the equivalent generator $V = -IZ_L$ in case of single load. That implies, the sum of the transmitting current generated from equivalent generator and the current from unloaded receiving loop due to external fields gives the effective current. Consider $I^T(s) = Vv(s)$, that implies: $I(0) = I^{(0)}(0) + I_y^{(1)}(0) - I(0)Z_L v(0)$. But, $v(0)$ is the total input admittance Y for the loop when being driven during $s = 0$. Based on that, load current is: $I_L = I(0) = \lambda K_B^{(1)} cB_{z0}^i + \lambda K_E^{(1)} E_{y0}^i$, with two single loaded sensitivity constants: $K_B^{(1)} = \frac{Y_L}{Y_L + Y} K_B$ and $K_E^{(1)} = \frac{Y_L}{Y_L + Y} K_E$. For the loop when loaded at $s = \frac{l}{2}$, which is similar to rotating 180^0 in its own plane, the load current is $I_L' = I(l/2) = \lambda K_B^{(1)} cB_{z0}^i - \lambda K_E^{(1)} E_{y0}^i$. Therefore, both the I_L and I_L' readings are required to measure the magnetic field.

4. CLASSIFICATION APPROACH

The EM traces from experiments have large number of sampling points. This in turn produces a high dimensionality problem. A mapping, $x \rightarrow Wx$, of the large dimension EM signature having $Wx \in R^n$ to be the lower dimensionality representation of x, and matrix $W \in R^{n,d}$, with $n < d$, lowers the dimensionality of vectors $x_1, x_2, \ldots\ldots, x_m$. Principal Component Analysis (PCA) is the best way to find the compression matrix W and for recovering matrix U, thereby making total squared distance between both to be minimal [15].

1.1 Adaptive Boosting

Adaptive Boosting (AdaBoost), a low empirical risk method, identifies a hypothesis on an EM signatures training set, $S = (x_1, y_1), \ldots\ldots\ldots, (x_m, y_m)$, dependent on labelling function f for each i, $y_i = f(x_i)$ [19]. Accordingly, a sequence of consecutive rounds are calculated. Here booster defines a distribution, $\mathbf{D}^{(t)}$ in S for a certain round t. That implies, $\mathbf{D}^{(t)} \in R_+^m$ and $\sum_{i=1}^m D_i^{(t)} = 1$. Then distribution and sample are passed on to the weak learner. Using distribution $\mathbf{D}^{(t)}$ and f, weak learner builds several *i.i.d.* examples. Weak hypothesis, h_t, which is error from weak learner can be represented as,

$\epsilon_t \overset{\text{def}}{=} L_{\mathbf{D}^{(t)}}(h_t) \overset{\text{def}}{=} \sum_{i=1}^m D_i^{(t)} 1_{[h_t(x_i) \neq y_i]} \leq \frac{1}{2} - \Upsilon$. The algorithm assigns weights related inversely to the error of h_t given as $w_t = \frac{1}{2} log\left(\frac{1}{\epsilon_t} - 1\right)$. A higher probability mass gets assigned if h_t increases the error and a lower probability mass if h_t reduces the error. Consequently, output is based on weighted sum of all the weak hypotheses.

1.2 Naïve Bayes

For an event, $E = (x_1, x_2, \ldots\ldots\ldots, x_n)$ for class c, based on Bayes rule, the probability becomes, $p(c|E) = [p(E|c)p(c)]/p(E)$. Here, E can be classified to be in a class $C = +$ when

$$f_b(E) = \frac{p(C = +|E)}{p(C = -|E)} \geq 1, \qquad (3)$$

where $f_b(E)$ is the Bayesian classifier. Based on the assumption of independence among attributes, class variable, $p(E|c) = p(x_1, x_2, \dots \dots, x_n|c) = \prod_{i=1}^{n} p(x_i|c)$, Naïve Bayes classifier can be defined as,

$$f_{nb}(E) = \frac{p(C=+|E)}{p(C=-|E)} \prod_{i=1}^{n} \frac{p(x_i|C=+|E)}{p(x_i|C=-|E)} \quad (4)$$

5. FUNCTIONAL EVALUATION

Functionality of a controller was verified on a Self-tuning regulator in MATLAB. A unit amplitude square wave command signal was used as the reference input. In the initial phase, we get an oscillatory process output, owing to the estimation error. However, in the next phase, estimated parameters converge to true parameters and the system stabilizes as in Fig.2.

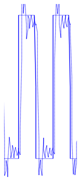

Figure 2 Process oscillations after initial transient state.

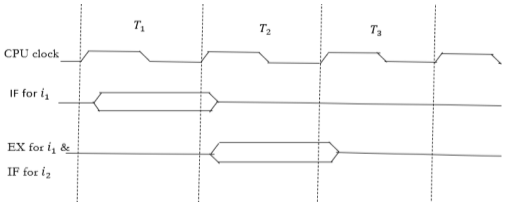

Figure 3 Two-stage Pipeline Operation of DUT

6. HARDWARE IMPLEMENTATION

The Device Under Test (DUT) running the controller is given in Fig.4. The feature vectors captured for training and classification of individual opcode of instructions in the controller is primarily due to the EM spectrum variations owing to the CMOS switching on every execution stage of individual instructions. In Fig.5, details of the DUT on a single instruction execution are outlined showing different stages like register operand fetch, ALU operation execute and Result write back stages. As we can see in Fig.5., the major challenge is that instruction execute of the first instruction and Instruction Fetch of next instruction overlaps because of the pipeline. This also entangles the side-channel power and EM signals of two overlapping instructions. Identification of two such overlapping instructions in pipeline is the main challenge in identifying instructions in controller. In this work, we mainly focus on this overlapping region of instructions with models to separate the leading instruction from the following instructions. The controller polynomial values were assumed to be arbitrary but reasonably inside real time limits. The hardware setup developed for the Instruction Disassembly experiments is given in Fig.4. The EM traces from the controller instructions in pipeline of Atmega328 were first received on the TPBS01 EM probe. Additional traces were captured on DPO 4032 oscilloscope with a bandwidth of 350MHz and sampling rate of 1.5GS/s. For the experiment, the EM probe was kept as a receiver at a 10cm distance from the DUT. The calibration of Oscilloscope and probe was by a sleep and trigger mechanism on the Atmega328. Subsequently, EM trace was obtained through a UART interface and then was preprocessed for the EM signal analysis. Further, to identify the probability of controller instructions in the frequency spectrum, it was transformed to spectral domain. Owing to the large dimensional feature space, dimensionality was reduced to 50 dimensions with principal component analysis. Subsequently, the processed data was trained with feature vector of individual instructions $i_1 i_2$,.... to 100% accuracy into different classes. Furthermore, traces from controller were identified using Adaptive Boosting (AdaBoost) and Naïve Bayes. Finally, the success rates of controller identification was evaluated using the ML classifiers.

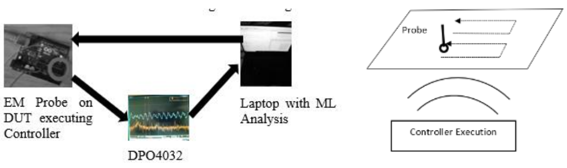

Figure 4. Hardware implementation for controller instruction disassembly

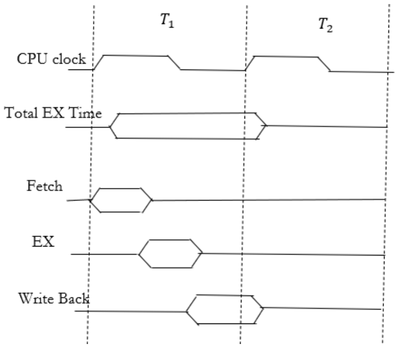

Figure 5 Single cycle ALU Operation of DUT

7. HARDWARE RESULTS AND EVALUATION

We recorded a stream of controller instructions in flight on the ATmega328. The onset of every EX stage of new instruction manifested onto the EM spectrum. For each instruction for the controller, it was classified based on models of the pre-trained independent instructions. Initially, EM spectrum had 200 dimensions for every instruction. Fig. 6 shows the important principal components with highest variance to be the first 8 components. Principal component analysis was used for dimensionality reduction.

In the experiment, individual instructions were run in a stream inside a loop with loop count, $n = 200$. Hence, we get 200 instances of different individual instruction classes. For identifying the controller instructions in flight, the controller was run in loop of loop count $n = 200$ as in real Adaptive control situation where control signal must be generated consistently. Further, Machine learning classifiers were used to identify the instructions of the controller running on the processor. We characterized the performance of different Machine Learning classifiers in identifying the controller based on prediction capability of overlapping instructions due to the pipeline which is summarized in Fig.7.

For performance evaluation, Machine Learning classifiers like AdaBoost and Naïve Bayes were programmed in Python. AdaBoost with 70 n_estimators and a learning_rate of 0.3 showed the best performance. In our earlier experiments of two instruction streams, AdaBoost showed reliable success rates and hence we give higher weightage to AdaBoost.

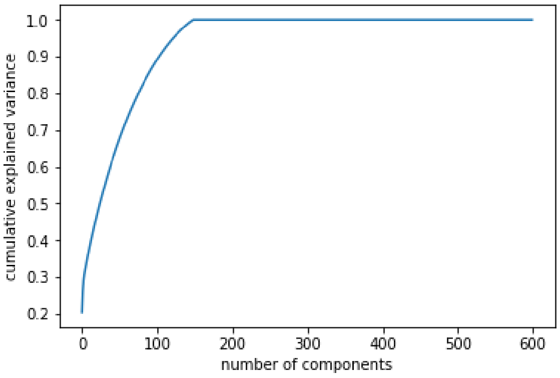

Figure 6 Cumulative Explained Variance of Training data of Instructions

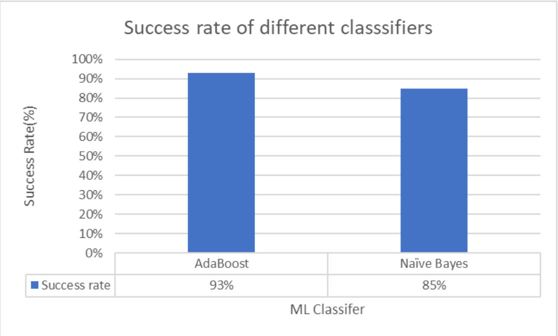

Figure 7. Performance evaluation of Classification approaches for controller

8. CONCLUSIONS

A novel Electromagnetic spectrum based Instruction disassembly and fault identification of controller in Adaptive control on a 2-stage pipelined architecture is introduced. The machine learning training models can be built and instructions in the controller can be identified without any alterations to the device. The training feature vectors for individual instructions were developed instead of instruction combinations or groups, in turn reducing combinatorial complexity of hierarchical classification. Performance evaluation with Adaptive Boosting (AB) and Naïve Bayes (NB) for the controller was conducted. Over 90% accuracy in instruction identification in spite of adjacent instruction interference in the pipeline is achieved. Since this EM approach can operate at distance from DUT without needing to alter the device or interfere with operation of the controller, it opens up many more possibilities in code reverse engineering and fault identification of Adaptive Control.

9. FUTURE WORK

The electromagnetic spectrum-based controller instruction disassembly of Adaptive control showed promise in predicting and identifying instructions in flight in a 2-stage pipeline. In future, we expect to identify entire Adaptive control system including Minimum degree pole placement.

10. ACKNOWLEDGMENTS

The work was supported in part by DARPA/AFRL through Contract FA8750-20-1-0504.

12. REFERENCES

[1] K.J. Astrom, and B. Wittenmark, On self-tuning regulators",**Automatica**, ,Vol. 9, No. 2, 1973, pp. 185–199.

[2] D.W. Clarke, and P .J. Gawthrop, "Self-tuning controller", **Proceedings of IEE**, Vol. 122, No. 9, 1973, pp. 929–934.

[3] M. A. Sheirah, O.P. Malik, G. S. Hope, "Self-tuning microprocessor universal controller", **IEEE Transactions on Industrial Electronics**, Vol. IE-29, No.1, 1982, pp. 31–38.

[4] M. Abadi, M. Budiu, U. Erlingsson, and J. Ligatti, "Control-flow integrity: principles, implementations, and applications", **ACM Transactions on Information and System Security**, Vol. 13, No. 1, 2009, pp. 4:1-4:40.

[5] L. Davi, A. Dmitrienko, M. Egele, T. Fischer, T. Holz,

R. Hund, S. Nurnberger and A.R Sadeghi, "MoCFI: A framework to mitigate control-flow attacks on smartphones", **In Network and Distributed System Security Symposium (NDSS),** 2012.

[6] R. Wartell, V. Mohan, K. Hamlen, and Z. Lin, "Securing untrusted code via compiler-agnostic binary rewriting", **In Proceedings of the 28th Annual Computer Security Applications Conference** (ACSAC'12) , 2012, pp. 299–308.

[7] X. Chen, A. Slowinska, D. Andriesse, H.Bos, and C. Giuffrida, "Stackarmor: Comprehensive protection from stack-based memory error vulnerabilities for binaries", **In Network and Distributed System Security Symposium,** 2015, pp. 8-11.

[8] V. Van der Veen, D. Andriesse, E. Goktas, B. Gras, L. Sambuc, A. Slowinska, H. Bos, and C. Giuffrid, "Practical context-sensitive cfi", **In Proceedings of the 22nd ACM Conference on Computer and Communications Security (CCS)**, 2015, pp. 927–940.

[9] V. Van der Veen, E.Goktas, M. Contag, A. Pawlowski, X. Chen, S. Rawat, H. Bos, T. Holz, E. Athanasopoulos, and C. Giuffrida, "A tough call: Mitigating advanced code-reuse attacks at the binary level", **In Proceedings of the 37th IEEE Symposium on Security and Privacy**, 2016, pp. 934-953.

[10]X. Chen, H. Bos, and C. Giuffrida, "CodeArmor: Virtualizing the Code Space to Counter Disclosure Attacks", **In Proceedings of 2017 IEEE European Symposium on Security and Privacy**, 2017, pp. 514-529.

[11]A.Sæbjørnsen, J.Willcock, T.Panas, D.Quinlan, and Z. Su, "Detecting code clones in binary executables," **in Proceedings of the Eighteenth International Symposium on Software Testing and Analysis** 2009, pp. 117–128.

[12]D. Gopan, E.Driscoll, D.Nguyen, D.Naydich, A.Loginov, and D.Melski, "Data-delineation in software binaries and its application to buffer-overrun discovery", **In Proceedings of the 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering**, 2015, pp. 145-155.

[13]E.Tilevich, and Y.Smaragdakis, "Binary refactoring: Improving code behind the scenes", **In Proceedings of the 27th International Conference on Software Engineering (ICSE'05)**, 2005, pp. 264-273.

[14]Ericsson. November 2018, the connected future - internet of things forecast, 2018.

[15]O.Bilodeau, and T.Dupuy, "Dissecting Linux/Moose the Analysis of a Linux Router-based Worm Hungry for Social Networks", **Technical Report**, 2015.

[16]J.Park, and A.Tyagi, "Using power clues to hack IoT devices: The power side channel provides for instruction-level disassembly" **IEEE Consumer Electronics Magazine**, Vol. 6, No. 3, 2017, pp.92–102.

[17]V.M.Vaidyan, and A.Tyagi, "Instruction Level Disassembly through Electromagnetic Side-Chanel: Machine Learning Classification Approach with Reduced Combinatorial Complexity", **Proceedings of 3rd International Conference on Machine Learning and Signal Processing**, 2020.

[18]H.Whiteside, and R.King, "The loop antenna as a probe", **IEEE Transactions on Antenna and Propagation**, Vol. 12, No. 3, 1964, pp. 291-297

[19]S.S. Shwartz, S.B. David, "Understanding Machine Learning: From Theory to Algorithms", **Cambridge University Press, UK, 2014.**